Clinical Relevance of ML Predictions using Health Datasets

by

Sowmya Balasubramanian

B.Sc, University of Madras, 1997

MCA, Madurai Kamaraj University, 2001

M.Sc., University of Victoria, 2013

A Dissertation Submitted in Partial Fulfillment of the

Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Computer Science

© Sowmya Balasubramanian, 2024

University of Victoria

All rights reserved. This dissertation may not be reproduced in whole or in part, by photocopying or other means, without the permission of the author.

Clinical Relevance of ML Predictions using Health Datasets

by

Sowmya Balasubramanian

B.Sc, University of Madras, 1997

MCA, Madurai Kamaraj University, 2001

M.Sc., University of Victoria, 2013

Supervisory Committee

Dr. Alex Thomo, Supervisor

(Department of Computer Science)

Prof. Kui Wu, Departmental Member

(Department of Computer Science)

Prof. Xuekui Zhang, Outside Member

(Department of Mathematics and Statistics)

ABSTRACT

This research explores how advanced data analysis and machine learning techniques can transform healthcare. By applying innovative computational methods, it addresses key challenges in medical diagnosis, data interpretation, and synthetic data generation. The aim is to enhance clinical practices, improve diagnostic accuracy, and maximize the utility of healthcare data. Through these cutting-edge approaches, the research seeks to provide more effective, accurate, and practical solutions for modern healthcare needs.

The first study tackles the challenge of diagnosing thyroid disorders, which can profoundly affect both physical and mental health. Instead of merely evaluating classifier performance, this research emphasizes the value of comprehensive feature analysis. Identifying the top four crucial features for predicting thyroid disorders, shows that using a complete thyroid panel leads to more accurate and cost-effective diagnoses. This approach reveals the flaws in current clinical practices that frequently skip a full thyroid panel, particularly in universal healthcare systems.

The second study delves into diagnosing Autism Spectrum Disorder (ASD) using fMRI data. It compares simple tabular data classifiers with cutting-edge graph-theoretic methods, discovering that the simpler approach performs just as well. Moreover, the research finds that adding higher-order connectivity information doesn't improve classification outcomes, and highlights the complexity of diagnosing ASD due to the similar brain networks in individuals with and without the disorder. This study underscores the necessity for clear and reliable diagnostic methods in neuroimaging.

The third study explores the creation of synthetic health data, which is essential for research and practical applications while addressing privacy and ethical concerns. It assesses various cutting-edge synthetic data generation (SDG) techniques, comparing their scalability, resemblance to real data, and practical utility. The findings reveal

that statistical models surpass machine learning-based methods in training time and data generation, with synthetic data from most models closely mimicking real data. This synthetic data proves highly valuable in practical applications, maintaining high classification accuracy whether trained on real or synthetic data.

In summary, this dissertation shows how advanced data analysis and machine learning can significantly improve healthcare. By enhancing diagnostic methods, making complex medical data easier to understand, and ethically using synthetic data, these techniques lead to more effective, accurate, and scalable healthcare solutions. This work demonstrates the powerful role of technology in advancing patient care.

Contents

List of Tables

List of Figures

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor, Prof. Alex Thomo, for his unwavering support and encouragement throughout this research. His invaluable feedback has greatly enhanced the quality of this dissertation, which would not have been possible without his guidance. I am also profoundly grateful to Prof. Kui Wu and Prof. Xuekui Zhang for their insightful guidance as members of my supervisory committee. A special thanks to Prof. Fan Jiang, who graciously accepted to be the external examiner for my oral examination.

I am deeply thankful to my collaborators, Prof. Smita Ghosh, Kazi Tabassum Ferdous, Bryan Maruyama, and Joe Howie, for their significant contributions and the many enjoyable discussions we shared. I extend my heartfelt thanks to Prof. Alex Thomo for the Data Mining course, Prof. Charles Perin for the Information Visualization course, Prof. Bruce Kapron for the Cryptography course, and Prof. Lisa Lix for the Foundations of Disease Analytics course. The knowledge gained from these courses has been instrumental in shaping this dissertation.

I would like to thank Prof. Hausi Muller for encouraging me to pursue a doctoral degree. I take this opportunity to extend my sincere thanks to all the staff members in the Computer Science Department, especially Wendy Beggs, Nancy Chan, J Cameron, Aimee Coueslan, and Erin Robinson, for their constant assistance with administrative tasks. Your support has been invaluable.

Last but not least, my heartfelt gratitude goes to my family for their unwavering support through this long journey. To my son, my husband, my dad and mom, and my sister – your unconditional love and encouragement have been my strength.

DEDICATION

I dedicate this dissertation *at thy divine lotus feet*

Chapter 1

Introduction

This dissertation explores three critical areas where advanced data analysis and machine learning are transforming healthcare. Firstly, it investigates the enhancement of diagnostic accuracy in thyroid disorders through comprehensive feature analysis, emphasizing the importance of a complete thyroid panel for effective clinical decision-making. Secondly, the study delves into Autism Spectrum Disorder (ASD) diagnosis using fMRI data, comparing traditional tabular data classifiers with advanced graph-theoretic methods to better understand the complex brain networks involved. Thirdly, it explores the generation of synthetic health data, evaluating various techniques to create data that mirrors real-world scenarios without compromising privacy or ethics.

1.1 Diagnosis of Thyroid Disorder

The endocrine system is a collection of glands that produce the hormones that control nearly all the important biological processes of the human body. These glands, situated in different parts of our body, include the hypothalamus, pituitary gland, and pineal gland in the brain; the thyroid and parathyroid glands in the neck; the thymus gland between the lungs; the adrenal gland on top of the kidneys; the pancreas behind

the stomach, and the ovaries or testes in the pelvic region [10]. All these glands work together to help us lead a normal and healthy life.

The thyroid gland is a butterfly shaped gland that releases three thyroid hormones into the bloodstream, levothyroxine (T4), triiodothyronine (T3), and calcitonin that regulate the body's metabolic rate, controlling heart, muscle, and digestive function, brain development, and bone maintenance [40]. An optimal supply of iodine from our diet aids in the correct functioning of the thyroid gland [43]. Malfunction of this gland results in thyroid disorders causing extreme fatigue, trouble sleeping, enlarged thyroid gland (goiter), vision problems, muscle weakness, weight gain or weight loss, intolerance to heat or cold, and many more symptoms [33].

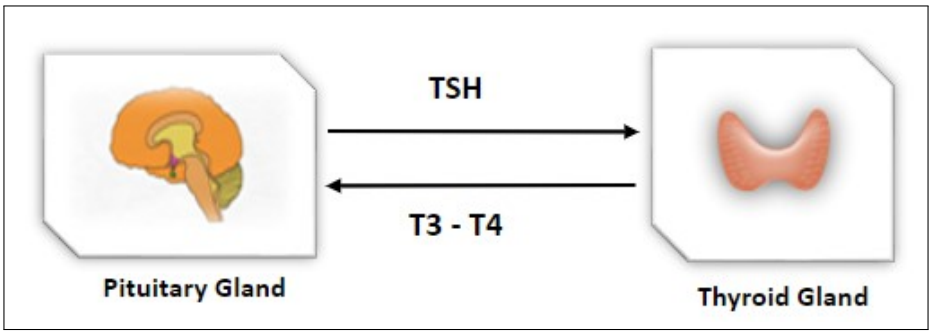


Figure 1.1: Pituitary Gland and Thyroid Gland: Relationship

The pituitary gland is a pea-sized gland located at the base of the brain below the hypothalamus that regulates other endocrine glands to release hormones [15]. Pituitary gland signals the amount of T3 and T4 the thyroid gland needs to release into the bloodstream by producing a regulating hormone called thyroid-stimulating hormone (TSH) [15] as shown symbolically in Fig. 1.1. A normal healthy thyroid shows an optimal balance of TSH, T3, and T4 hormones. However, when the thyroid does not produce the right amount of hormones it causes a thyroid disorder or disease. Hypothyroidism is a condition of having too much TSH in the bloodstream indicating that the thyroid gland is not making enough T3 or T4 [33]. Hyperthyroidism is

a condition of having low TSH levels showing that the thyroid gland is producing too much T3 and T4 [33]. As thyroid hormones act on virtually every cell in the body to alter gene transcription, abnormal production of these hormones has strong effects [14].

Thyroid disease is one of the most common health conditions in adults that often remains undiagnosed. The Thyroid Foundation of Canada website indicates that 1 in 10 Canadians suffer from a thyroid condition and 50% of them are undiagnosed [69]. American Thyroid Association estimates that around 20 million Americans have some form of thyroid disease and around 60 percent have not been diagnosed [9]. Also, available statistics suggest that women may have up to eight times higher chance than men of getting a thyroid disorder [9]. An undiagnosed variation in the level of thyroid hormones can have a phenomenal effect on one's physical and mental health.

Undiagnosed hypothyroidism results in a life-threatening condition called Myxedema with intense cold intolerance and extreme lethargy leading to Myxedema coma. Recently, the thyroid disease risk has been found to be linked with systemic lupus erythematosus, an autoimmune disease [96]. There is also evidence of a correlation between thyroid disease and low levels of vitamin D [49]. Clinicians use a simple blood test to diagnose thyroid disorders. It measures the amount of thyroid hormones in the bloodstream and reports if the person has a thyroid disorder or not. Such a test can either measure the complete thyroid panel (TSH, FT4 (Free T4), and FT3 (Free T3)) or only the TSH as a cost saving way. I observed that the clinical practice of measuring TSH only is common in countries with universal health care [32, 13]. In order to understand the efficacy of these tests and provide clinicians with information they can trust, I initiate a systematic study of feature importance in the diagnosis of thyroid disorder.

This work develops a predictive model to facilitate early diagnosis of this disease.

Using data mining techniques, I identify important predictors of thyroid disease that enable clinicians to diagnose this thyroid disease effectively. Our main contributions in this study are as follows:

1. Using an explainable classifier (CART), I show that the top-4 features for thyroid diagnosis are FTI, TSH, TT4, and T3 (See Table 2.3) which can be measured cost effectively using a simple blood test.

2. I also show that a clinical test in which TSH is measured in isolation is not sufficient leading to possible misdiagnosis of the disorder.

3. I perform experiments to demonstrate the robustness of our conclusions using different approaches, including the use of a new classifier (Random Forest), principal component analysis, and methods to handle the imbalance in our dataset.

4. Finally, I also shed light on an ambiguity in the existing literature when defining class labels for the thyroid dataset I use and provide explanations to resolve it.

1.2 ASD Diagnosis from fMRI Scans

Autism Spectrum Disorder (ASD) is a developmental disorder caused by abnormal brain development, and encompasses a wide range of symptoms and severity levels, varying from mild to severe [30]. Individuals with ASD may have co-occurring conditions such as Attention Deficit Hyperactivity Disorder (ADHD), anxiety, or depression, which need to be addressed for comprehensive support [31]. They also face challenges in social interactions, exhibit repetitive behaviors, and often have heightened sensitivity (hypersensitivity) or reduced sensitivity (hyposensitivity) to stimuli

like light, touch, taste, or smell [53]. Despite this, they also possess remarkable strengths, such as visual thinking and problem-solving abilities.

The exact cause of ASD is not fully understood, but research suggests that it results from a complex interplay of genetic and environmental factors [28, 39, 21, 53]. Genetic conditions like Fragile X Syndrome and Tuberous Sclerosis increase the risk of developing ASD [58]. Environmental influences, including premature birth and maternal alcohol use during pregnancy [58], are suggested potential risk factors. ASD Symptoms typically emerge in early childhood affecting approximately 1 in 36 children, with boys being diagnosed four times higher than girls. It occurs across different racial, ethnic, and socioeconomic backgrounds without specific limitations [16]. Although there is no cure for ASD, early intervention, specialized services, and parental support improve a child's growth and development [44].

Diagnosing ASD requires a comprehensive specialist evaluation [64] and a detailed clinical assessment based on specific criteria outlined in diagnostic manuals like the DSM-5 [6]. Nevertheless, this traditional diagnostic approach lacks definitive laboratory tests and relies heavily on clinical judgment and behavioral observations. Therefore, it is important to employ reliable methods to improve ASD diagnosis for all ages.

The discovery of Functional MRI (fMRI), a modern brain imaging technique, has enabled researchers to identify and partition the brain into regions of interest (ROIs) based on their specific functions. By constructing a graph from a fMRI scan, with ROIs as vertices and edges representing the co-activation of these regions, researchers can employ graph classification techniques to effectively classify fMRI scans [12]. Several techniques have been proposed for general graph classification such as kernel methods [**?**], graph embeddings [35], and deep learning [50]. Metrics such as accuracy, precision, and recall are essential for evaluating any such classifier [26], and it has been

shown that some of these methods can achieve impressively high scores for the various metrics.

However, a drawback of these techniques is their complexity, large number of parameters, and black-box nature making it challenging to understand their predictions. Recently, there has been a growing focus on *explainability* within the AI domain [36, 55]. In critical sectors like healthcare, decision-makers are hesitant to adopt prediction models solely based on the high reported accuracy without comprehending their decision-making processes [8]. This cautious approach is especially crucial in healthcare, where *explainability* is vital for gaining the trust of medical practitioners [85].

In response to the need for explainability, Lanciano *et al.* [51] used *contrast subgraph* method for diagnosing ASD. The goal is to find subgraphs in brain connectivity data that display dense connections among individuals with ASD while being sparse in neurotypical individuals, or vice versa. This approach aims to create an interpretable classification method revealing unique brain connectivity patterns in individuals with ASD. However, computing contrast subgraphs is complex and computationally intensive. In a recent study, Enns *et al.* [27] proposed a simpler *discriminative edges method*, which identifies the most important edges or connections that help distinguish individuals with ASD from neurotypical individuals. As shown in [27], both these methods obtained a mean accuracy of 60% on larger datasets of individuals with ASD. In light of these results, Enns *et al.* [27] poses the following question: Can brain imaging data lead to more accurate ASD diagnoses while maintaining explainability? If not, can I determine the reasons behind this limitation?

In our research, I seek to address this question by exploring an alternative pathway for explainable ASD diagnosis methods, complementing the findings of Lanciano *et al.* [51] and Enns *et al.* [27]. Our work views graphs as tables and focuses on

demonstrating the effectiveness of simple and explainable tabular ML methods as alternatives to the graph techniques utilized in prior studies (e.g., [51, 27]). Furthermore, with the goal of improving the accuracy of our method, I explore the possibility of adding higher-order information as attributes to aid classification. While the methods I propose are simple and explainable, I observe that they did not achieve high accuracy though they matched the performance of the previous methods. Therefore, I investigate the potential barriers that hinder the achievement of strong performance metrics, aiming to provide insights into the question raised by [27]. Our main contributions in this study are as follows:

1. Converting the brain network data into a tabular format and using explainable classifiers yield comparable results to graph-theoretic techniques used in prior works.

2. Incorporating higher-order connectivity patterns, the number of triangles in a node's neighbourhood, as attributes do not improve the classifier performance.

3. Studying similarities between brain networks of individuals with and without ASD, using similarity measures such as Jaccard similarity of k-cores and Hamming distance reveals the underlying barriers to ASD prediction.

1.3 Synthetic Generation of PSU Data

In today's data-driven world, finding valuable insights from the available data is crucial. Understanding data relies on quality attributes like accuracy, timeliness, consistency, relevance, and completeness. These are vital for making informed decisions and gaining insights. Large datasets significantly help in understanding data by uncovering intricate patterns and relationships within information, thereby playing a key role in improving machine learning models. They enhance accuracy, refine

predictions with diverse data, and fortify models' resilience, leveraging their capacity to reveal intricate patterns and relationships within information. Particularly in training complex models like those used in deep learning, large datasets are invaluable. Nevertheless, maintaining a balance between data quality and quantity is vital for reliable models and meaningful insights. However, acquiring extensive, high-quality data presents multifaceted challenges. Privacy regulations, exorbitant collection costs, limited data availability, accuracy concerns, ethical considerations, and the complexities of integrating diverse data sources pose significant hurdles.

Synthetic data emerges as a potential solution by mimicking statistical characteristics without divulging sensitive information. It addresses privacy concerns, supplements datasets, and broadens accessibility. Nevertheless, ensuring its accuracy compared to real-world data remains an important consideration in leveraging its potential.

Early approaches to generating synthetic data in the past were simplistic and lacked sophistication. Methods included simple random number generation, which failed to capture the underlying patterns or relationships within the data. Another approach involved duplicating existing data, resulting in a dataset without introducing any new information or reflecting the true variability. Replacing missing values with mean, median, or mode disregarded the nuanced relationships among variables. Basic mathematical models like linear equations or manual generation based on subjective assumptions lacked the ability to accurately replicate the complexities of real-world data. Sampling and interpolation methods attempted to imitate distributions, yet often failed to truly represent the underlying data dependencies. Overall, these approaches were limited in their capacity to replicate the intricacies and statistical properties of genuine datasets.

Modern techniques for synthetic data generation have shown remarkable advance-

ment by integrating sophisticated statistical methodologies with cutting-edge machine learning algorithms such as Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and transformer models. GANs employ a two-part system comprising a generator network that learns to produce synthetic data resembling the original dataset and a discriminator network trained to distinguish between real and synthetic data. This adversarial process drives GANs to generate highly realistic synthetic data while maintaining statistical properties. VAEs, on the other hand, leverage probabilistic modeling to encode and reconstruct data, allowing the generation of new samples that closely adhere to the original dataset's structure. Transformer models, known for their sequential learning capabilities, have also shown promise in generating synthetic data by learning intricate patterns and relationships from sequential or time-series data. Alongside these techniques, statistical methodologies such as copulas have played a pivotal role in capturing complex dependencies and multivariate structures in the data. Copulas offer a flexible framework for modeling and generating synthetic data, preserving individual variable distributions and accurately capturing their interdependencies. By integrating advanced machine learning models with robust statistical methods, hybrid approaches excel in creating synthetic data that mirrors the statistical properties, underlying structures, and intricate relationships of authentic datasets. However, there is no systematic study that compares the scalability of state-of-the-art methodologies for synthetic data generation (SDG) for patient service utilization data.

In this work, I address this gap by using five well known SDG models to generate synthetic data from a real dataset of patients accessing a regional health service system in Canada that serves many municipalities. In such a system, there is a clinical context coding scheme that labels service classes to number IDs. The data sets I analyze have the sequential ordering of the service classes a patient has visited;

I call these Patient Service Utilization data (PSU). From our data sets, I analyze four cohorts: *Schizophrenia Services (SS), Addiction Services–Post Withdrawal (AS), Homeless–Ever (HE), and Opioid Overdose (OO).*

Using a broad spectrum of statistical (FastML, Gaussian Copula), ML-based (CTGAN, Transformers), and hybrid models (CopulaGAN), I generate synthetic cohorts of patient data of four increasing sizes, 1x, 2x, 5x, and 10x. While doing so, I measure the *training time*, the time taken to train each model, and the *generation time*, the time to generate the synthetic data of different sizes. After the generation step, our study aims to answer two important questions about the quality of the synthetic data generated: (1) how closely does the synthetic data resemble the real data measured using *resemblance metrics*; (2) How useful is the synthetic data in practical scenarios measured using *utility metrics*. The contributions of this study are as follows:

1. I compare five SDG methodologies using two parameters, the time taken to train the model and the time taken to produce the synthetic data of various sizes, and show that the statistical models significantly outperform the ML-based and hybrid models.

2. I measure how closely the synthetic data resembles the real data using the Jensen-Shannon distance between marginal and joint distributions of important features in the real and synthetic datasets and show that they diverge only by a small amount (less than 3%).

3. By training a Binary RNN to differentiate between two cohorts, SS and AS, using synthetic data and then testing the trained classifier on real data, I demonstrate that the quality of synthetic data produced is excellent in terms of utility metrics. The classifier achieved high accuracy above 95%.

The three works in this dissertation are interconnected by their shared goal of ad-

vancing healthcare through data-driven machine-learning methodologies. The first work lays the foundation by improving diagnostic precision for thyroid disorders through comprehensive feature analysis, highlighting the critical role of data completeness in clinical decision-making. Building on this, the second study tackles the complex challenge of diagnosing Autism Spectrum Disorder (ASD) using fMRI data, showcasing the need for both accurate and explainable models in understanding intricate brain networks. The third work extends these principles to data synthesis, addressing privacy concerns and enabling scalable healthcare applications through the generation of realistic synthetic datasets. Together, these studies demonstrate a cohesive approach to enhancing healthcare diagnostics and data utilization while emphasizing interpretability, scalability, and ethical considerations.

Overall, the research in this dissertation aims to harness technological advancements to provide more accurate diagnoses, deeper insights into medical data, and practical applications for improving patient care in modern healthcare systems.

Chapter 2

Identifying Important Features for Clinical Diagnosis of Thyroid Disorder

Abnormal production of thyroid hormones in our body causes thyroid disorders such as hypothyroidism, hyperthyroidism, Hashimoto's disease, Graves' disease, and thyroid nodules. Undiagnosed thyroid disorders can affect the quality of life of an individual both physically and mentally. Thyroid disorders are common but sometimes become difficult to diagnose since the symptoms can be easily associated with other health conditions. Clinicians identify thyroid disorders by measuring the levels of thyroid hormones in our bloodstream. This work aims to help clinicians by carefully investigating if thyroid diagnosis improves when all important features (a complete thyroid panel) are measured as opposed to a select few. The reason is that the endocrine system is a complex network and having complete information about various hormones improves diagnosis and eventually treatment.

Much of previous work has focused on the performance of classifiers, supervised

and unsupervised, for the prediction of this disorder. Departing from this tradition, I focus on the concept of feature importance and its clinical implications. I identify the top four important features that predict the presence of thyroid disorders and show that these can be measured by clinicians cost-effectively. I also identify the pitfalls of current clinical practice of not checking the entire thyroid panel, prevalent in many countries with universal health care. Finally, I show that our results are quite robust and are unlikely to change with the choice of classifier or due to the inherent nature of a dataset in hand like imbalance.

This chapter starts with an overview of the literature on classification methods for the diagnosis of thyroid disorders. Next, I introduce the dataset I use and the proposed supervised and unsupervised methods for feature importance. I then outline the three research questions and summarize our experimental results.

2.1 Related Work

There is extensive literature analyzing the performance of various classification methods, supervised and unsupervised, for the diagnosis of thyroid dysfunction.

Early research focused on the use of neural networks for this task. Sharpe et al. [81] initiated the use of ANN for the diagnosis of thyroid disorder. Zhang et al. [97] showed that neural networks are more robust to sampling variation compared to traditional Bayesian classifiers. Hoshi et al. [41] analyzed the performance of two neural networks: self organizing maps and Bayesian regularized neural networks in the study of thyroid function and showed both to be useful. Temurtas [83] compared the performance of three different types of neural networks, multilayer (MLNN), probabilistic (PNN), and learning vector quantization (LVQ-NN) for thyroid disease diagnosis with PNN giving the best result. Saiti et al. [76] investigated the performance of genetic algorithms,

based on combining SVM with PNN, for this task.

Liu et al. [56] built a classifier based on fuzzy K-nearest neighbour (KNN) with a strong performance. Chen et al. [18] explore the performance of a hybrid system based on support vector machines for thyroid disease diagnosis. Li et al. [54] designed a computer-aided diagnosis system based on principle component analysis (PCA) and extreme learning machine (ELM) to assist in the task of thyroid disease diagnosis. In recent years, [70, 88, 74, 62] has focused on analyzing the effectiveness of simple decision tree algorithms for this prediction task. Please see [61] for a detailed survey of all the research on the prediction of thyroid disease.

2.2 Dataset and Methods

2.2.1 Thyroid Dataset Description

This work uses the thyroid disease dataset available in the KEEL repository [1]. KEEL cites as its source one of the databases available in the UCI repository [2]. The dataset in the UCI repository was contributed by J. R. Quinlan. The dataset, obtained from Daimler-Benz, contains 7200 instances and 21 features, of which 6 are continuous and 15 are binary datatypes, and has no missing data. I note that the class label definitions are different in KEEL [1] and UCI [2] thyroid dataset. Table 2.1 summarizes class label definitions as reported in KEEL identifying **hypothyroid (93%) as majority class**. Table 2.2 summarizes the UCI class label definitions that report **normal (93%) as the majority class.**

Table 2.1: Class Label Definitions - KEEL Repository

Class Label	Description	No. of Records	%
1	Normal	166	2%
2	Hyperthyroid	368	5%
3	Hypothyroid	6666	93%

Among prior research works, KEEL [1] is used by [38, 17] and UCI [2] is used by [5, 78, 79, 37]. The KEEL [1] and UCI [2] thyroid datasets having different class variable definitions pose a potential challenge regarding which one to use. I overcome this challenge by preprocessing and analyzing both the KEEL [1] and UCI [2] datasets carefully to help obtain more insights about the thyroid data in hand to aid with our decision making. See Section 2.2.3 below.

Table 2.2: Class Label Definitions - UCI Repository

Class Label	Description	No. of Records	%
1	Hyperthyroid	166	2%
2	Hypothyroid	368	5%
3	Normal	6666	93%

Table 2.3 elaborates on the acronyms for the most relevant variables used in KEEL [1] and UCI [2] thyroid dataset.

Table 2.3: Thyroid Dataset Abbreviations

S.No	**Acronyms**	**Description**
1	TSH	Thyroid Stimulating Hormone
2	FTI	Free T4 Index (relates to FT4)
3	TT4	Total Thyroxine (relates to FT4)
4	T3	Triiodothyronine (relates to FT3)

2.2.2 Data Preprocessing

The primary objective of our work is to establish the presence or absence of a thyroid disorder while identifying important features that support clinical diagnosis. Keeping our objective in mind and the ambiguity in class label description found in previous literature as described earlier in Sec. 2.2.1, I decided to transform KEEL [1] and UCI [2] datasets to a binary classification problem. This is done by combining class labels hyperthyroid and hypothyroid into a new class label "1-Thyroid" indicating the

presence of "thyroid disease" and Class label "0-Normal" identifies normal samples without "thyroid disease". Dataset descriptions with the new binary class labels are outlined in Tables 2.4 and 2.5.

Table 2.4: KEEL Repository - Binary Classifications

Class Label	Description	No. of Records	%
0	Normal	166	2%
1	Thyroid	7034	98%

Table 2.5: UCI Repository - Binary Classifications

Class Label	Description	No. of Records	%
0	Normal	6666	98%
1	Thyroid	534	2%

This decision not only left KEEL [1] and UCI [2] datasets heavily imbalanced as noticed in Table 2.4 and Table 2.5 but also raised another difficult question as to which choice of class label descriptions is convincing for us to utilize. This challenge is overcome by analyzing KEEL [1] and UCI [2] thyroid dataset. Among many observations, insights provided by Fig. 2.1(e) in Sec. 2.2.3 helped us choose the dataset to use for this work. Fig. 2.1(e) reports a very high number of patients getting treated with various thyroid treatments that are correctly classified as "1-Thyroid" by KEEL [1] making it the preferred choice for this work.

Table 2.6: Treatment Options - UCI Repository

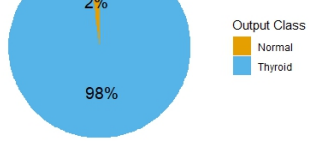
Class	Thyroxine	I131	Surgery	Antithyroid	Lithium
0 - Normal	923	109	97	99	85
1- Thyroid	17	12	4	4	6

On the other hand, UCI [2] classifies these patients as "0-Normal" despite getting treated for thyroid disease. This causes confusion and raises the following question for which I could *not* find a convincing answer - why are numerous patients classified

as "0-Normal" by the UCI [2] dataset receiving thyroid treatment as reported in Table 2.6?

Output Class	
Normal	166
Thyroid	7,034

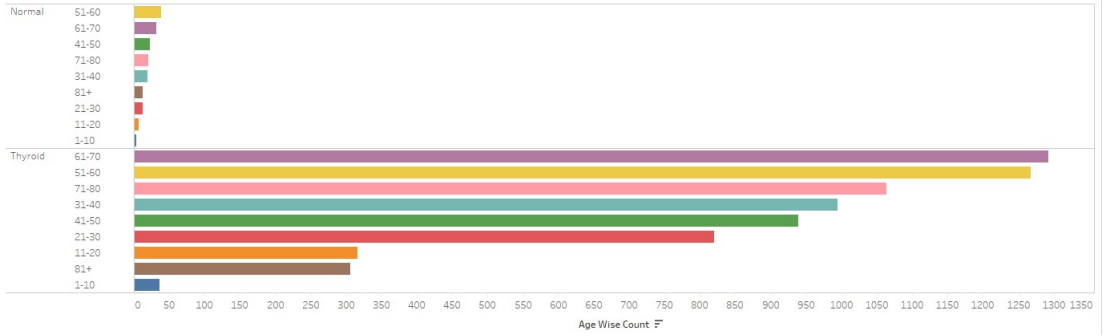
(a) Binary Class Conversion



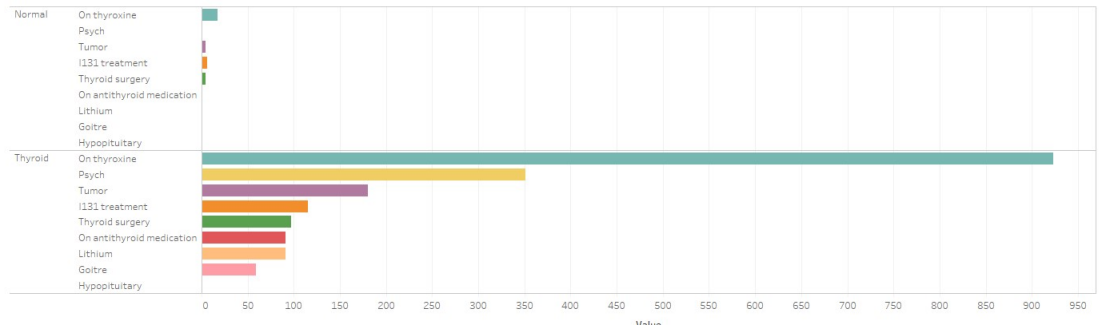
(b) Imbalance

Sex	Output Class: Normal	Output Class: Thyroid
F	132	4,877
M	34	2,157

(c) Distribution on Gender



(d) Age Group Affected with Thyroid Disease



(e) Symptoms and Various Treatment Options for Patients with Thyroid disease

Figure 2.1: Descriptive Statistics: KEEL Thyroid Dataset

2.2.3 Descriptive Statistics

Data analysis convinced us to use the binary version of KEEL [1] as defined in Table 2.4 for this work. This dataset is referred to as the KEEL dataset in the rest of

this chapter.

Fig 2.1 presents various observations of the KEEL dataset. Fig. 2.1(a) and 2.1(b) show the binary class distribution and severe imbalance with thyroid at 98% and normal at 2%. Fig. 2.1(c) clarifies that women are more susceptible to this disease than men. Fig. 2.1(d) exhibits that "thyroid disease" is a progressive disease that predominantly occurs between the ages of 30 and 70 and can even affect children less than 10 years.

Fig. 2.1(e) highlights the small number of outliers, those who are among the "0-Normal" patients but are treated using surgery or medications for the presence of thyroid disease. This project includes those outliers (a select few) without eliminating them since the samples identified as "0-Normal" are already low in the KEEL dataset. Fig 2.1(e) also highlights the various symptoms and treatments among patients with thyroid disease conveying thyroxine as the most common treatment and not all thyroid patients end up having a goiter making this symptom less common.

2.2.4 Feature importance

This work focuses on understanding feature importance in supervised and unsupervised settings and its clinical relevance.

Supervised Learning using CART & Random Forest

Supervised learning is a machine learning (ML) approach where a model gets trained to classify labeled datasets and later apply them to predict outcomes accurately [24]. Prior works [78, 79, 37] using [2] have established that decision tree classifiers achieve strong performance metrics. Furthermore, an added advantage of these classifiers is that they are highly interpretable. Therefore, Classification and Regression Trees (CART) is an excellent choice for the KEEL dataset and is our algorithm of choice

for supervised learning. The CART model built using the training data is represented as a binary tree in which the internal nodes are labeled by features and the leaves by the class variable. Given the tree representation of the CART model, the important features are easily identified as they label the internal nodes at the top levels of the tree.

Random Forest is another ML approach that is used to solve classification and regression problems. As opposed to CART which builds one decision tree, the Random forest algorithm is a collection of decision trees. The algorithm bases its prediction on the resulting "forest" of decision trees by taking the majority (or mean) of the output from various trees. Intuitively, a decision based on a collection of trees increases the accuracy of the outcome. CART and Random Forest have a built-in feature importance algorithm that uses Gini importance or mean decrease impurity.

Unsupervised Learning using Principle Component Analysis (PCA)

Unsupervised learning is an ML approach used to analyze unlabeled datasets and find patterns [24]. Analyzing the KEEL dataset by eliminating class labels not only helps with identifying the top four important features but also validates the results provided by the supervised approach. I chose this approach having found ambiguity in the literature about the class label definition as described earlier in Sec. 2.2.2.

PCA is an extensively used method for reducing the dimensionality of the feature set and is suited to work well on the continuous variables in a dataset. KEEL dataset has six continuous variables: Age, TSH, FTI, T3, TT4, and T4U. PCA internally reduces the dimensions of multivariate data to six principal components (PC), that can be visualized graphically, with minimal loss of information. PCA also reveals the data attributes contributing to each of these dimensions. The features contributing to principal dimension 1 (PC1) are considered important as it is the dimension with

the largest variance [89].

2.2.5 Metrics

In this work, I use several well known metrics to evaluate the performance of a classifier. Let TP, TN, FP and FN represent the True Positive, True Negative, False Positive and False Negative predictions of a chosen classifier. For example, false positives are cases that are actually negative but the model incorrectly labels as positive, or in our example, the model classifies a person as having a thyroid condition when they are actually normal.

Sensitivity is the fraction of actual positives that got predicted as a (true) positive. Sensitivity is also referred to as *Recall* or true positive rate. I have the following definition:

$$Sensitivity(or)Recall = \frac{TP}{TP + FN}$$

Specificity or true negative rate is defined as the fraction of actual negatives, which got predicted as a negative. That is,

$$Specificity = \frac{TN}{TN + FP}$$

Precision measures the quality of the positive predictions of the model.

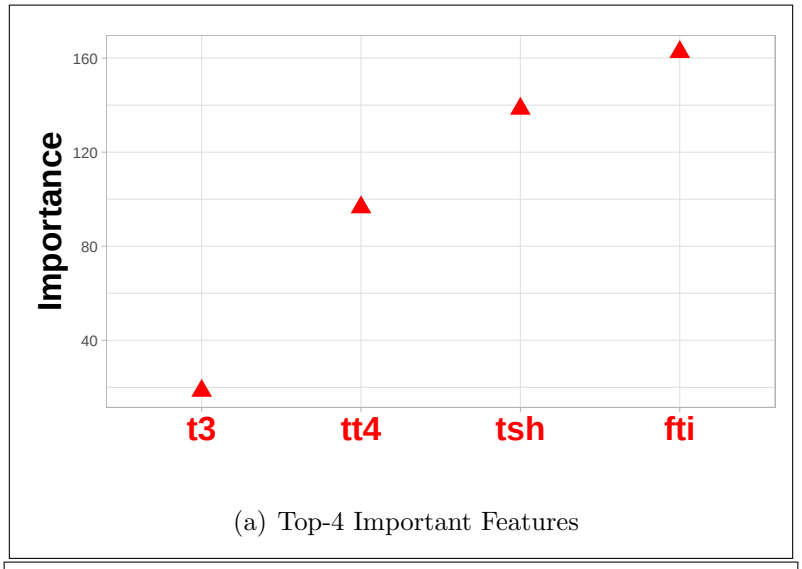
$$Precision = \frac{TP}{TP + FP}$$

Accuracy measures the fraction of correct predictions by the model.

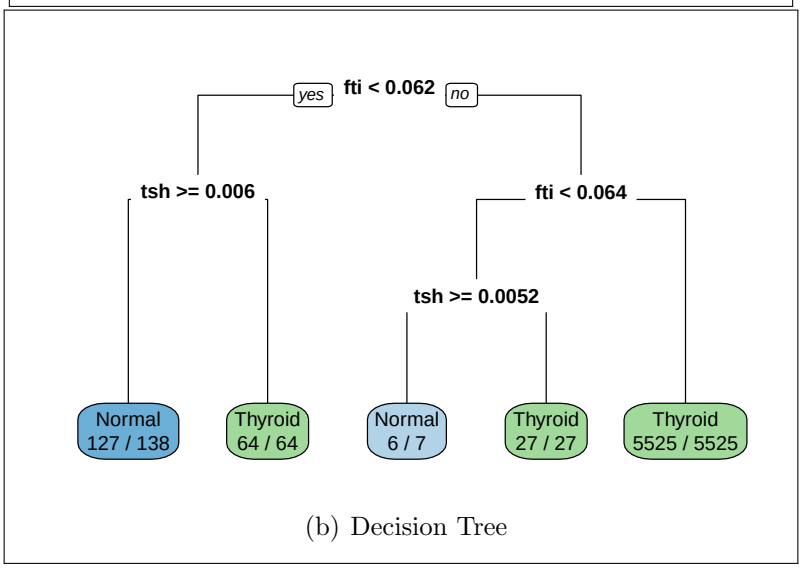
$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

When I want to find an optimal blend of precision and recall I can combine the two metrics using the *F1 score*. The *F1 score* is the harmonic mean of precision and recall taking both metrics into account as follows:

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$



(a) Top-4 Important Features

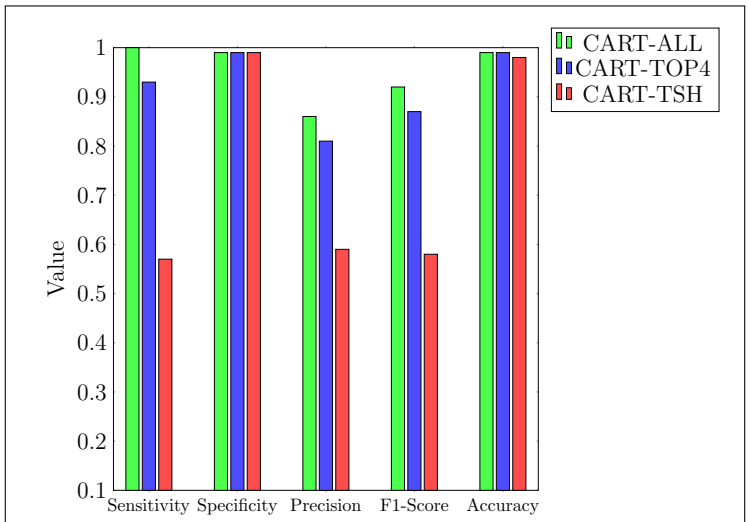


(b) Decision Tree

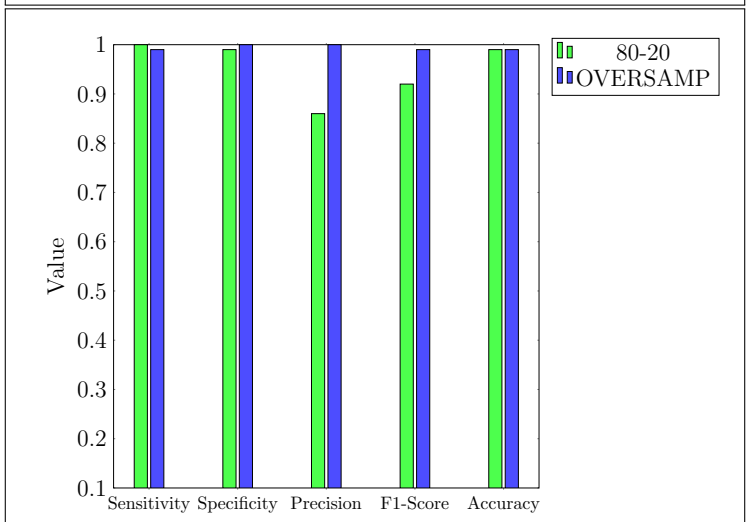
Figure 2.2: Supervised Learning: CART Results

2.3 Results and Discussion

This section summarizes our results and observations on the KEEL dataset obtained using supervised learning (CART and Random Forest), and unsupervised learning (PCA) as outlined in Sec. 2.2.4. I refer the reader to Table 2.3 for the description of terminology used in this section.



(a) Metrics - All(Green), Top-4(Blue) & TSH only(Red)



(b) Metrics - All(Green) & Oversampling(Blue)

Figure 2.3: Supervised Learning: CART Results

RQ1 : Among the 21 different features, what are the top-4 predictors for clinical thyroid disease diagnosis? Can these be measured cost effectively?

2.3.1 Supervised Learning: CART

This work uses all 21 data attributes of the KEEL dataset to build a predictive CART model. Initially, our approach is to build a successful predictive model for the imbalanced KEEL dataset by performing a simple 80:20 train/test split and evaluating performance metrics. I will later re-evaluate this model using oversampling and a more sophisticated model like Random Forest. The training dataset consisting of 5761 records is used to fit the chosen model (in our work, CART) enabling it to learn from this data. The test dataset, with 1439 records, is used to provide an unbiased evaluation of a final model fit. The test dataset provides the gold standard used to evaluate the model and is only used once a model is completely trained.

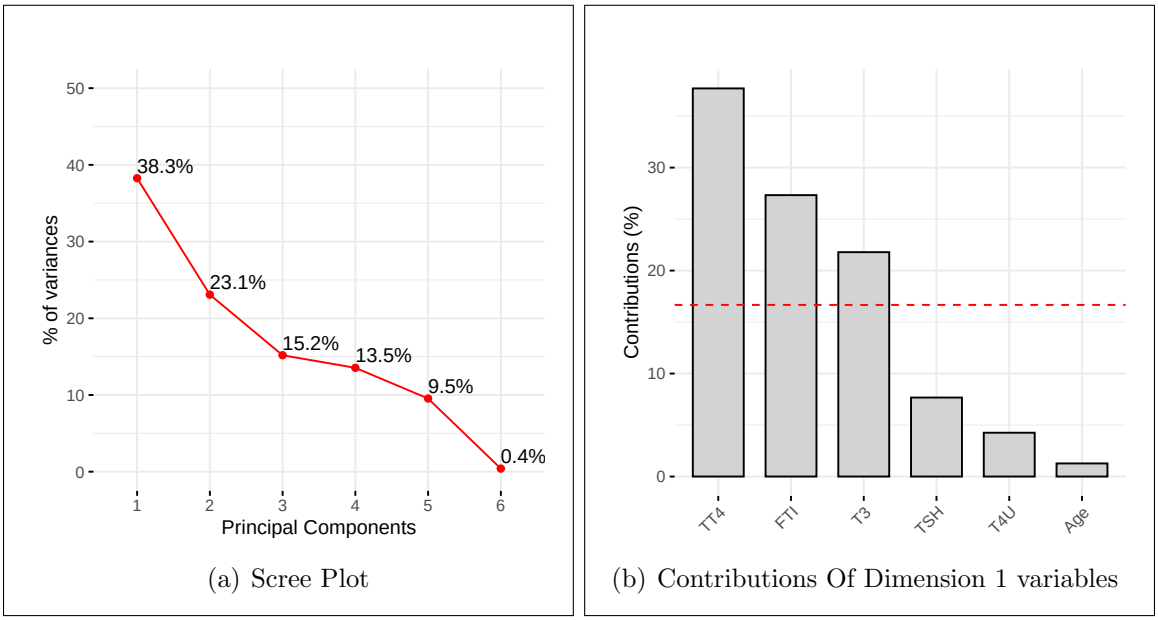
Visualization of results of the CART generated model in Fig. 2.2 and Fig. 2.3 shows the important features used in the classification decision tree and are summarized below.

1. Fig. 2.2(a) identifies FTI as the most important predictor for the risk of a thyroid disease based on the Gini index closely followed by TSH, TT4, and T3 in that order. These top-4 important features can be measured cost effectively by a simple blood test.

2. Resulting CART decision tree, Fig. 2.2(b), shows FTI as the root node of the tree followed by TSH as the second level node. Note that the printed tree does not show the other two important features, TT4 and T3, due to the pruning that the CART algorithm applies to the tree. Nonetheless, these two attributes are indeed deemed important by CART during the tree construction.

2.3.2 Unsupervised Learning: PCA

Results obtained using PCA are summarized in Fig. 2.4. PCA uses the six continuous data attributes, FTI, TT4, T3, T4U, TSH, and Age from the KEEL dataset. Therefore, the Scree Plot shown in Fig. 2.4(a) identifies six principal components (PC) and plots the eigenvalues, ordered from largest to smallest. Fig. 2.4(b) displays the list of important data attributes that contribute to principal dimension 1 (PC1) (dimension with the largest eigenvalue and maximum variance) and their % contribution [89].



(a) Scree Plot

(b) Contributions Of Dimension 1 variables

Figure 2.4: Unsupervised Learning: PCA Results

PCA results reported in Fig. 2.4(b) align well with RQ1 results in the CART model (Fig. 2.2) by identifying the very same features, FTI, TSH, TT4, and T3 as the top-4 important features for predicting thyroid disease. The red dotted line in Fig. 2.4(b) indicates the expected average contribution. The variables above the dotted line contribute more than the average and the ones below contribute lower than the average. This also demonstrates that the class label assumptions I made in Sec. 2.2.2 do not intervene with identifying the top-4 important features.

RQ2: How effective is TSH alone in predicting the presence of thyroid disease? That is, does the prediction improve by adding other three important features to the testing palette?

To understand the significance of the top-4 important features identified by CART in Fig. 2.2(a), I further study the corresponding performance metrics when I consider:

- All 21 attributes
- Only the top-4 important features
- Only TSH

The metrics obtained are summarized in Fig. 2.3(a) using green, blue, and red bars respectively. It shows that there is no noticeable loss of performance when using only top-4 attributes (the blue bars) as opposed to all 21 attributes (the green bars).

However, there is a significant loss in performance metrics like sensitivity, precision and F1 score using TSH only (the red bars) when compared with other two options. For example, the sensitivity falls to 0.57 while it is 0.91 when top-4 important attributes are included. Similar reduction is observed for precision and F1-score as well.

This result is of clinical importance as it shows that measuring TSH alone in the blood test is not reliable and can result in misdiagnosis of thyroid disease. I note that the practice of measuring TSH alone is prevalent in many countries that have universal health care such as Canada [13].

RQ3: How much does the imbalance in the KEEL dataset influence the results for RQ1 and RQ2? More generally, how robust are our answers to RQ1 and RQ2?

I explore two different approaches to answer RQ3.

- **Over Sampling:** To address the imbalance in the KEEL dataset, I use oversampling that works with minority class by replicating the observations. Fig 2.3(b)

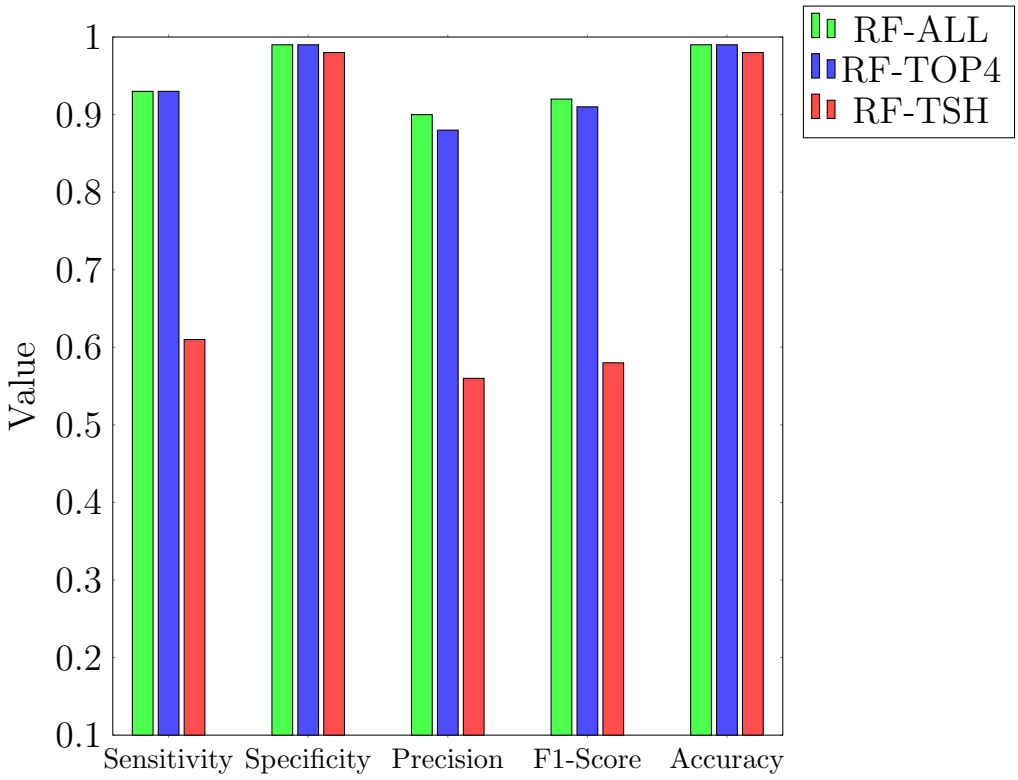


Figure 2.5: Random Forest Metrics: All (Green), Top-4 (Blue) & TSH only (Red)

shows that all the performance metrics improve after the use of oversampling. However, for the purpose of this work, the crucial observation is that oversampling also reports the top-4 important features as FTI, TSH, TT4, and T3 which are the same as Fig. 2.2(a) showing that balancing the dataset does not affect our result for RQ1.

- **Random Forest:** I also test the robustness of our conclusions by revisiting RQ1 and RQ2 using a more sophisticated classifier, Random Forest. I observed that the top-4 important features are TSH, FTI, T3, and TT4 in that order. That is, the top-4 important features remain the same as reported for CART in Fig. 2.2(a) though their relative order has changed. In addition, I summarize the performance of Random Forest on the KEEL dataset using the three scenarios described in RQ2 in Fig. 2.5. I again notice that using the top-4 im-

portant features gives almost the same performance as using all the variables. However using only TSH causes a significant drop in performance for sensitivity, precision, and F1. For example, the sensitivity drops from 0.93 to 0.61 and the F1-score drops from 0.91 to 0.58.

Publication

The results in this chapter are based on the following publication.

- Sowmya Balasubramanian, Venkatesh Srinivasan, and Alex Thomo. Identifying Important Features for Clinical Diagnosis of Thyroid Disorder. *Proceedings of the IEEE/ACM International Symposium on Network Enabled Health Informatics, Biomedicine and Bioinformatics (HI-BI-BI)*, 2022, pages 363–369.

Chapter 3

Brain network similarity using k-cores

Autism Spectrum Disorder (ASD) is extensively studied by medical practitioners, health researchers, and educators. ASD symptoms appear in early childhood, within the first two years of life, but diagnosing it remains challenging due to its complex and diverse nature. Nevertheless, early diagnosis is crucial for effective intervention. Traditional methods rely on behavioral observations, while modern approaches involve applying machine learning (ML) to brain networks derived from fMRI scans. The limited explainability of these advanced techniques poses a significant challenge in gaining clinician's trust.

This chapter builds on recent works that design explainable approaches for ASD diagnosis from fMRI data preprocessed as graphs. Our research makes three key contributions. Firstly, I demonstrate that a simple approach based on viewing graphs as tables and using tabular data classifiers can achieve the same performance as state-of-the-art, explainable graph theoretic methods. Secondly, I provide evidence that adding higher-order connectivity information as attributes does not improve their

performance. Most importantly, I show why the classification of brain networks is challenging by demonstrating the similarity between graphs belonging to individuals with ASD and those without, using a novel k-core based approach.

This chapter starts with an overview of the literature on classification methods for the diagnosis of ASD using fMRI data. Next, I introduce the dataset I use and the proposed methods for the classification and comparison of brain networks. I then outline the three research questions and summarize our experimental results for each of them.

3.1 Related Work

Over the years, there has been extensive research on the classification of ASD [77, 7, 93, 59, 68, 48] using fMRI data. Several studies have explored diverse approaches to address this problem.

The traditional approach involves utilizing behavioral and family history information for ASD diagnosis. Misman *et al.* [63] have claimed impressive accuracy rates of up to 99% by employing Deep Neural Networks (DNNs) on ASD datasets that incorporate comprehensive behavioral and family history data. In an effort to improve the accessibility of these diagnosis techniques, Abbas *et al.* [3] developed mobile applications that coupled with machine learning techniques, show potential in aiding ASD diagnosis. However, it is important to note that relying solely on behavioral information may not provide an early and accurate diagnosis, as behavioral symptoms associated with ASD may not manifest until later in a child's development. Therefore, alternative methods focusing on biologically-based markers derived from fMRI scans are being explored.

Machine learning approaches have been widely employed in ASD classification

using fMRI data [57, 84]. Researchers have utilized correlation matrices and deep learning models to achieve accurate classifications. For example, Liu *et al.* [57] have used the Extra Trees algorithm to select relevant features from correlation matrices derived from fMRI data, resulting in an accuracy of 72% on the ABIDE dataset. Deep Learning models, such as Dense Neural Networks (DNNs), have also shown promise in achieving a high accuracy of 88%, often surpassing classical machine learning models [82]. Feature selection techniques, such as sparse auto-encoders, have been employed to enhance classification performance and obtain accuracies above 90% [50].

However, it is important to consider the limitations of studies conducted on small datasets, as they may overfit the models and limit their generalizability to new datasets. Additionally, there is ongoing research to strike a balance between model performance and interpretability, as deep learning models are often considered "black-box" classifiers. Efforts are being made to develop explainable classification methods [95, 36, 55], allowing researchers and neuroscientists to gain insights and trust the predictions made by these models. These methods, such as [71], often involve deriving explanations for the model's decisions, such as SHAP values.

The present study is inspired by the work of Lanciano *et al.* [51]. They prioritized interpretable and simple features to aid neuroscientists' understanding, rather than solely aiming for high accuracy. In a similar vein, Coupette *et al.* [19] developed an algorithm to identify characteristic subgraphs with common and contrasting structures in graph groups, and illustrated their technique using brain networks from adolescents in the ABIDE dataset. Finally, Enns *et al.* [27] proposed the discriminative edges method with the goal of identifying a set of important edges that can separate the two classes. All these studies aim to uncover meaningful patterns in brain networks to enhance our understanding of ASD.

3.2 Datasets and Methods

In this section, I will first describe the datasets I use and the preprocessing steps involved in generating brain networks. Then, I will outline the methodologies employed for the classification and comparison of these brain networks.

3.2.1 Dataset Description

ASD Dataset

The study described in Section 3.3.1 utilizes the ASD dataset obtained from Lanciano *et al.* [51] (https://github.com/tlancian/contrast-subgraph), which was originally released by the Autism Brain Imaging Data Exchange (ABIDE) project [22]. The dataset consists of neuroimaging data from 1112 individuals, comprising 573 Typically Developed (TD) individuals and 539 individuals diagnosed with Autism Spectrum Disorder (ASD). Typically Developed (TD) individuals have normal brain function without neurological disorders whereas Autism Spectrum Disorder (ASD) individuals face autism-related challenges. Each individual in the dataset is represented by an undirected unweighted graph containing 116 vertices, where each vertex corresponds to a Region of Interest (ROI). The presence of an edge in the graph indicates strong a correlation in the activity between the two ROIs. The graphs are represented by an adjacency matrix of size 116×116.

Lanciano *et al.* created four distinct datasets from the original ABIDE source [22]. These datasets were curated by selecting individuals based on shared characteristics, such as age, gender, and scan conditions (e.g., eyes closed or male), as shown in Table 3.1. Each of the datasets is divided into two classes namely TD and ASD and the dataset's description reflects shared phenotypic features among the observations. For instance, the "Children" dataset comprises individuals aged 9 years or younger,

the "Adolescents" dataset includes individuals aged between 15 and 20 years, the "EyesClosed" dataset consists of individuals who underwent fMRI scans with their eyes closed, and the "Male" dataset exclusively includes male individuals.

Dataset	Description	TD	ASD
Children	Age ≤ 9	52	49
Adolescents	Age in [15,20]	121	116
EyesClosed	Eyes closed during scanning	158	136
Male	Male individuals	418	420

Table 3.1: ASD Dataset - Lanciano *et al.*

ADHD Dataset

In our experiments, as described in Section 3.3.2, I also used an ADHD dataset, which is another neurodevelopmental disorder impacting individuals of various age groups. ADHD is characterized by attention difficulties and impulsivity. Like ASD, the exact causes of ADHD remain unclear, and there is presently no cure for either condition. Nevertheless, treatments such as behavioral therapy and medication aid in symptom management and improving daily functioning [66]. By incorporating the ADHD dataset into our experiments, I gain valuable insights into the applicability of our techniques and results for ASD to other related disorders like ADHD.

Dataset	Description	TD	ADHD
ALL	ALL	330	190

Table 3.2: ADHD Dataset - Abrate *et al.*

This study uses the ADHD dataset from [4] (https://github.com/carlo-abrate/CounterfactualGraphs) which includes 330 individuals with typical development (TD) (normal brain function) and 190 individuals diagnosed with ADHD, as summarized in Table 3.2. Similarly to the ASD dataset in Section 3.2.1, the ADHD dataset also portrays each individual with an undirected unweighted graph of 190 vertices,

representing regions of interest (ROIs). The presence of an edge in the graph signifies a substantial correlation in the activity between the two ROIs, resulting in an adjacency matrix of size of 190×190.

For the parcellation of the brain, [51] used the AAL atlas for the ASD dataset ($|V| = 116$) and the authors of [4] used Craddock 200 (CC200) for the ADHD dataset $|V| = 190$).

3.2.2 Data Preprocessing

In my work, I use the ASD and ADHD graph datasets provided by [51, 4], without requiring any additional preprocessing. However, for completeness, I outline the preprocessing steps needed to convert fMRI scans into graphs.

Resting-state fMRI is a neuroimaging technique that works by measuring the blood-oxygen-level-dependent (BOLD) signals in the brain. When a region of the brain becomes active, there is an increased demand for oxygenated blood to support the active neurons in that region. Therefore, the body responds by increasing the flow of oxygenated blood to that region. fMRI takes advantage of this body response to measure brain activity indirectly via BOLD signal intensities in different regions. The choice of the size of each region, and hence the number of such regions, is done using a brain atlas (AAL atlas, CC200). These regions are referred to as regions of interest (ROI). In summary, the output of an fMRI scan is a 3-dimensional image of the BOLD signal intensities in different ROIs of the brain measured over time. After obtaining the BOLD time series for each Region of Interest (ROI), the process of transforming it into graph data involves three essential steps:

1. **Analyze Patterns.** The communication pattern between different brain regions is examined by comparing their BOLD time series. The underlying premise is that the level of functional connections between two regions can be

determined by assessing the correlation in their BOLD time series. The higher the correlation, the higher the functional connectedness.

2. **Calculate Pearson correlation coefficients (PCC).** Pairwise PCC is calculated between the BOLD time series for every pair of ROIs. This step yields a correlation matrix of size 116 × 116 (for ASD) or 190 × 190 (for ADHD), containing values in the range [-1, +1]. The correlation matrix acts as a weighted adjacency matrix, with ROIs as nodes and correlation coefficients as edge weights.

3. **Apply threshold.** Thresholding retains only the strongest connections, creating an undirected, unweighted graph like the ASD and ADHD datasets.

3.2.3 Classification Methods of Brain Networks

Graphs as tables. Recall that brain networks are simple, undirected graphs on 116 vertices in which each vertex has a unique id between 1 and 116. In order to convert a collection of such vertex-labeled brain networks into a table, I create a table with $\binom{116}{2} = 6670$ columns so that the table has one column for each possible edge in the graph. I can then represent any brain network G as a binary vector, $T(G)$, of length 6670 such that a bit location labeled (i, j) stores a 1 if the edge (i, j) is present in G and 0 otherwise. I assume that the edges are listed in the lexicographic order.

Example. For the graph G shown in Fig. 3.1(a), the binary vector corresponding to G, $T(G)$, is [101001110110101].

Tabular Classifiers. Transforming graph data into tables enables organized and structured analysis. Tables provide a tabular representation that allows for easier data manipulation, sorting, filtering, and statistical analysis compared to the graphical representation of the graph data. In this study, I utilize various tabular classifiers,

including *SVM, Linear Regression, Random Forest, XGBoost, AdaBoost, and Perceptrons*. I evaluate performance using the four metrics: *Accuracy, Precision, Recall. and F1-score*. While all metrics are important, I present the top-3 classifiers selected based on their *accuracy* in Section 3.3. Our emphasis on accuracy aligns with previous works [51, 27].

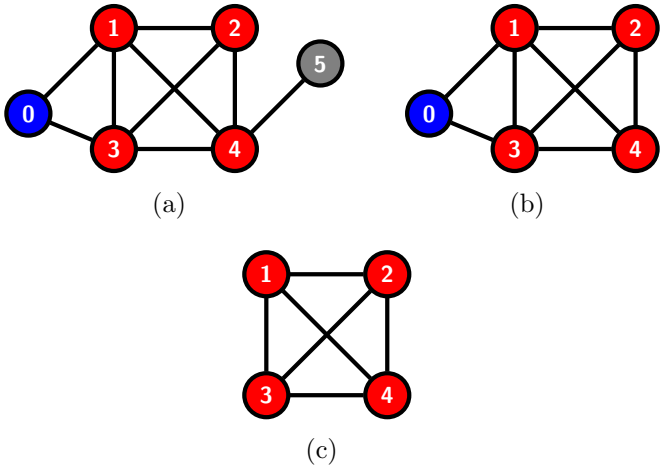


Figure 3.1: a) Graph $\mathcal{G}$, b) 2-core of $\mathcal{G}$, c) 3-core and Max-Core of $\mathcal{G}$.

Local Clustering Coefficient. Informally, the local clustering coefficient of a node i, C_i, in a graph measures the likelihood that the neighbours of i are also connected. Formally,

$$C_i = \frac{|\{(j,k) \mid j,k \in N_i, (j,k) \in E\}|}{\binom{|N_i|}{2}}$$

where N_i is the set of neighbours of node i and there is an edge between j and k. For each brain network, I will compute a tuple of size 116 containing the local clustering coefficient of its vertices.

Example. For the graph $\mathcal{G}$ in Fig. 3.1(a), the local clustering coefficient of vertex 0 is $C_0 = \frac{1}{\binom{2}{2}} = 1$ while $C_1 = \frac{4}{\binom{4}{2}} = 4/6$.

The local clustering coefficient is a measure introduced by Watts and Strogatz to study small world theory in social networks. As it measures interconnectedness among neighbors by counting triangles (a small graph pattern) centered at a node

in essence, adding it as an additional attribute provides extra information about the graph and helps with ML tasks.

3.2.4 Comparison Methods of Brain Networks

To compare the collection of brain networks belonging to the two classes, ASD and TD, I use local clustering coefficient, Hamming distance, and k-core decomposition.

k-Core of a Graph. For a graph G and an integer k, $1 \leq k \leq n$, the k-core of G is the maximal subgraph H of G such that the induced degree of every vertex in H is at least k.

Note that, by the above definition, the $k+1$-core of G is a subset of the k-core of G, and hence the set of k-cores, $1 \leq k \leq n$, as k increases from 1 to n form a nested structure. This nested structure is referred to as k-core decomposition in the literature [47].

Example. Fig. 3.1(b) is the 2-core of the graph $\mathcal{G}$ in Fig. 3.1(a). In Fig. 3.1(b), each vertex is connected to at least 2 other vertices and it is also the maximal subgraph with that property. Note that 1-core of $\mathcal{G}$ is $\mathcal{G}$.

The study of k-cores offers valuable insights into the structure, resilience, and communities of complex networks, making it a popular topic in large-scale network analysis.

Max-Core of a Graph G. Let m, $1 \leq k \leq n$ be the integer such that the m-core of G is non-empty but its $m+1$-core is empty. If so, I refer to the m-core of G as its max-core.

Example. Fig. 3.1(c) is the 3-core of graph $\mathcal{G}$ in Fig. 3.1(a). In Fig. 3.1(c), each vertex is connected to at least 3 other vertices and it is also the maximal subgraph with that property such that this is the max-core as the 4-core of $\mathcal{G}$ is empty.

To compare max-cores of two different graphs G_1 and G_2, I use the notion of Jaccard similarity. Let V_1 and V_2 denote the set of vertices of G_1 and G_2.

Jaccard Similarity. Given two sets V_1 and V_2, I define their Jaccard similarity, $JS(V_1, V_2)$ as:

$$JS(V_1, V_2) = \frac{|V_1 \cap V_2|}{|V_1 \cup V_2|}$$

Example. Fig. 3.1(c) is the max-core G_1 of graph $\mathcal{G}$ and has vertices $V_1 = \{1, 2, 3, 4\}$. Lets assume G_2 is the max-core of another graph and has vertices $V_2 = \{1, 2, 7, 8, 9\}$, Jaccard similarity is computed as: $JS(V_1, V_2) = \frac{2}{7}$

Another notion that is useful to compare how close two graphs $\mathcal{G}$ and $\mathcal{G}'$ are based on their edges is the notion of Hamming distance.

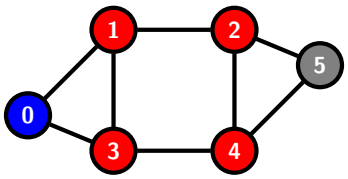


Figure 3.2: Graph $\mathcal{G}'$

Hamming Distance. Given two graphs $\mathcal{G}$ and $\mathcal{G}'$ on n vertices, the Hamming distance between $\mathcal{G}$ and $\mathcal{G}'$ is the minimum number of edge insertions and deletions needed to convert $\mathcal{G}$ to $\mathcal{G}'$. Equivalently, the Hamming distance between $\mathcal{G}$ and $\mathcal{G}'$ is the number of bit positions in which the two binary strings $T(\mathcal{G})$ and $T(\mathcal{G}')$ differ.

Example. Consider $\mathcal{G}$ and $\mathcal{G}'$ in Fig.3.1(a) and Fig.3.2. I need to add edges $(1, 4)$ and $(2, 3)$ and delete the edge $(2, 5)$ to convert $\mathcal{G}$ to $\mathcal{G}'$. Hence, the Hamming distance is 3.

3.3 Results and Discussion

I now present the results I obtained from analyzing the ASD and ADHD datasets. The code and charts for our results can be found at https://github.com/sowbalas/

HIBIBI2023.git.

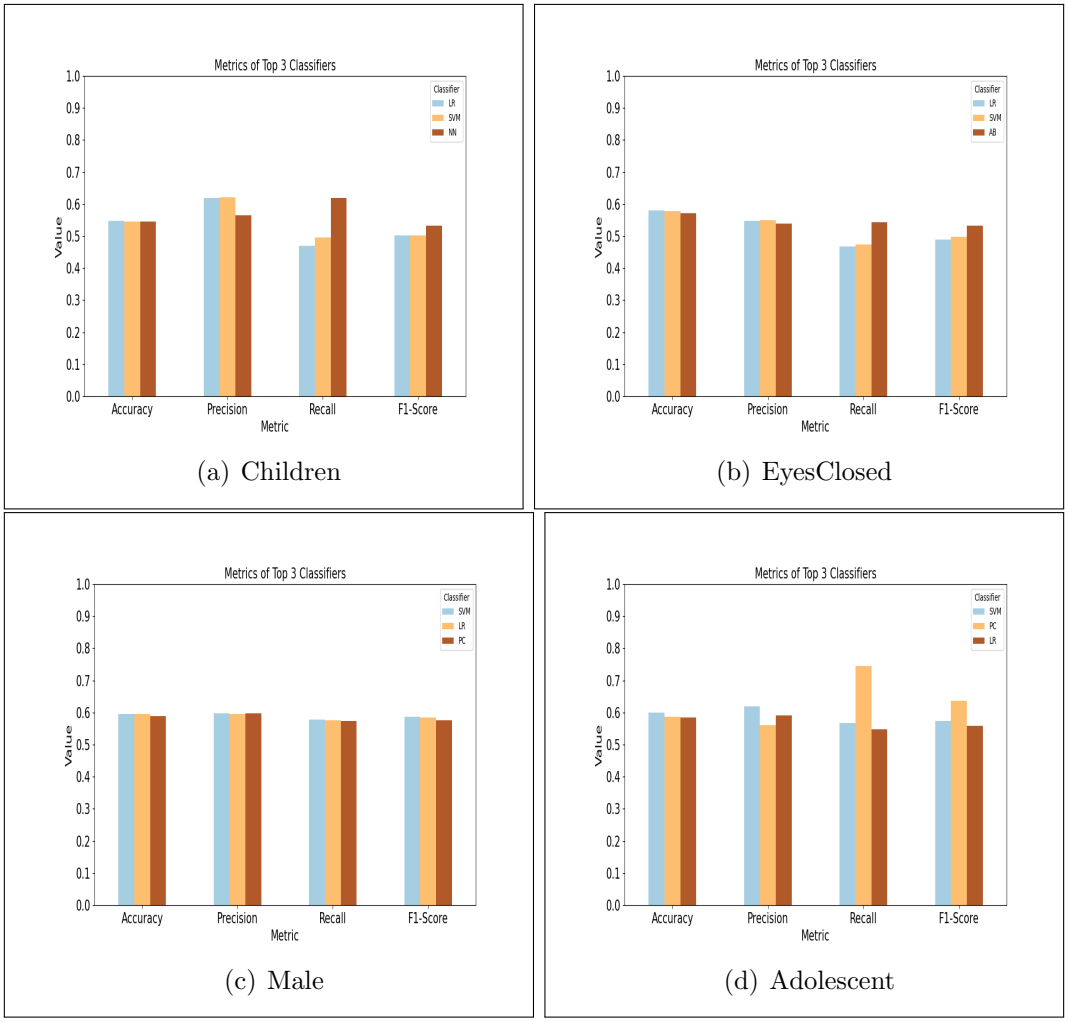


Figure 3.3: RQ1 - Performance Metrics of Top-3 Classfiers

3.3.1 Insights on ASD Dataset

The first research question of this study focuses on exploring the possibility of achieving explainability through a simple alternate pathway: the conversion of a graph into a table.

RQ1: How do well-known tabular classifiers perform on brain networks in tabular format?

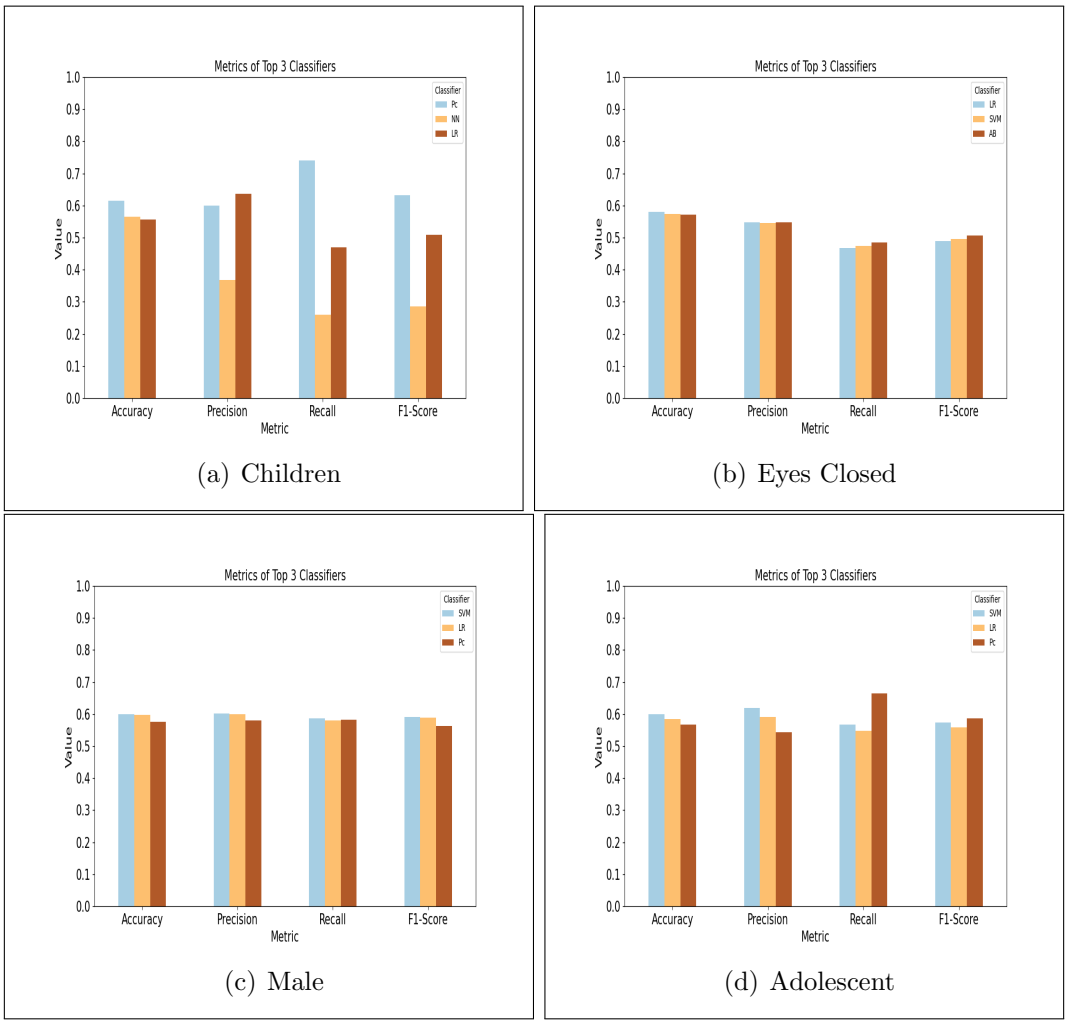


Figure 3.4: RQ2 - ASD - Metrics

By converting the graph dataset into tabular form, as described in Section 3.2, I can employ a wide array of well-known tabular classifiers for flattened brain networks. Figure 3.3 presents our results (using ten-fold cross-validation), showing SVM (with a linear kernel) and Linear Regression (LR) consistently ranking among the top-3 classifiers across all four ASD datasets. These classifiers achieve a mean accuracy of close to 60% on larger datasets, like Male. The balanced nature of the ASD datasets sets the baseline accuracy at 50%.

The strong performance of SVM and LR classifiers, especially on larger datasets, indicates their potential significance in ASD diagnosis. Interestingly, these classi-

fiers are highly explainable, and their accuracy closely matches that of sophisticated graph-theoretic methods from [27]. Enns *et al.* aimed to replicate Lanciano *et al.* work to comprehend the reported high accuracy. However, their results differed from the original study, demonstrating mean accuracies of 73.5% for Children, 60.8% for Adolescents, 58.5% for EyesClosed, and 59.3% for Male on the ASD dataset (Enns *et al.*, Table 4.2 in [27]). Our RQ1 results demonstrate a simple, alternate strategy to achieve explainability in ASD diagnosis using brain networks.

Our second question stems from the knowledge that graph classifiers can benefit from additional attributes beyond node and edge information as shown in [73].

RQ2: Can incorporating higher-order connectivity patterns, such as triangles, as attributes improve the performance of tabular classifiers?

To address this question, I created an augmented table by adding 116 new attributes, namely local clustering coefficients, to the table used for RQ1. However, the performance metrics (using ten-fold cross-validation) did not significantly improve. The top three classifiers on the ASD male dataset achieved a mean accuracy of 60%, as shown in Table 3.3.

Accuracy	**SVM**	**LR**	**NN**
RQ2-Augmented table	0.60	0.60	0.60
RQ2-Clustering Coeff. only	0.55	0.55	0.54

Table 3.3: RQ2 Top-3 Classifiers : ASD Male Dataset

In a related experiment, I further explored the classifiers' performance when provided solely with higher order connectivity patterns. For this purpose, I created a tabular dataset with each row representing a brain network and 116 columns containing the local clustering coefficients of the nodes within the brain network. Table 3.3 displays the top three classifier results in this scenario. SVM and LR, the most successful classifiers, achieved a lower mean accuracy of 55%.

To summarize, our findings for RQ1 and RQ2 reveal that tabular classifiers achieve a mean accuracy of around 60%. Despite attempting to enhance performance by incorporating higher-order information, such as local clustering coefficients, I did not observe notable improvements. This leads us to question whether there is a fundamental reason underlying this phenomenon. I further explore this inquiry through the concept of similarity measures.

RQ3: Can I provide evidence showing that the two classes of networks (ASD and TD) are quite similar?

I explore the presence of similarities between ASD and TD networks, through Hamming distance and Jaccard similarity of k-cores. The first approach, Hamming distance, focuses on edge-based similarity, while the second approach, Jaccard Similarity, centers on vertex-based similarity [45, 67].

Similarity based on Hamming distance.

Our first approach examines the similarity between the two categories of brain networks (TD and ASD) using the Hamming distance metric. I consider datasets from RQ1, where each brain network is represented as a binary string of length 6670. Each bit in the string represents a possible edge, with its value indicating the presence or absence of that edge. The Hamming distance between two brain networks is the minimum number of edge flips required to transform one network into the other, as defined in Section 3.2.4.

For each of the four datasets, I do the following (See Algorithm 1): For each brain network G_i in the dataset D containing ASD and TD files, I compute its Hamming distance to every other brain network G_j, $i \neq j$, in D. Using this information, I identify the brain network G_k that is closest to G_i in terms of Hamming distance (Lines 3 to 5 of Algorithm 1). That is, the brain network G_k requires the fewest number of edge additions and deletions to convert to G_i. G_i is a *good file* if G_k is in

the same class as G_i and *bad file* if G_k is not in the same class as G_i (Lines 6 to 10).

Algorithm 1 Similarity based on Hamming distance

1: **Input**: A Dataset $D = \{G_1, G_2, \ldots, G_m\}$ consisting of two classes, ASD and TD files
2: **Output**: f_{ASD} and f_{TD} (the fraction of good ASD and TD files based on Hamming distance)
3: for each $G_i \in D$ do
4: $class \leftarrow class(G_i)$
5: $k \leftarrow \arg\min_{i \neq j} HD(G_i, G_j)$
6: **if** G_k is in the same class as G_i **then**
7: $count_{class}$++; $good_{class}$++
8: **else**
9: $count_{class}$++
10: **end if**
11: **return** $good_{ASD}/count_{ASD}$ and $good_{TD}/count_{TD}$

As shown in Figure 3.5, in the Adolescent dataset, I observed that both the ASD and TD classes have approximately 40% of good files (in green), indicating that around 60% are bad files (in red). This finding is significant since it demonstrates that for the majority of brain networks, the most similar network belongs to the opposite class, not its own. Similar results across the other three datasets reinforce the conclusion that the widely used similarity measure fails to effectively distinguish between the two classes.

Jaccard Similarity based on k-cores.

Our approach involves employing the k-core as a "glocal" similarity measure, which combines aspects of both local and global metrics. This approach overcomes limitations found in traditional local (e.g., Hamming distance) and global (e.g., random walk-based) measures. Notably, prior research has recognized the importance of such glocal similarity measures, as discussed in [45, 67].

More specifically, I use the max-core of a brain network, as described in Section 3.2.4. The primary objective is to assess whether the max-core of a given brain

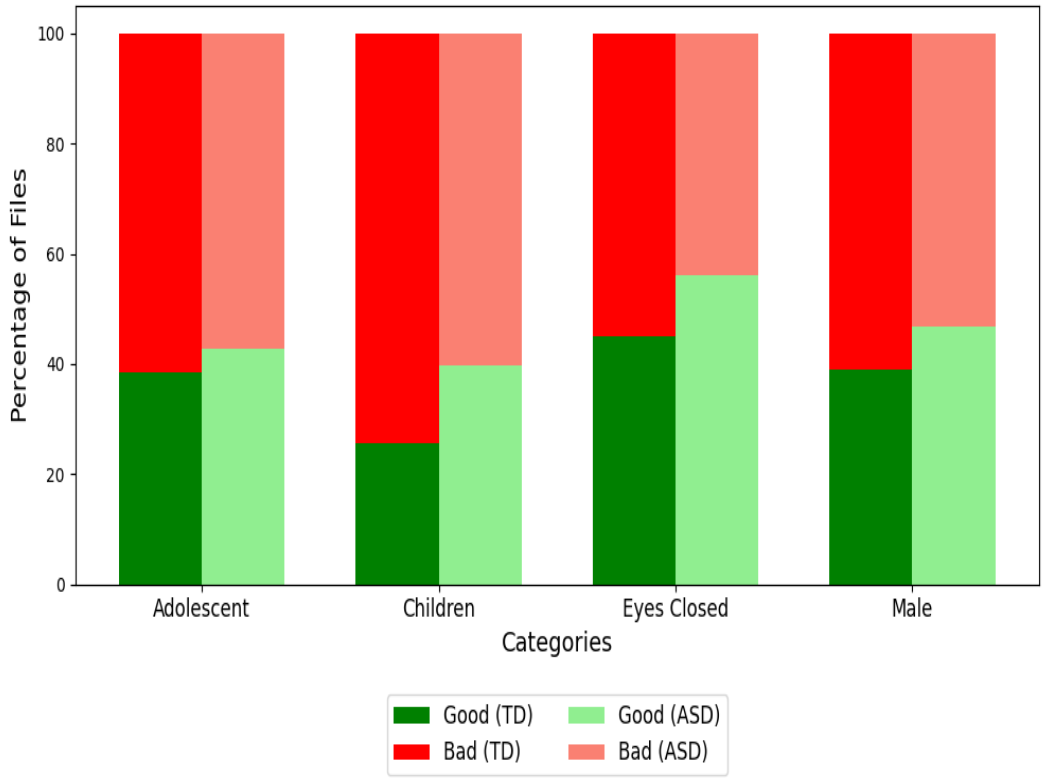


Figure 3.5: ASD Dataset: % of good and bad files using Hamming Distance

network resembles that of a typical ASD network or a TD network, using the Jaccard similarity metric. The max-core of a brain network comprises a set of ROIs where each ROI's time series exhibits strong correlations with at least k other ROIs. However, it's worth noting that computing the max-core is computationally intensive compared to the Hamming distance metric. To address this computational challenge, I have devised a more efficient procedure, drawing inspiration from ideas presented in [51] (see Algorithm 3).

1. Given a dataset, I partition the ASD files in that dataset into two sets, S_{ASD} and T_{ASD} using an 80:20 split. Similarly, I partition the TD files into two sets, S_{TD} and T_{TD} using an 80:20 split (Line 3 of Algorithm 3).

2. Using the files in S_{ASD}, I create a single graph, SG_{ASD}, that I call the ASD

Algorithm 2 Jaccard similarity of Max-Core

1: **Input**: A Dataset $D = \{G_1, G_2, \ldots, G_m\}$ consisting of two classes, ASD and TD files
2: **Output**: f_{ASD} and f_{TD} (the fraction of good ASD and TD files based on Jaccard similarity of max-core)
3: Partition D into four sets S_{ASD}, T_{ASD}, S_{TD}, and T_{TD} using 80:20 split.
4: Compute SG_{ASD} and SG_{TD} using 75% threshold.
5: Compute their max-cores, MC_{ASD} and MC_{TD}
6: for each $G_i \in T_{ASD}$ do
7: Compute max-core MC_i of G_i
8: **if** $JS(MC_i, MC_{ASD}) > JS(MC_i, MC_{TD})$ **then**
9: $count_{ASD}$++; $good_{ASD}$++
10: **else**
11: $count_{ASD}$++
12: **end if**
13: for each $G_i \in T_{TD}$ do
14: Compute max-core MC_i of G_i
15: **if** $JS(MC_i, MC_{TD}) > JS(MC_i, MC_{ASD})$ **then**
16: $count_{TD}$++; $good_{TD}$++
17: **else**
18: $count_{TD}$++
19: **end if**
20: **return** $good_{ASD}/count_{ASD}$ and $good_{TD}/count_{TD}$

summary graph. This graph is a graph on 116 vertices. It contains an edge (i, j) if and only if more than 75% of the graphs in S_{ASD} contain that edge. Similarly, I create the TD summary graph SG_{TD} (Line 4).

3. I compute the max cores of the two summary graphs, SG_{ASD} and SG_{TD} (Line 5).

4. Now, for each file in T_{ASD}, I compute its max-core and check if it is closer to the max-core of SG_{ASD} or the max-core of SG_{TD} using Jaccard similarity. I say that it is *good* if it is closer to the max-core of SG_{ASD} and *bad* otherwise. Compute the percentage of good ASD files (Lines 6-12).

5. Repeat the previous step for TD files (Lines 13-19).

Figure 3.6 illustrates the results of our approach, representing mean percentages over ten runs. Taking the Adolescent dataset as an example, I observed that both the ASD and TD classes have approximately 40% and 60% of good files (displayed in green), respectively. This implies that the percentage of bad files is approximately 60% and 40% for ASD and TD, respectively. The noteworthy aspect of this observation is that for nearly half of the brain networks, the most similar network based on max-core comes from the opposite class, not its own. I found similar results for the other three datasets as well. This finding suggests that the Jaccard similarity of max-cores fails to differentiate between the two classes effectively.

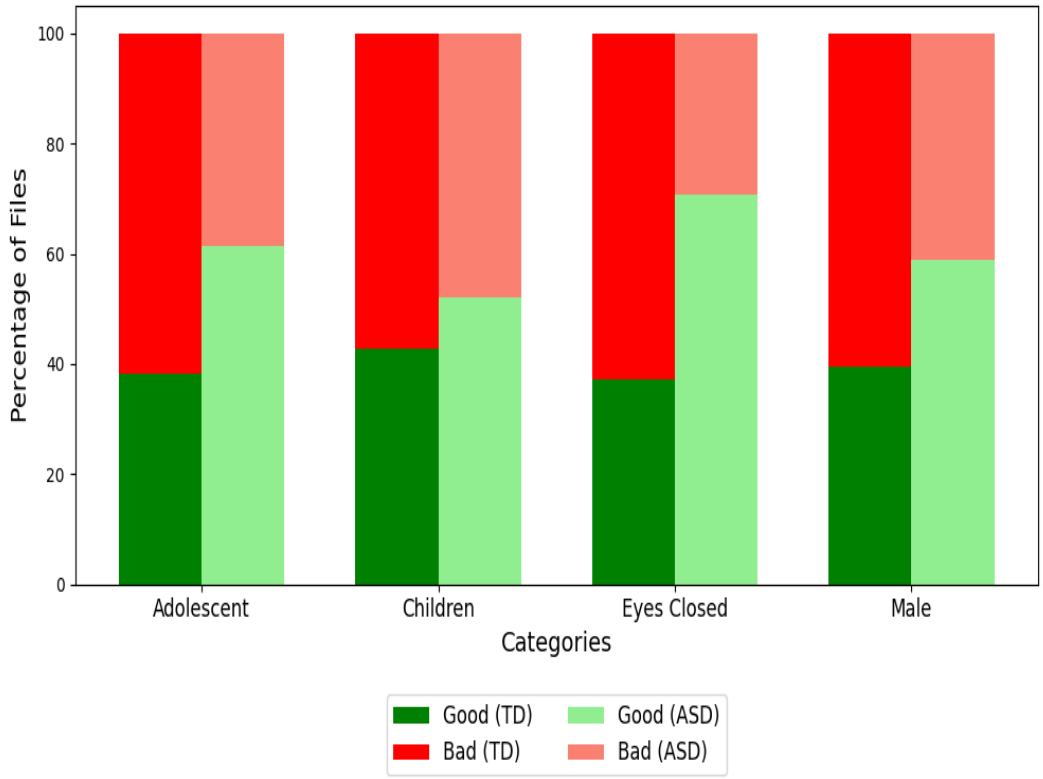


Figure 3.6: ASD Dataset: % of good and bad files using Jaccard Similarity

To summarize, the results obtained using the two approaches in RQ3 provide robust and persuasive evidence of a high degree of similarity between the graphs in the two categories: ASD and TD. This explains the challenges faced by our classification

methods and graph theoretic techniques used in prior research (e.g., contrast subgraph or discriminative edges method) in achieving strong performance metrics.

Run-time. All experiments were run on a Windows machine with Intel i5 CPU and 8 GB RAM. I report here the run times of our algorithms for RQ3. On the largest dataset (Male), a run of Algorithm 1 took 110 minutes to finish and on the smallest (Children), it took 75 seconds. This is because Algorithm 1 computes the Hamming distance between every pair of brain networks in the dataset. Algorithm 3 was much faster and required only 92 seconds on the Male dataset and 16 seconds on the Children dataset for one run as it only compared the max-core of 20% of the networks with the two summaries computed from the rest.

Similarity based on Eigenspectrum.

Our third approach uses the *eigenspectrum* of a graph as a "global" similarity measure. This approach complements the previous two approaches that rely on local and glocal measures.

Given any graph $G = (V, E)$ wih $|V| = n$, let $A(G)$ denotes its $n \times n$ adjacency matrix. Then, the n eigenvalues of A are the solutions to the system of linear equations $AX = \lambda X$ for some scalar λ and a non-zero vector X. I refer to λ as the eigenvalue and the corresponding X as the eigenvector. Any $n \times n$ symmetric matrix with real entries has n eigevalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ with the corresponding eigenvectors $X_1, X_2, \ldots, X_n$. Then the sequence of eigvalues in sorted order, $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$ is called its eigenspectrum.

Given two eigenspectrums, $A = [\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n]$ is and $B = [\lambda'_1 \leq \lambda'_2 \leq \ldots \leq \lambda'_n]$, their similarity is measured using the notion of L_2 distance. It is defined as

$$L_2(A, B) = \sqrt{\sum_{i=1}^{n} (\lambda_i - \lambda'_i)^2}$$

Small values of L_2 distance between two eigenspectrums indicate that the two are similar.

Algorithm 3 Similarity of Eigenspectrum

1: **Input**: A Dataset $D = \{G_1, G_2, \ldots, G_m\}$ consisting of two classes, ASD and TD files
2: **Output**: f_{ASD} and f_{TD} (the fraction of good ASD and TD files based on the similarity of eigenspectrum)
3: Partition D into four sets S_{ASD}, T_{ASD}, S_{TD}, and T_{TD} using 80:20 split.
4: Compute SG_{ASD} and SG_{TD} using 75% threshold.
5: Compute their eigenspectrum, ES_{ASD} and ES_{TD}
6: for each $G_i \in T_{ASD}$ do
7: Compute eigenspectrum ES_i of G_i
8: **if** $L_2(ES_i, ES_{ASD}) < L_2(ES_i, ES_{TD})$ **then**
9: $count_{ASD}$++; $good_{ASD}$++
10: **else**
11: $count_{ASD}$++
12: **end if**
13: for each $G_i \in T_{TD}$ do
14: Compute eigenspectrum ES_i of G_i
15: **if** $L_2(ES_i, ES_{TD}) < L_2(ES_i, ES_{ASD})$ **then**
16: $count_{TD}$++; $good_{TD}$++
17: **else**
18: $count_{TD}$++
19: **end if**
20: **return** $good_{ASD}/count_{ASD}$ and $good_{TD}/count_{TD}$

1. Given a dataset, I partition the ASD files in that dataset into two sets, S_{ASD} and T_{ASD} using an 80:20 split. Similarly, I partition the TD files into two sets, S_{TD} and T_{TD} using an 80:20 split (Line 3 of Algorithm 3).

2. Using the files in S_{ASD}, I create a single graph, SG_{ASD}, that I call the ASD summary graph. This graph is a graph on 116 vertices. It contains an edge (i, j) if and only if more than 75% of the graphs in S_{ASD} contain that edge. Similarly, I create the TD summary graph SG_{TD} (Line 4).

3. I compute the eigenspectrum of the two summary graphs, SG_{ASD} and SG_{TD}

(Line 5).

4. Now, for each file in T_{ASD}, I compute its eigenspectrum and check if it is closer to the eigenspectrum of SG_{ASD} or the eigenspectrum of SG_{TD} using the notion of L_2 distance. I say that it is *good* if it is closer to the eigenspectrum of SG_{ASD} and *bad* otherwise. Compute the percentage of good ASD files (Lines 6-12).

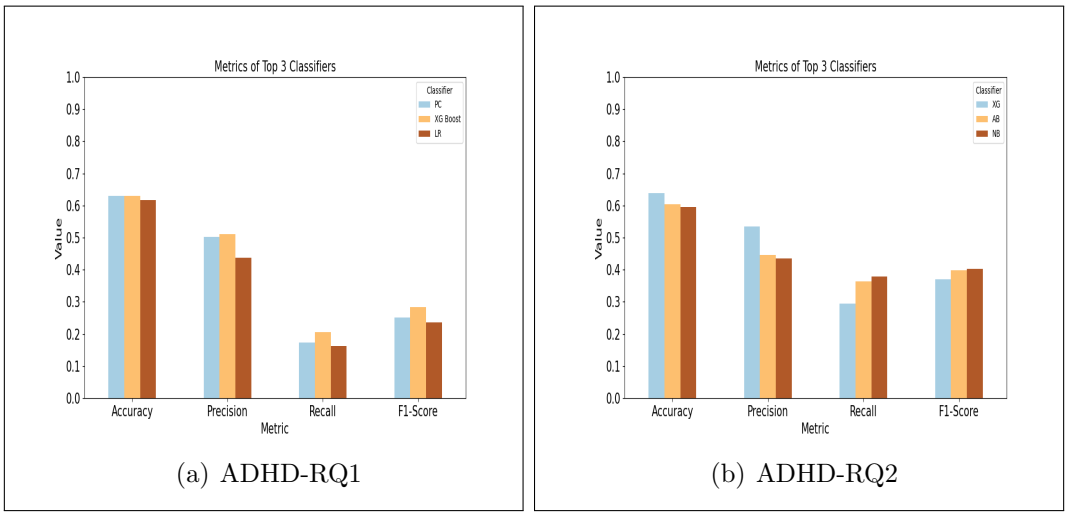
5. Repeat the previous step for TD files (Lines 13-19).

Our results of this approach were very surprising. For all the datasets, I observed that $f_{ASD} = 0$ and $f_{TD} = 1$. That is, all the ASD test files were bad and all the TD test files were good. Put differently, all the test files from both classes were more similar to the TD summary graph compared to the ASD summary graph. These results again reinforce our previous results that the eigenvalue based approach cannot be used to differentiate between the two classes effectively.

3.3.2 Insights on ADHD Dataset

Our results for RQ1-RQ3 prompt the question of whether these outcomes are specific to the ASD dataset I studied. I am interested in understanding whether our approach could yield different results when applied to a dataset focused on a different but closely related health condition. To address this, I conduct an investigation using the ADHD dataset and apply the same methodology as described for the ASD dataset. I noted that, for this dataset, the top-3 tabular classifiers achieved an accuracy of 63% for RQ1 and did not perform much better than the baseline classifier as seen in Figure 3.7(a) and Table 3.4

In addition, the inclusion of clustering coefficients as new attributes for RQ2 did not yield any significant enhancement in the performance metrics, as shown in Figure 3.7(b) and Table 3.4.



(a) ADHD-RQ1

(b) ADHD-RQ2

Figure 3.7: RQ1 and RQ2 - ADHD - Metrics

Accuracy	**PC**	**XG Boost**	**Logistic Regression**
RQ1	0.63	0.63	0.62
RQ2-Augmented Table	0.63	0.64	0.62

Table 3.4: ADHD Dataset: Accuracy

When I compared the two classes, ADHD and TD, for RQ3, the average percentage of good files remains around 50% for Jaccard similarity and 35% for Hamming distance, (refer to Figure 3.8). These results indicate that our findings for ASD datasets extend to the ADHD dataset as well.

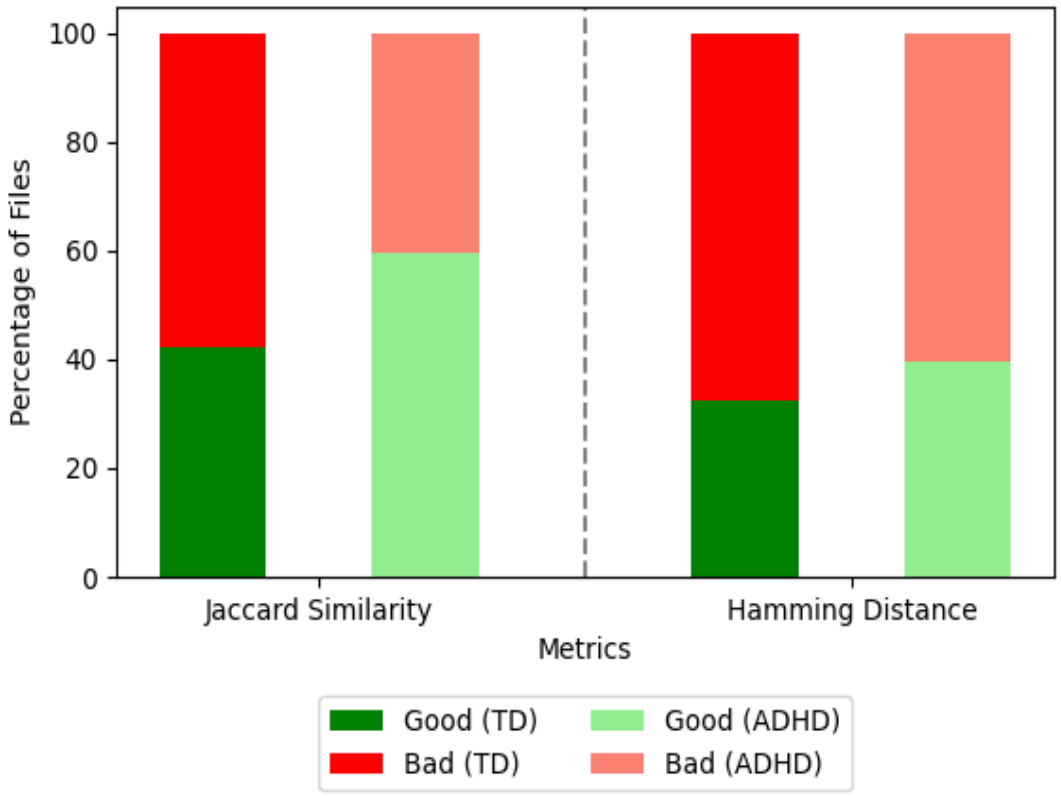


Figure 3.8: ADHD Dataset: % of good and bad files using Jaccard Similarity & Hamming Distance

Publication

The results in this chapter are based on the following publication(s).

- Kazi Tabassum Ferdous, Sowmya Balasubramanian, Venkatesh Srinivasan, and Alex Thomo. Brain network similarity using k-cores. *Proceedings of the IEEE/ACM International Symposium on Network Enabled Health Informatics, Biomedicine and Bioinformatics (HI-BI-BI)*, 2023, pages 575-582.

- Keanelek Enns, Kazi Tabassum Ferdous, Sowmya Balasubramanian, Smita

Ghosh, Venkatesh Srinivasan, and Alex Thomo. Are Brain Networks Classifiable? *Network Modeling Analysis in Health Informatics and Bioinformatics (NetMAHIB)*, Springer, 13(1), 44, 2024, 19 pages.

Chapter 4

Synthetic Generation of Patient Service Utilization Data: A Scalability Study

While high-quality health data plays a critical role in improving the quality of health care, there are several hurdles in acquiring such data for machine learning purposes due to privacy regulations and ethical concerns. To overcome this barrier, it is important to compare existing and design new methodologies for synthetic health data generation. Early simplistic methods failed to capture the complexity of real-world datasets, leading to the development of modern techniques like GANs, VAEs, copulaGAN, and transformer models. However, there is no study that carefully compares the scalability of state of the art methodologies for synthetic data generation (SDG) for patient service utilization data.

Our work systematically compares various SDG methods for patient service utilization data obtained from a regional health authority in Canada. I evaluate five SDG models across four patient cohorts, measuring training and generation times,

investigating the resemblance between real and synthetic data, and evaluating the utility of synthetic data in practical scenarios. Our findings show that statistical models outperform machine learning-based and hybrid models in training time and data generation. Moreover, our study demonstrates that the generated synthetic data from most models closely resembles real data and proves useful in practical applications, showing high accuracy in classifying synthetic data when trained on real data or vice versa using a Binary RNN classifier.

This chapter starts with an overview of the literature on the generation of synthetic data. Next, I introduce the dataset I use and the proposed methods for generating synthetic data. I then outline the three research questions and summarize our experimental results for each of them.

4.1 Related Work

There are numerous papers in the literature that propose methods for generating synthetic health data in different formats. These works are broadly classified into knowledge driven, data driven, and hybrid methods.

Knowledge-driven methods contain ground truth from publicly available sources that contain domain-specific knowledge. Examples of such sources include academic and research publications, web resources, and medical practitioners. This approach is solely theory based and involves the use of hand-crafted models based on a set of generation rules or state charts. Examples that use this method are Advanced Patient Data Generator (APDG) [42, 23] that relies on an XML-based Patient Data Definition Language (PDDL) to define the data generation rules and the state transition method (synthea) that uses state charts to formalize the domain knowledge [90, 60]. A notable fact of this approach is that it does not require and use real-world data in

this generation process.

In contrast, data driven approaches rely on a generative model that extracts knowledge from the real data in order to produce synthetic data. Popular techniques used in the extraction are either classical or machine learning based. Classical methods use statistical methods to estimate the underlying distribution of the real data and then derive synthetic samples using this knowledge [46]. Understanding the real data mostly involves estimating the parameters and the shape of the real distribution. A popular classical method is the copula method [92]. Copulas are particularly suited for multivariate data as they show how to link the marginal to the joint distribution. Many recent SDG approaches are based on machine learning. Examples include using Graph Adversarial Networks (GAN), neural networks, transformers, autoencoders, and decision trees [72, 20, 86, 80, 65].

Hybrid methods combine the two approaches above by building models with overall structure extracted from the data and the knowledge gained from theory making the resulting synthetic data more realistic [11, 52, 91, 75].

Once the synthetic data is generated, it is important to measure its quality. This is done using two measures: realism and privacy. Realism focuses on understanding how similar the synthetic data is to real data. It is studied using two types of metrics - resemblance metrics [34] and utility metrics [25]. Resemblance metrics quantify the statistical closeness between synthetic and real data while utility metrics study the usefulness of the synthetic data in practical applications. An example of a resemblance metric is dimension-wise statistics [94] in which I measure the similarity between the marginal distributions of real and synthetic data. An example of a utility metric is performance agreement in which I study the performance of a classifier when trained on real data and tested on synthetic data (TRTS) and vice versa [29]. Once the realism of the synthetic data is established, it is also important to ensure that it preserves

privacy [87] as data driven methods are known to be prone to privacy leakages.

4.2 Datasets and Methods

4.2.1 Health Service Data

The data sets I work with in this analysis are of patients accessing service classes within a regional health service (HS) system in Canada over the last six years. That is, let P be the set of all patients that access a health service system HS; and let S be the set of service classes in HS. Then for each patient $p \in P$, there is a time-ordered sequence of service classes that p accessed in the health service system HS, called their Patient Service Utilization (PSU). In the health service system I am working with in this chapter, there are 269 distinct service classes that patients can access. The datasets, a collection of health records obtained from each service location, are stored on a secure server and given to us in a de-identified form using an SQL relational database. The dataset contains many different pieces of information about each patient, but for the purposes of our analyses, I typically am most concerned with the order of events. So I collect the data into arrays of service class IDs, which indicate the order of event interactions for each patient. For simplicity, I do not include the date of interaction. While the date is useful information in many cases, I will start by simulating the sequence of events and explore simulating the dates in future work.

In a health service system, there are some patients that form natural groupings. These groupings are sometimes based on access to certain services or through specific health related events. I call these groupings cohorts. In our analysis, I have four different cohorts of patients each with their own defining features. There are 74 defined cohorts in our health service system, each with its own defining criteria. The cohorts I analyzed are listed in Table 4.1.

Cohort Name	Cohort Code	# Real Patients
Schizophrenia Services	SS	1829
Homeless – Ever	HE	2221
Addictions Service – Post Withdrawal	AS	2592
Opioid Overdose	OO	5381

Table 4.1: Cohorts

4.2.2 Preparing the Data

From our SQL database, the patient information is stored in a schema where each interaction gets its own row entry, containing additional information about the patients interaction. For each patient in a cohort, I collapse the interaction history into one long string of sequential events. Then all the patients in the cohort are entered into a new table with four columns, where each row has the patient ID, age, sex, and list of ordered service class interactions. For some data generation models, the data is inputted in this table format, these are referred to as Single Table Data models. For the transformer though, I convert this table into one long string. In order for the GPT model to learn the data I format it in the following manner:

```
Patient <current age> <sex> :
<PSU>
```

For each cohort of patients, I prepare a text file that contains all PSUs. After the model has trained on this data format, it will then reproduce it when asked to generate new content. The synthetic data now comes in a predictable format that will allow easy post processing.

4.2.3 Synthetic Data Generation Models

Our work utilizes a broad spectrum of data synthesizers. These include statistical models (Gaussian copula, Fast ML), deep learning-based generative models (GAN and GPT), and a combination of the two (CopulaGAN). Except for the GPT model that we implemented ourselves, the code for the other methods is taken from Synthetic Data Vault (SDV), a public, source-available Python library for generating and evaluating synthetic data. For all the SDV models, I used the default settings when training the model and when generating the synthetic datasets. These can be found on the documentation page for the SDV models.

The Gaussian Copula Model It is a classical, statistical model that utilizes two main properties of the real-world data viewed as a single table: its distributions and covariances. The marginal distributions describe the values in each column. This can be done, for example, by giving its CDF. However, it is less expensive computationally to calculate how closely the distribution resembles one of the well-known distributions: truncated normal, uniform, beta, or exponential and choose the best. Besides the individual distributions, the model must also take into account the dependencies between them. That is, how the values in one affect the values in the other. This is done by calculating the covariance between them. Finally, to mitigate the bias that the shape of one marginal distribution can have on another, this method uses the Gaussian Copula to fit a standard normal distribution to each of the columns before calculating the pairwise covariances. The *Fast ML* synthesizer specifically focuses on modeling speed. Formally, it uses the Gaussian copula synthesizer with fixed settings.

The CTGAN Model Generative adversarial networks are deep learning based generative models. In general, GANs can be thought of as an architecture for training a generative model consisting of two sub-models: a generator and a discriminator.

The generator takes a random vector of fixed length as input and outputs a sample from the problem domain. Its adversary, the discriminator, attempts to distinguish samples output by the generator from the samples drawn from the training data. Both the generator and the discriminator are updated in rounds based on their performance viewed as a zero-sum game. This continues until the generator produces samples that reduce the success probability of the discriminator to about 50%. While traditionally, GANs were used for generating images, they have been extended recently to generate tabular data using conditional GANs such as CTGAN. In conditional GANs, both the generator and the discriminator are given additional information such as an example from the domain.

The CopulaGAN Model As its name indicates, this model combines statistical learning with GAN-based learning. In the statistical learning phase, the synthesizer learns the shape of the distribution corresponding to each column, also referred to as the marginal distribution. For example, this distribution could be a truncated normal or a beta distribution. It then applies the Gaussian normalization procedure in which it fits each distribution to a standard normal distribution with a mean of 0 and a standard deviation of 1. In the GAN-based learning phase, it uses a conditional GAN to model the normalized data obtained from the statistical learning phase. In summary, this model is a combination of the Gaussian copula model and the CTGAN model.

The GPT Model The GPT models are neural network and deep learning based language prediction models that use the transformer architecture. Specifically, I implemented a GPT model consisting of four layers: an embedding layer with 64 nodes, two hidden layers, four self-attention heads, and a learning rate of 0.001. The model itself uses Bigrams along with positional embeddings to capture the positional infor-

mation of the dataset. The data was split 3:1 for training and testing, with batch and block sizes of 16 and 32 respectively. Once the model is trained, I generate text until I have a data set of comparable size as the input. The token set is done at the word level, making each service class id a token. This is to stop the model from making new IDs.

These models were selected for testing because they are all open source and have Python API available. Every model with the exception of the transformer model comes from Synthetic Data Vaut (SDV) and works for single table data, meaning that data needed to be simulated is completely contained in one table and is not part of a larger SQL schema. The transformer code was adapted from a simple implementation of Bigram Language Model using the PyTorch library.

4.2.4 Validation metrics

Distributions The data produced from the model constitutes our synthetic data, as I show later in Section 4.3. The synthetic data is similar in kind yet is a completely different population. That is, let T be the true population of real patients, and let S be the population of synthetic patients. T and S are sets of patients with a particular ailment (like Schizophrenia), but the patients in each are completely different records. If this is the case, then I expect that the distributions of characteristic variables would be the same. Hence to validate the synthetic data, I compare distributions of the data characteristics like the number of services visited by patients, distribution of ages, number of times services were visited by the cohort, and number of patients that visit each service in the cohort. Next, I discuss measures for detecting similarity between two distributions.

K-L Divergence The distance between the distributions of the two datasets can be measured with K-L Divergence. Specifically, let V, be defined as the set of values the distributions can take, let $Q(x)$ be the probability distribution of the values in V for the real dataset, and let $P(x)$ be the probability distribution of the values in V for the synthetic dataset. Then K-L Divergence is

$$D_{KL}\left(P||Q\right) = \sum_{x \in V} P(x) \log \frac{P(x)}{Q(x)} \tag{4.1}$$

Jensen-Shannon Divergence Since K-L Divergence can range from zero to positive infinity, I also calculate the Jensen-Shannon divergence which is a normalized variant of K-L Divergence. The Jensen-Shannon Divergence is a symmetric measure of the similarity between two probability distributions, derived from the K-L Divergence, and ranges from 0 (identical distributions) to 1 (completely dissimilar distributions).

$$JSD\left(P||Q\right) = \frac{1}{2}\left[D_{KL}\left(P||M\right) + D_{KL}\left(Q||M\right)\right] \tag{4.2}$$

where $M = \big(P + Q\big)/2$.

In experimentation, I only use JSD as it is the normalized measure, and thus the results are easier to interpret. That is, the distributions are identical when $JSD = 0$, and they are maximally different when $JSD = 1$.

RNN Classification In this work, I take a pair of down sampled cohorts, and train a Recurrent Neural Network (RNN), using the real and synthetic datasets, to classify patients into one of the two cohorts. Then I take the synthetic data trained RNN and ask the model to classify the never-before-seen real data. To use this model as a validation metric, I compare the train confusion matrix to the confusion matrix obtained from predicting on the synthetic data. That is if both the real and synthetic

data have high percentages (90% and up) in the confusion matrices, then this gives a very strong indication that the patterns within the cohort PSU are similar in the two datasets.

4.3 Results and Discussion

To compare our five synthetic data generation methods, I produce four sets of synthetic data for the datasets listed in Table 4.1 and perform a series of experiments comparing the synthetic data to the real data from which it was generated. Due to the sensitive nature of our data, all models were trained and generated on a secure server through a virtual machine with 16Gb of RAM emulating Windows 10.

RQ1: How do the generative models compare for training and producing synthetic data?

The training times for each model across the four cohorts were recorded and compiled in Table 4.2. It's worth noting that cohorts SS, HE, and AS share a similar size range (around 1800 to 2600 entries), while the OO cohort is more than double the size (approximately 5400 entries). Particularly, the SDV models required notably more time to train on the OO cohort compared to the other three cohorts.

I observe that, although Fast ML and the Gaussian Copula use the same methodology their training times for the four cohorts vary significantly. Also, note that the Fast ML is 4x slower when training on each cohort than the Gaussian Copula model, even though it is supposed to be optimized for speed. The training time for CTGAN and the Copula GAN models doubles from SS to HE and from HE to AS. Furthermore, their run on the OO cohort is 10x more than the run time on AS. In comparison, the Transformer model effectively takes the same amount of time, about 455 seconds, with the settings I used. I found that the Transformer's run time is only

affected by its hyperparameters. That is, when the sizes of the neural net number of layers, heads, and nodes in the embedding layer increases; the training time also increases. Overall, the Gaussian Copula is the fastest model and the Transformer is the only model that was unaffected by the cohort size.

Generative Method	SS	HE	AS	OO
Fast ML	1.75s	2.29s	2.91s	8.37s
Gaussian Copula	0.55s	0.53s	0.48s	1.20s
CTGAN	2973.24s	6748.23s	14176.77s	170854.28s
CopulaGAN	2973.01s	6740.78s	14039.20s	204707.17s
Transformer	469.23s	453.68s	454.34s	459.88s

Table 4.2: Time to fit each Cohort set to each model

Next, I analyze the time to generate synthetic data from the trained models. Using each trained model, synthetic data sets were generated at increasingly larger sizes compared to the original size of the cohort. The results are shown in Figures 4.1 to 4.5. Each figure contains four canvases with two lines in each canvas. Each canvas represents one of the four cohorts. The solid blue line is the measured time with linear interpolation, and the dashed orange line is the Out of all the models. The Transformer performs the slowest at data generation, taking over 8 hours to produce a dataset 10x the size of the original. As shown in Figure 4.1, the time to generate data sets is linear with respect to the number of records in the generated data sets. All the other four models, Fast ML, Gaussian Copula, CTGAN, and Copula GAN, exhibit *super-linear* generation time. That is, as the size of the synthetic dataset grows, the time required to generate it grows at a rate faster than linear. The Fast ML and Gaussian Copula models generate datasets the fastest, taking less than 10 minutes to generate a dataset 100x the size of the original (See Figures 4.2, 4.3). The

CTGAN and Copula GAN models, while super-linear, take much longer than Fast ML and Gaussian Copula, but are still faster than the Transformer model; taking under an hour to generate datasets 100x the original size (See Figures 4.4, 4.5). Also note that the Copula GAN model takes slightly longer to generate datasets compared to CTGAN, while still outperforming the Transformer model. Regarding the sizes of the generated datasets, I note that due to memory limitations of the VM, the transformer model was not able to generate a dataset larger than 10x the OO cohort. In contrast, I was able to use the CTGAN and CopulaGAN models to generate a dataset 30x-45x the original for the OO cohort and was able to generate 100x of the OO cohort using the Fast ML and Gaussian Copula models. To summarize, for both model training and data generation, the Gaussian Copula and the Fast ML models perform the best. This leads to our next question: how good are these models at maintaining the distributions within each cohort when synthetic data is generated?

RQ2: How do the generation models perform with respect to resemblance metrics? That is, how closely does the synthetic data resemble the real data?

For the four cohorts, I focus on the marginal probability distributions for six features and joint distributions for two pairs of features. I will describe them now.

1. Ages
2. Number interactions is the number of services accessed by each patient.
3. Number interactions male only
4. Number interactions female only
5. Total counts the number of interactions all the patients in a cohort have with each service class.

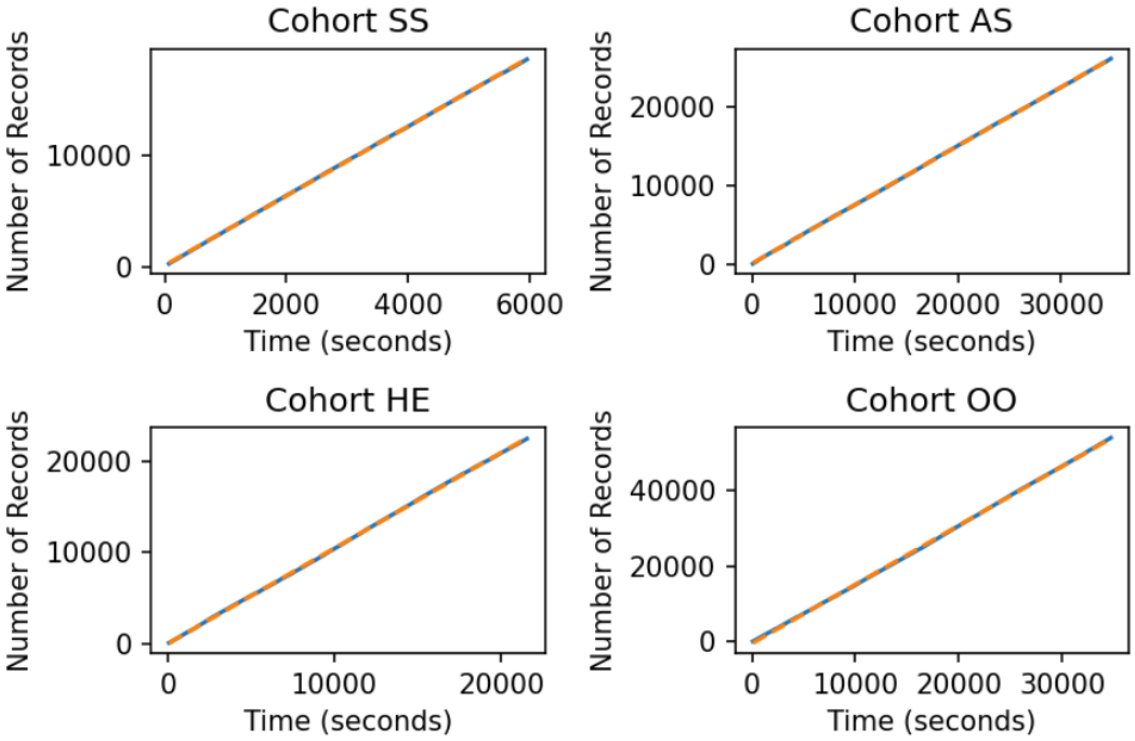


Figure 4.1: Time to Generate Data with the Transformer model.

6. Usage counts the number of distinct patients in a cohort that access each service. Note, in this measure a patient only contributes one at most to each count.

7. (Age, lengths) is the joint distribution of ages and number of interactions.

8. Usage/Total is the distribution of the ratio of the usage count over the total count.

These eight distributions were computed for both the real and synthetic datasets. Then, for each of the eight distributions, the Jensen-Shannon Divergence between the distributions for real and synthetic datasets was calculated. The results for cohort SS are given in Tables 4.3 to 4.7. The results for the other three cohorts are in Section 4.4. The general trend across Tables 4.3 to 4.7 is that the Jensen-Shannon distance becomes smaller as the size of the synthetic dataset increases indicating

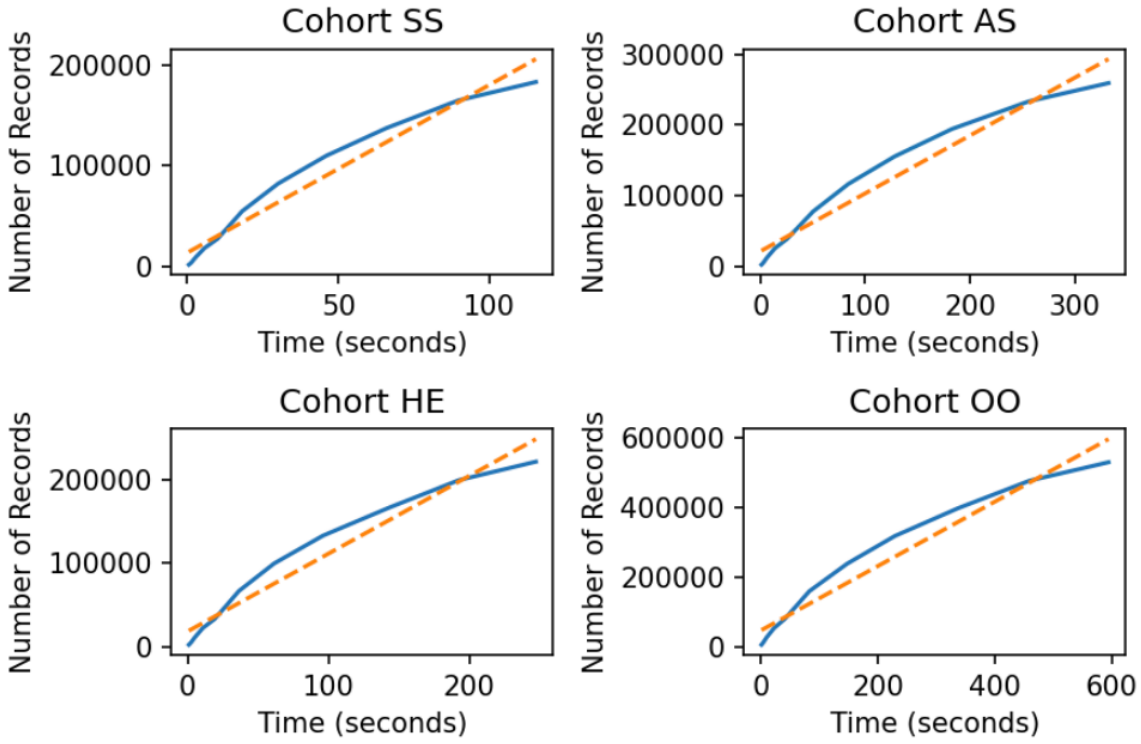


Figure 4.2: Time to Generate Data with the Fast ML model.

higher resemblance. While the general trend shows that increasing dataset size yields smaller Jensen-Shannon Distance, there are some anomalies that go against this pattern. Part of the reason for this divergence is that I am analysing the results of one run of a statistical process. To get a better picture, I would need to generate at least thirty distinct datasets at each scaled size and calculate the Jensen-Shannon distance for all of them, then average the results. However, I believe that these results on one instance of the synthetic data are sufficient to establish general trends in behavior as I scale to larger sizes.

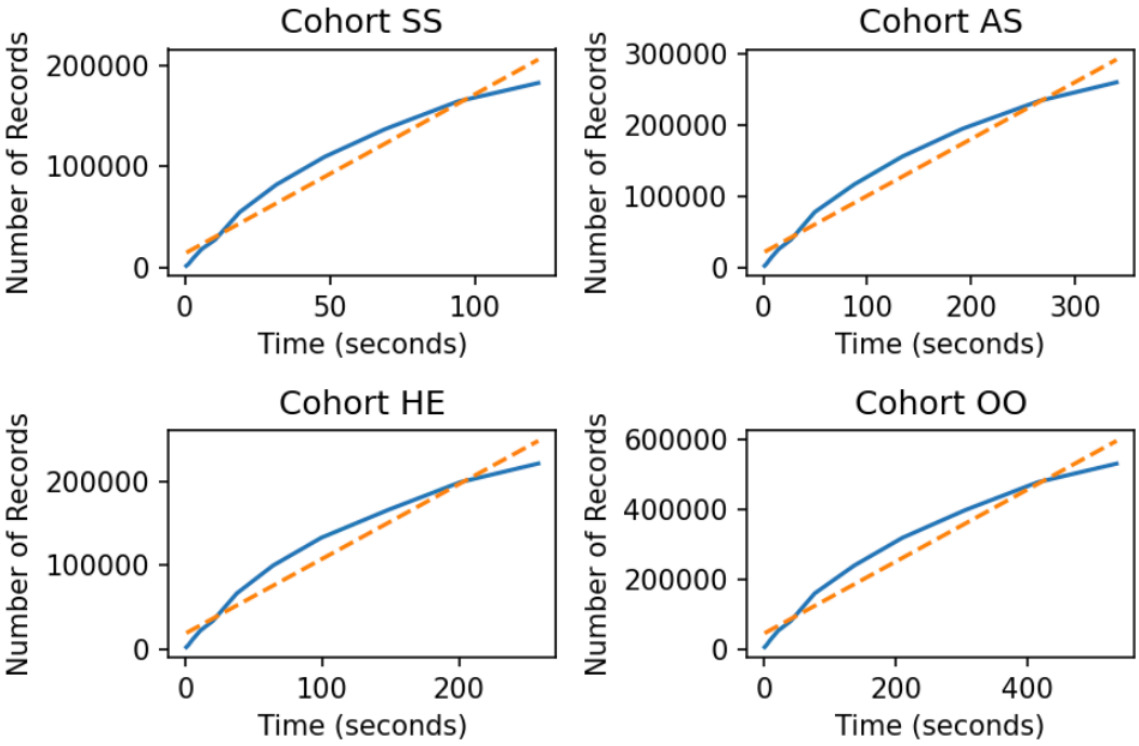


Figure 4.3: Time to Generate Data with the Gaussian Copula model.

	1x	2x	5x	10x
ages	1.341	1.314	1.304	1.289
number interactions	0.03	0.011	0.005	0.004
number interactions (men only)	0.013	0.011	0.01	0.007
number interactions (women only)	0.458	0.227	0.365	0.362
total	0.287	0.111	0.073	0.052
usage	0.151	0.093	0.058	0.057
(ages, lengths)	7.632	7.77	7.569	7.345
usage/total	0.247	0.116	0.033	0.03

Table 4.3: Jensen Shannon Distance in percentages for Cohort SS with the Fast ML Model as the dataset size scales

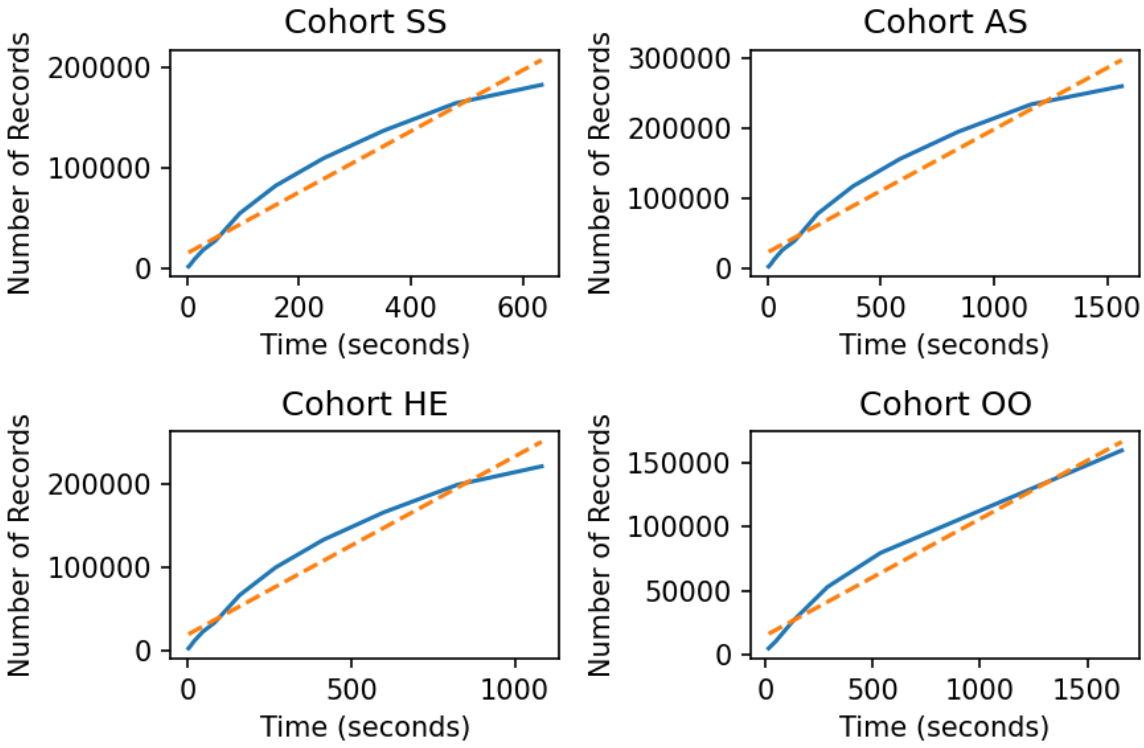


Figure 4.4: Time to Generate Data with the CTGAN model.

	1x	2x	5x	10x
ages	0.683	0.546	0.513	0.481
number interactions	0.476	0.003	0.002	0.002
number interactions (men only)	0.661	0.021	0.012	0.014
number interactions (women only)	0.455	0.46	0.31	0.326
total	0.099	0.046	0.032	0.011
usage	0.072	0.039	0.023	0.011
(ages, lengths)	39.4	5.1	4.3	4.1
usage/total	0.157	0.148	0.04	0.012

Table 4.4: Jensen Shannon Distance in percentages for Cohort SS with the Gaussian Copula Model as the dataset size scales

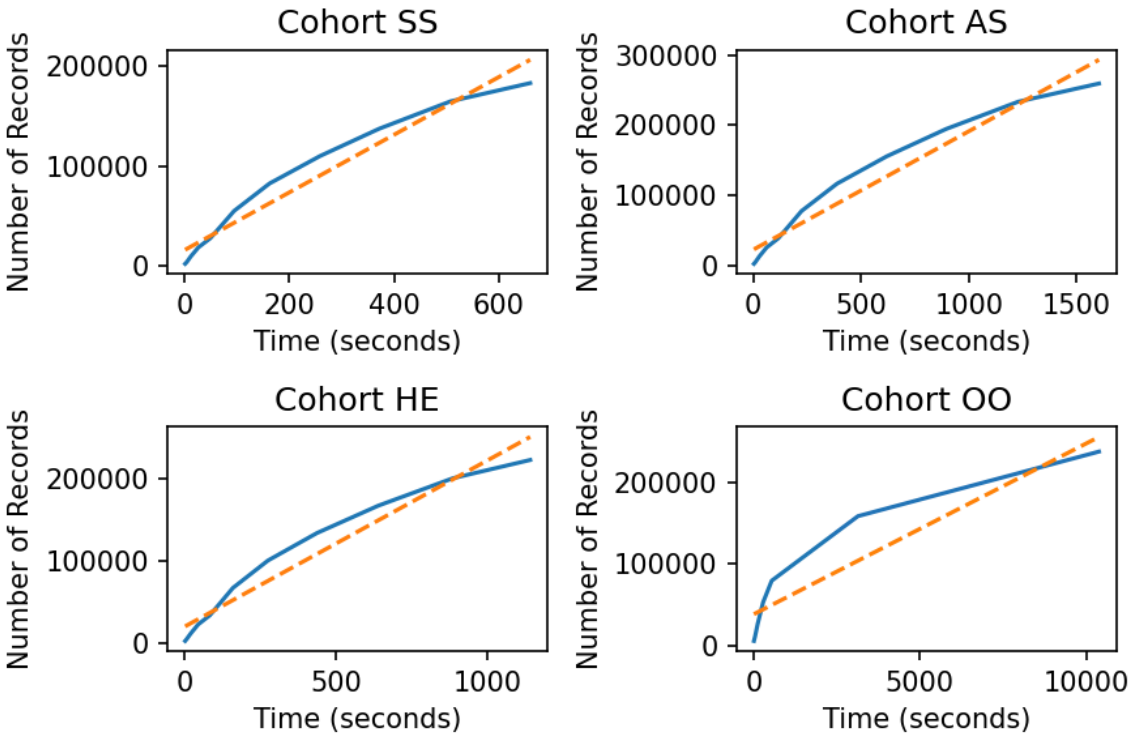


Figure 4.5: Time to Generate Data with the CopulaGAN model.

	1x	2x	5x	10x
ages	2.388	0.737	1.323	2.079
number interactions	0.066	0.005	0.003	0.003
number interactions (men only)	0.111	0.018	0.013	0.014
number interactions (women only)	0.233	0.102	0.347	0.317
total	0.213	0.103	0.08	0.041
usage	0.127	0.067	0.049	0.037
(ages, lengths)	57.2	9.7	11.2	13.4
usage/total	0.179	0.091	0.043	0.039

Table 4.5: Jensen Shannon Distance in percentages for Cohort SS with the CTGAN Model as the dataset size scales

	1x	2x	5x	10x
ages	0.828	0.472	0.439	0.422
number interactions	0.467	0.075	0.003	0.003
number interactions (men only)	0.566	0.125	0.017	0.024
number interactions (women only)	0.076	0.074	0.361	0.348
total	0.228	0.144	0.117	0.096
usage	0.141	0.127	0.083	0.07
(ages, lengths)	42.6	57.7	6.209	5.848
usage/total	0.177	0.171	0.043	0.048

Table 4.6: Jensen Shannon Distance in percentages for Cohort SS with the copula GAN Model as the dataset size scales

	1x	2x	5x	10x
ages	0.588	0.598	0.601	0.619
number interactions	1.468	1.42	0.441	0.44
number interactions (men only)	1.546	1.504	0.465	0.458
number interactions (women only)	0.708	0.661	0.251	0.247
total	1.289	1.209	1.232	1.263
usage	0.643	0.617	0.572	0.558
(ages, lengths)	49.8	49.5	50.6	50.7
usage/total	1.054	0.903	0.771	0.775

Table 4.7: Jensen Shannon Distance in percentages for Cohort SS with the Transformer Model as the dataset size scales

All the numbers reported in Tables 4.3 to 4.7 are given as percentages. From the values in the tables above, I see that the marginal distributions for the synthetic data very sharply resemble that of the real data. With the exception of ages which ranged

between 0.5 and 3 percent, all the marginal distributions had a Jensen-Shannon Distance well under 1%. For the joint distribution of age and length, the JSD was not as good, for most models it stayed above 5% and when the dataset size was small (1x) this value was as high as 50% for some models. However, the Fast ML model challenged this trend by remaining around 7% regardless of dataset size. The Gaussian Copula model performed well with the JSD drop to 4% for the joint distribution of age and length after the 1x dataset. The CTGAN and Copula GAN model, both started out with a JSD for the age length joint distribution around 50%, which then dropped off at higher dataset sizes to about 10 and 6 percent respectively. The Transformer model never improved its JSD for the (age, length) joint distribution as the size increased, hovering at 50% for each scaled-up dataset.

RQ3: How do the generation models perform with respect to the utility metrics? That is, how useful is the synthetic data in practical scenarios?

To test the utility of the synthetic data, I invoke the Binary RNN model to learn to distinguish between patients from two different cohorts based on their PSU data. Each RNN model was trained over 20 epochs with batch and buffer of 100 and 16 respectively, and a learning rate of 0.001. The training datasets were down sampled so that there was an equal number of both cases.

There are two ways to train and test with the RNN models. The first is to train them with the real dataset, and then test how accurately they predict on the synthetic data. This method is referred to as **Train on Real, Test on Synthetic** or **TRTS**. The second method is to train five models on each of the synthetically generated datasets and test how accurately they predict the real dataset. This method is referred to as **Train on Synthetic, Test on Real** or **TSTR**.

In Tables 4.8 through 4.16 I perform three sets of experiments on the differently scaled synthetic datasets. First at the original size (1x), then at 5x, and 10x. For

each set I train six models, the real and the five synthetically trained models, then I run the TSTR followed by the TRTS. Note that for the scaled-up dataset sizes, the real dataset never scales, since I cannot generate an arbitrary number of patient records like with the synthetic data.

In the first set of experiments (Tables 4.8-4.10), I train six models, the real datasets, and the five synthetic models. I evaluate their performance during the training phase by computing the four performance metrics using the validation data. I note that all four metrics have high values ranging from 0.951 to 1.0 for all six models.

Table 4.8: Performance metrics for SS vs. AS during training each RNN model. The datasets used were 1x the original size.

	Accuracy	F-measure	Precision	Recall
real	0.984	0.984	0.991	0.976
transformer	0.976	0.975	0.967	0.983
gaussian copula	0.981	0.982	0.985	0.979
fast ml	0.991	0.991	0.986	0.995
CTGAN	0.98	0.98	0.986	0.973
CopulaGAN	0.981	0.981	0.976	0.987

Table 4.9: Performance metrics for SS vs. AS during training each RNN model. The datasets used were 5x the original size.

	Accuracy	F-measure	Precision	Recall
real	0.959	0.959	0.958	0.96
transformer	0.965	0.963	0.968	0.957
gaussian copula	0.989	0.989	0.981	0.998
fast ml	0.987	0.987	0.975	1.0
CTGAN	0.985	0.986	0.974	0.997
CopulaGAN	0.988	0.988	0.979	0.997

Table 4.10: Performance metrics for SS vs. AS during training each RNN model. The datasets used were 10x the original size.

	Accuracy	F-measure	Precision	Recall
real	0.959	0.958	0.951	0.966
transformer	0.961	0.961	0.971	0.951
gaussian copula	0.986	0.986	0.974	0.999
fast ml	0.988	0.988	0.977	1.0
CTGAN	0.988	0.988	0.978	1.0
CopulaGAN	0.989	0.989	0.982	0.997

In the second set of experiments (Tables 4.11-4.13), I focus on TSTR, in which the models are trained on synthetic data and tested on real data. I note that all the models achieve excellent performance metrics ranging from 0.946 to 0.999 with CopulaGAN being the best.

Table 4.11: TSTR: Performance metrics for SS vs. AS using 1x Datasets.

	Accuracy	F-measure	Precision	Recall
transformer	0.968	0.968	0.982	0.954
gaussian copula	0.981	0.981	0.984	0.978
fast ml	0.984	0.984	0.98	0.987
CTGAN	0.979	0.979	0.985	0.974
CopulaGAN	0.981	0.981	0.985	0.977

Table 4.12: TSTR: Performance metrics for SS vs. AS using 5x Datasets.

	Accuracy	F-measure	Precision	Recall
transformer	0.96	0.961	0.946	0.977
gaussian copula	0.988	0.988	0.979	0.997
fast ml	0.986	0.986	0.975	0.998
CTGAN	0.986	0.986	0.975	0.998
CopulaGAN	0.988	0.988	0.979	0.997

Table 4.13: TSTR: Performance metrics for SS vs. AS using 10x Datasets.

	Accuracy	F-measure	Precision	Recall
transformer	0.956	0.956	0.95	0.962
gaussian copula	0.988	0.988	0.977	0.999
fast ml	0.988	0.988	0.976	0.999
CTGAN	0.988	0.988	0.978	0.999
CopulaGAN	0.988	0.988	0.979	0.997

In the third set of experiments (Tables 4.14-4.16), I focus on TRTS, in which the models are trained on real data and tested on synthetic data. Again, all models

performed well with metric values ranging from 0.915 to 0.99. In these experiments, CTGAN and CopulaGAN models achieved the best performance.

Table 4.14: TRTS: Performance metrics for SS vs. AS using 1x Datasets.

	Accuracy	F-measure	Precision	Recall
transformer	0.941	0.94	0.963	0.917
gaussian copula	0.979	0.979	0.971	0.987
fast ml	0.975	0.975	0.971	0.98
CTGAN	0.98	0.98	0.973	0.989
CopulaGAN	0.98	0.98	0.971	0.99

Table 4.15: TRTS: Performance metrics for SS vs. AS using 5x Datasets.

	Accuracy	F-measure	Precision	Recall
transformer	0.942	0.94	0.962	0.92
gaussian copula	0.978	0.978	0.97	0.986
fast ml	0.979	0.979	0.971	0.987
CTGAN	0.978	0.979	0.969	0.988
CopulaGAN	0.977	0.977	0.967	0.987

Table 4.16: TRTS: Performance metrics for SS vs. AS using 10x Datasets.

	Accuracy	F-measure	Precision	Recall
transformer	0.938	0.937	0.96	0.915
gaussian copula	0.975	0.975	0.966	0.985
fast ml	0.979	0.979	0.971	0.987
CTGAN	0.978	0.978	0.97	0.986
CopulaGAN	0.978	0.979	0.97	0.987

Looking across the second and third sets of experiments (1x, 5x, and 10x in both TRTS and TSTR) I found that the metrics values maintained percentages in the high 90's. Only the Transformer declined as the scale of the datasets increased, albeit still above the ninetieth percentile. Other than the Transformer, the remaining models performed more or less the same in all experiments, with percentages only subtly changing between them.

4.4 RQ2 Results for Other Cohorts

1. Results of RQ2 for the cohort AS

The general trend observed for cohort SS is also observed for cohort AS (Tables 4.17 to 4.21): Jensen-Shannon distance becomes smaller as the size of the synthetic dataset increases indicating a higher resemblance (with a few anomalies). Recall that the numbers reported in Tables 4.17 to 4.21 are given as percentages. I note that most models perform well for the marginal distributions with Jensen-Shannon distance below 1%. The most prominent exception was the transformer model for which the Jensen-Shannon distance for ages was around 3%. Many models had a high Jensen-Shannon distance for one joint distribution: (age, length) with the largest values arising from the transformer model.

	1x	2x	5x	10x
ages	0.52	0.406	0.407	0.428
number interactions	0.004	0.003	0.001	0.002
number interactions (men only)	0.026	0.009	0.008	0.01
number interactions (women only)	0.275	0.009	0.004	0.005
total	0.217	0.237	0.196	0.188
usage	0.445	0.465	0.454	0.448
(ages, lengths)	6.028	4.99	4.497	4.435
usage/total	0.265	0.198	0.151	0.149

Table 4.17: Jensen Shannon Distance for Cohort AS with the Fast ML Model as the dataset size scales

	1x	2x	5x	10x
ages	0.161	0.079	0.086	0.076
number interactions	0.001	0.001	0.001	0.0
number interactions (men only)	0.007	0.083	0.006	0.007
number interactions (women only)	0.003	0.005	0.001	0.003
total	0.052	0.019	0.011	0.008
usage	0.043	0.023	0.01	0.005
(ages, lengths)	4.487	3.34	2.911	2.817
usage/total	0.192	0.072	0.028	0.015

Table 4.18: Jensen Shannon Distance for Cohort AS with the gaussian-copula Model as the dataset size scales

	1x	2x	5x	10x
ages	1.308	0.609	0.581	0.736
number interactions	0.003	0.007	0.001	0.0
number interactions (men only)	0.01	0.003	0.01	0.007
number interactions (women only)	0.009	0.014	0.002	0.002
total	0.05	0.03	0.016	0.01
usage	0.055	0.027	0.019	0.017
(ages, lengths)	5.681	59.534	4.444	4.165
usage/total	0.158	0.052	0.022	0.014

Table 4.19: Jensen Shannon Distance for Cohort AS with the CTGAN Model as the dataset size scales

	1x	2x	5x	10x
ages	0.84	1.412	1.132	0.648
number interactions	0.002	0.001	0.0	0.0
number interactions (men only)	0.009	0.008	0.008	0.007
number interactions (women only)	0.005	0.008	0.005	0.004
total	0.056	0.025	0.012	0.006
usage	0.062	0.031	0.014	0.009
(ages, lengths)	5.979	4.953	4.708	4.761
usage/total	0.206	0.059	0.04	0.022

Table 4.20: Jensen Shannon Distance for Cohort AS with the copula GAN Model as the dataset size scales

	1x	2x	5x	10x
ages	3.21	3.196	3.214	3.237
number interactions	0.063	0.073	0.296	0.294
number interactions (men only)	0.066	0.06	0.128	0.128
number interactions (women only)	0.081	0.079	0.312	0.31
total	0.853	0.868	0.943	0.916
usage	1.174	1.155	1.137	1.134
(ages, lengths)	57.928	56.225	41.985	41.948
usage/total	0.648	0.676	0.691	0.691

Table 4.21: Jensen Shannon Distance for Cohort AS with the Transformer Model as the dataset size scales

2. Results of RQ2 for the cohort HE

As before, the general trend observed for cohort SS is also noticed for cohort HE (Tables 4.22 to 4.26): Jensen-Shannon distance becomes smaller as the size of the synthetic dataset increases indicating a higher resemblance. All the numbers reported in Tables 4.22 to 4.26 are given as percentages. I note that most models perform well for the marginal distributions with Jensen-Shannon distance below 1%. Compared to the SS and SS cohorts, an important difference is that all models have a much higher Jensen-Shannon distance of about 3% for ages. As seen for other cohorts, many models had a higher JS distance for one joint distribution: (age, length) with the largest distance arising from the transformer model.

	1x	2x	5x	10x
ages	2.271	1.955	2.097	2.094
number interactions	0.001	0.001	0.0	0.0
number interactions (men only)	0.001	0.001	0.001	0.001
number interactions (women only)	0.379	0.908	0.831	0.83
total	0.108	0.101	0.053	0.055
usage	0.16	0.157	0.133	0.135
(ages, lengths)	4.948	4.446	3.989	3.872
usage/total	0.303	0.09	0.041	0.037

Table 4.22: Jensen Shannon Distance for Cohort HE with the Fast ML Model as the dataset size scales

	1x	2x	5x	10x
ages	3.496	3.22	3.332	3.119
number interactions	0.0	0.132	0.0	0.0
number interactions (men only)	0.0	0.132	0.001	0.0
number interactions (women only)	0.321	0.384	0.814	0.821
total	0.074	0.028	0.013	0.011
usage	0.056	0.034	0.022	0.008
(ages, lengths)	12.809	55.404	11.18	10.697
usage/total	0.296	0.039	0.03	0.012

Table 4.23: Jensen Shannon Distance for Cohort HE with the gaussian-copula Model as the dataset size scales

	1x	2x	5x	10x
ages	3.815	2.9	2.753	3.408
number interactions	0.158	0.136	0.0	0.0
number interactions (men only)	0.149	0.177	0.001	0.126
number interactions (women only)	0.185	0.374	0.886	0.809
total	0.104	0.028	0.037	0.011
usage	0.084	0.052	0.032	0.02
(ages, lengths)	51.276	51.433	7.912	9.681
usage/total	0.319	0.114	0.035	0.036

Table 4.24: Jensen Shannon Distance for Cohort HE with the CTGAN Model as the dataset size scales

	1x	2x	5x	10x
ages	4.474	3.281	3.664	3.273
number interactions	0.125	0.134	0.001	0.001
number interactions (men only)	0.127	0.131	0.001	0.001
number interactions (women only)	0.332	0.303	0.356	0.818
total	0.108	0.1	0.067	0.039
usage	0.153	0.103	0.081	0.06
(ages, lengths)	48.622	48.015	19.713	17.397
usage/total	0.149	0.146	0.045	0.03

Table 4.25: Jensen Shannon Distance for Cohort HE with the copula GAN Model as the dataset size scales

	1x	2x	5x	10x
ages	2.325	2.284	2.347	2.266
number interactions	0.051	0.053	0.067	0.014
number interactions (men only)	0.048	0.053	0.162	0.043
number interactions (women only)	0.219	0.38	0.506	0.684
total	1.569	1.616	2.054	1.601
usage	1.404	1.276	1.059	1.286
(ages, lengths)	65.464	64.547	62.877	67.905
usage/total	0.644	0.651	0.731	0.626

Table 4.26: Jensen Shannon Distance for Cohort HE with the Transformer Model as the dataset size scales

3. Results of RQ2 for the cohort OO

The general trend observed for previous cohorts is also seen for cohort OO (Tables 4.27 to 4.31): Jensen-Shannon distance becomes smaller as the size of the synthetic dataset increases indicating a higher resemblance. All the numbers reported in Tables 4.27 to 4.31 are given as percentages. I note that most models perform well for the marginal distributions with Jensen-Shannon distance below 1%. A prominent exception was the Jensen-Shannon distance for ages for the copula GAN model which was around 3%. Many models had a higher Jensen-Shannon distance for one joint distribution: (age, length) with the largest distance arising from the transformer model.

	1x	2x	5x	10x
ages	0.562	0.416	0.309	0.324
number interactions	0.0	0.0	0.0	0.0
number interactions (men only)	0.0	0.001	0.0	0.0
number interactions (women only)	0.522	0.058	0.016	0.277
total	0.15	0.134	0.141	0.132
usage	0.248	0.217	0.221	0.218
(ages, lengths)	3.457	3.03	2.827	2.835
usage/total	0.176	0.077	0.054	0.05

Table 4.27: Jensen Shannon Distance for Cohort OO with the Fast ML Model as the dataset size scales

	1x	2x	5x	10x
ages	0.061	0.06	0.051	0.047
number interactions	0.0	0.0	0.0	0.0
number interactions (men only)	0.124	0.001	0.002	0.001
number interactions (women only)	0.27	0.021	0.259	0.26
total	0.027	0.013	0.012	0.017
usage	0.046	0.028	0.017	0.014
(ages, lengths)	2.527	1.984	1.908	1.8
usage/total	0.1	0.057	0.026	0.013

Table 4.28: Jensen Shannon Distance for Cohort OO with the gaussian-copula Model as the dataset size scales

	1x	2x	5x	10x
ages	1.872	1.997	1.442	1.582
number interactions	0.145	0.004	0.003	0.004
number interactions (men only)	0.14	0.006	0.005	0.005
number interactions (women only)	0.026	0.272	0.262	0.258
total	0.546	0.52	0.554	0.57
usage	0.236	0.225	0.227	0.225
(ages, lengths)	58.738	14.528	11.278	12.664
usage/total	0.385	0.16	0.119	0.114

Table 4.29: Jensen Shannon Distance for Cohort OO with the CTGAN Model as the dataset size scales

	1x	2x	5x	10x
ages	4.474	3.281	3.664	3.273
number interactions	0.125	0.134	0.001	0.001
number interactions (men only)	0.127	0.131	0.001	0.001
number interactions (women only)	0.332	0.303	0.356	0.818
total	0.108	0.1	0.067	0.039
usage	0.153	0.103	0.081	0.06
(ages, lengths)	49.762	49.469	50.579	50.685
usage/total	1.054	0.903	0.771	0.775

Table 4.30: Jensen Shannon Distance for Cohort OO with the copula GAN Model as the dataset size scales

	1x	2x	5x	10x
ages	0.471	0.64	0.642	0.641
number interactions	0.082	0.046	0.046	0.046
number interactions (men only)	0.249	0.048	0.047	0.047
number interactions (women only)	0.064	0.092	0.09	0.09
total	4.21	4.289	4.184	4.251
usage	1.109	1.055	1.053	1.058
(ages, lengths)	63.726	64.006	64.096	64.107
usage/total	0.776	0.73	0.728	0.729

Table 4.31: Jensen Shannon Distance for Cohort OO with the Transformer Model as the dataset size scales

Publication

The results in this chapter are based on the following publication.

- Joe Howie, Sowmya Balasubramanian, Jonas Bambi, Kenneth Moselle, Venkatesh Srinivasan, and Alex Thomo. Synthetic Generation of Patient Service Utilization Data: A Scalability Study. *Proceedings of the 34th Medical Informatics Europe Conference (MIE)*, 2024, pages 705–709.

Chapter 5

Conclusion and Future Work

In this dissertation, I have explored three distinct areas where advanced data analysis and machine learning techniques have the potential to significantly impact healthcare practices. Each study focuses on leveraging computational methods to address specific challenges and enhance diagnostic capabilities in different medical domains.

Firstly, investigation into thyroid disorder diagnosis emphasizes the importance of comprehensive feature analysis. The main objective in applying machine learning (ML) approaches to health data should be the usability of the results by the end users, the clinicians. With the help of a user study, [85] highlights the properties that clinicians look for in ML approaches such as clinical relevance, explainability, and clear specification of *the important features that helped the model make a decision*. The results in this work are geared towards meeting these requirements and gaining clinicians' trust. A rule-based ML algorithm, CART (considered transparent by clinicians), is applied to the KEEL thyroid dataset, providing clear evidence that measuring all the important features (as opposed to a select few) is key to efficiently diagnosing thyroid disease. Specifically, through careful data analysis in the settings of supervised and unsupervised learning, it is shown that testing TSH only can re-

sult in misdiagnosis but measuring the complete thyroid panel (FTI, TT4, TSH, and T3) is highly effective and recommended for medical practitioners. This result is of paramount importance to the diagnosis of thyroid disorder from a clinical perspective.

Secondly, our study on Autism Spectrum Disorder (ASD) diagnosis from fMRI data highlights the potential of tabular classifiers in achieving comparable performance to complex graph-theoretic methods. Early diagnosis of ASD or other related developmental disorders is crucial for providing individuals with the medical services and social support needed. Due to a lack of understanding about its causes and cure, new techniques are needed to improve diagnosis. This work investigates the potential of biomarkers obtained from fMRI scans in the diagnosis of ASD and ADHD. It shows that tabular classifiers can achieve performance comparable to the best-known graph-theoretic methods that are explainable. At the same time, it demonstrates the challenges in classifying brain networks using two similarity measures, Hamming distance and Jaccard similarity of max-cores on available datasets. Our results imply further research using larger fMRI datasets could alleviate the current challenges and thus lead to further progress.

Lastly, our exploration of synthetic data generation (SDG) methods for patient service utilization (PSU) data reveals significant advancements. This work compared state-of-the-art data-driven synthetic data generation methods for PSU data, exploring statistical, machine learning-based, and hybrid approaches, and making three key contributions. Firstly, the time taken to train each model and generate synthetic datasets of different sizes was studied, with Gaussian Copulas and Fast ML models demonstrating the fastest performance. Secondly, synthetically generated datasets were compared against their real counterparts by measuring the distance between various marginal distributions. Using Jensen-Shannon Divergence, it was shown that synthetic datasets are highly similar to real data, with divergences no larger than

2.5%. Joint distributions were also analyzed, revealing larger distances than marginal ones, but Gaussian Copulas and Fast ML Models still outperformed other models. Finally, a Binary RNN was used to validate the utility of the synthetic datasets by training the RNN on real data and predicting on synthetic datasets, and vice versa. In both cases, confusion matrix measures were all above 90%, with most in the high 90s. Overall, among the five models evaluated, Gaussian Copulas and Fast ML models emerged as the most preferred for generating PSU data.

To summarize, these studies highlight how advanced data analytics and machine learning can revolutionize healthcare by improving diagnostic accuracy, making medical data easier to interpret, and ensuring ethical use of synthetic data. These methodologies promise more effective, precise, and scalable healthcare solutions. As technology evolves, integrating these insights into clinical practice offers great potential for enhancing patient care and outcomes across various medical fields.

Many interesting open questions arise from this research. A key direction for future work is exploring feature correlations and testing new feature combinations to better understand their dependencies and impact on identifying important features for thyroid disease diagnosis. New and refined methodologies are needed to convert fMRI data to other forms of data that capture the subtle differences between the brain networks belonging to the two classes. With further research, it could be possible to identify the regions of interest in the brain relevant to a particular disorder and modify the imaging procedure to focus on those regions. In our work, we use thresholding to make the summary graphs unweighted. Alternatively, summary graphs could be treated as probabilistic graphs. This raises the question: can existing algorithms for computing k-cores in probabilistic graphs enhance the accuracy of our method?

The high-quality synthetic data generated can be used by the health authority to predict future demands on its various services and improve patient care in its region.

However, challenges persist in achieving the goal of releasing this data in the public domain. Further work is needed to alleviate privacy concerns so that the synthetic data can be made available for public use. More generally, it would be interesting to validate the findings of this work using new datasets on thyroid disease, autism spectrum disorder, and other related health conditions.

Bibliography

[1] Keel. https://sci2s.ugr.es/keel/dataset.php?cod=67.

[2] UCI - ANN Dataset is the source for keel. http://archive.ics.uci.edu/ml/datasets/Thyroid+Disease.

[3] Halim Abbas, Ford Garberson, Stuart Liu-Mayo, Eric Glover, and Dennis P Wall. Multi-modular ai approach to streamline autism diagnosis in young children. *Scientific reports*, 10(1):1–8, 2020.

[4] Carlo Abrate and Francesco Bonchi. Counterfactual graphs for explainable classification of brain networks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 2495–2504, 2021.

[5] A.H.Shahid, M.P.Singh, R.K.Raj, R.Suman, D.Jawaid, and M.Alam. A study on label tsh, t3, t4u, tt4, fti in hyperthyroidism and hypothyroidism using machine learning techniques. *Proceedings of the International Conference on Communication and Electronics Systems (ICCES)*, pages 930–933, 2019.

[6] APA. *Diagnostic and statistical manual of mental disorders: DSM-5™*. American Psychiatric Publishing, a division of American Psychiatric Association, Washington, DC, 5th edition, 2013.

[7] Mohammad R Arbabshirani, Sergey Plis, Jing Sui, and Vince D Calhoun. Single subject prediction of brain disorders in neuroimaging: Promises and pitfalls. *Neuroimage*, 145:137–165, 2017.

[8] Onur Asan, Alparslan Emrah Bayrak, Avishek Choudhury, et al. Artificial intelligence and human trust in healthcare: focus on clinicians. *Journal of medical Internet research*, 22(6):e15154, 2020.

[9] American Thyroid Association. Thyroid disease facts. https://www.thyroid.org/media-main/press-room/#:~:text=Up.

[10] Barbara Brody. The endocrine system and glands of the human body. https://www.webmd.com/diabetes/endocrine-system-facts/.

[11] Anna L Buczak, Steven Babin, and Linda Moniz. Data-driven approach for creating synthetic electronic medical records. *BMC medical informatics and decision making*, 10(1):1–28, 2010.

[12] Ed Bullmore and Olaf Sporns. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature reviews neuroscience*, 10(3):186–198, 2009.

[13] BC Guidelines Canada. Thyroid function tests. https://www2.gov.bc.ca/gov/content/health/practitioner-professional-resources/bc-guidelines/thyroid-testing.

[14] John D. Carmichael. Endocrinology: An integrated approach. https://www.ncbi.nlm.nih.gov/books/NBK28/.

[15] John D. Carmichael. Overview of the pituitary gland. https://www.merckmanuals.com/en-ca/home/hormonal-and-metabolic-disorders/pituitary-gland-disorders/overview-of-the-pituitary-gland.

[16] CDC. Autism data research. https://www.cdc.gov/autism/data-research/.

[17] Khushboo Chandel, Veenita Kunwar, Sai Sabitha, Tanupriya Choudhury, and Saurabh Mukherjee. A comparative study on thyroid disease detection using k-nearest neighbor and naive bayes classification techniques. *CSI transactions on ICT*, 4:313–319, 2016.

[18] Hui-Ling Chen, Bo Yang, Gang Wang, Jie Liu, Yi-Dong Chen, and Da-You Liu. A three-stage expert system based on support vector machines for thyroid disease diagnosis. *Journal of medical systems*, 36:1953–1963, 2012.

[19] Corinna Coupette, Sebastian Dalleiger, and Jilles Vreeken. Differentially describing groups of graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(4):3959–3967, 2022.

[20] Jessamyn Dahmen and Diane Cook. Synsys: A synthetic data generation system for healthcare applications. *Sensors*, 19(5):1181, 2019.

[21] Luis de la Torre-Ubieta, Hyejung Won, Jason L Stein, and Daniel H Geschwind. Advancing the understanding of autism disease mechanisms through genetics. *Nature medicine*, 22(4):345–361, 2016.

[22] Adriana Di Martino and Steward Mostofsky. ABIDE. http://fcon_1000.projects.nitrc.org/indi/abide/abide_I.html.

[23] Yanan Du, Shaofu Lin, and Zhisheng Huang. Generation of semantic patient data for depression. In *Health Information Science: 6th International Conference, HIS 2017, Moscow, Russia, October 7-9, 2017, Proceedings 6*, pages 102–112. Springer, 2017.

[24] IBM Cloud Education. Supervised learning. https://www.ibm.com/cloud/learn/supervised-learning.

[25] Khaled El Emam, Lucy Mosquera, Xi Fang, and Alaa El-Hussuna. Utility metrics for evaluating synthetic health data generation methods: validation study. *JMIR medical informatics*, 10(4):e35734, 2022.

[26] Charles Elkan. Evaluating classifiers. *San Diego: University of California*, 2012.

[27] Keanelek Enns, Venkatesh Srinivasan, and Alex Thomo. Identifying autism spectrum disorder using brain networks: Challenges and insights. *2023 14th International Conference on Information, Intelligence, Systems & Applications (IISA)*, pages 1–8, 2023.

[28] Peter G Enticott, Hayley A Kennedy, Nicole J Rinehart, Bruce J Tonge, John L Bradshaw, John R Taffe, Zafiris J Daskalakis, and Paul B Fitzgerald. Mirror neuron activity associated with social impairments but not age in autism spectrum disorder. *Biological psychiatry*, 71(5):427–433, 2012.

[29] Cristóbal Esteban, Stephanie L Hyland, and Gunnar Rätsch. Real-valued (medical) time series generation with recurrent conditional gans. *arXiv preprint arXiv:1706.02633*, 2017.

[30] Gerald Fischbach. Leo kanner's 1943 paper on autism. https://www.spectrumnews.org/opinion/viewpoint/leo-kanners-1943-paper-on-autism/.

[31] World Health Organization. Regional Office for the Eastern Mediterranean. Autism spectrum disorders. Technical documents, 2019.

[32] British Thyroid Foundation. Thyroid function tests. https://www.btf-thyroid.org/thyroid-function-tests.

[33] British Thyroid Foundation. What is thyroid disorder. https://www.btf-thyroid.org/what-is-thyroid-disorder.

[34] Andre Goncalves, Priyadip Ray, Braden Soper, Jennifer Stevens, Linda Coyle, and Ana Paula Sales. Generation and evaluation of synthetic patient data. *BMC medical research methodology*, 20:1–40, 2020.

[35] Leonardo Gutiérrez-Gómez and Jean-Charles Delvenne. Unsupervised network embeddings with node identity awareness. *Applied Network Science*, 4(1):1–21, 2019.

[36] Ali Hassan, Riza Sulaiman, Mansoor Abdulgabber, and Hasan Kahtan. Towards user-centric explanations for explainable models: A review. *Journal of Information System and Technology Management*, 6(22):36–50, 2021.

[37] Yoichi Hayashi, Satoshi Nakano, and Shota Fujisawa. Use of the recursive-rule extraction algorithm with continuous attributes to improve diagnostic accuracy in thyroid disease. *Informatics in Medicine Unlocked*, 1:1–8, 2015.

[38] D Hemalatha and S Poorani. Supervised machine learning models for classification of thyroid data. *International Journal of Scientific & Technology research*, 9, 2020.

[39] Leanna M Hernandez, Jeffrey D Rudie, Shulamite A Green, Susan Bookheimer, and Mirella Dapretto. Neural signatures of autism spectrum disorders: insights into brain network dynamics. *Neuropsychopharmacology*, 40(1):171–189, 2015.

[40] Jerome M. Hershman. Overview of the thyroid gland. https://www.merckmanuals.com/en-ca/home/hormonal-and-metabolic-disorders/thyroid-gland-disorders/overview-of-the-thyroid-gland.

[41] Kenji Hoshi, Junko Kawakami, Mitiko Kumagai, Sanae Kasahara, Noriaki Nishimura, Hitoshi Nakamura, and Kenichi Sato. An analysis of thyroid func-

tion diagnosis using Bayesian-type and SOM-type neural networks. *Chemical and pharmaceutical bulletin*, 53(12):1570–1574, 2005.

[42] Zhisheng Huang, Frank van Harmelen, Annette ten Teije, and Kathrin Dentler. Knowledge-based patient data generation. In *Proceedings of the International Workshop on Process-oriented Information Systems in Healthcare*, pages 83–96. Springer, 2013.

[43] MD Hye Rim Chung. Iodine and thyroid function. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4049553/.

[44] Susan L Hyman, Susan E Levy, Scott M Myers, Dennis Z Kuo, Susan Apkon, Lynn F Davidson, Kathryn A Ellerbeck, Jessica EA Foster, Garey H Noritz, Mary O'Connor Leppert, et al. Identification, evaluation, and management of children with autism spectrum disorder. *Pediatrics*, 145(1), 2020.

[45] Giuseppe Jurman, Roberto Visintainer, Michele Filosi, Samantha Riccadonna, and Cesare Furlanello. The HIM glocal metric and kernel for network comparison and classification. In *Proceedings of the IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–10, 2015.

[46] Dhamanpreet Kaur, Matthew Sobiesk, Shubham Patil, Jin Liu, Puran Bhagat, Amar Gupta, and Natasha Markuzon. Application of bayesian networks to generate synthetic health data. *Journal of the American Medical Informatics Association*, 28(4):801–811, 2021.

[47] Wissam Khaouid, Marina Barsky, Venkatesh Srinivasan, and Alex Thomo. K-core decomposition of large networks on a single PC. *Proceedings of the VLDB Endowment*, 9(1):13–23, 2015.

[48] Marjane Khodatars, Afshin Shoeibi, Delaram Sadeghi, Navid Ghaasemi, Mahboobeh Jafari, Parisa Moridian, Ali Khadem, Roohallah Alizadehsani, Assef Zare, Yinan Kong, et al. Deep learning for neuroimaging-based diagnosis and rehabilitation of autism spectrum disorder: a review. *Computers in biology and medicine*, 139:104949, 2021.

[49] Dohee Kim. The role of vitamin D in thyroid diseases. *International journal of molecular sciences*, 18(9):1949, 2017.

[50] Yazhou Kong, Jianliang Gao, Yunpei Xu, Yi Pan, Jianxin Wang, and Jin Liu. Classification of autism spectrum disorder by combining brain connectivity and deep neural network classifier. *Neurocomputing*, 324:63–68, 2019.

[51] Tommaso Lanciano, Francesco Bonchi, and Aristides Gionis. Explainable classification of brain networks via contrast subgraphs. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3308–3318, 2020.

[52] Xabat Larrea, Mikel Hernandez, Gorka Epelde, Andoni Beristain, Cristina Molina, Ane Alberdi, Debbie Rankin, Panagiotis Bamidis, and Evdokimos Konstantinidis. Synthetic subject generation with coupled coherent time series data. *Engineering Proceedings*, 18(1):7, 2022.

[53] Marlene Briciet Lauritsen. Autism spectrum disorders. *European child & adolescent psychiatry*, 22:37–42, 2013.

[54] Li-Na Li, Ji-Hong Ouyang, Hui-Ling Chen, and Da-You Liu. A computer aided diagnosis system for thyroid disease using extreme learning machine. *Journal of medical systems*, 36(5):3327–3337, 2012.

[55] Pantelis Linardatos, Vasilis Papastefanopoulos, and Sotiris Kotsiantis. Explainable AI: A review of machine learning interpretability methods. *Entropy*, 23(1), 2021.

[56] Da-You Liu, Hui-Ling Chen, Bo Yang, Xin-En Lv, Li-Na Li, and Jie Liu. Design of an enhanced fuzzy k-nearest neighbor classifier based computer aided diagnostic system for thyroid disease. *Journal of medical systems*, 36:3243–3254, 2012.

[57] Yaya Liu, Lingyu Xu, Jun Li, Jie Yu, and Xuan Yu. Attentional connectivity-based prediction of autism using heterogeneous rs-fMRI data from CC200 atlas. *Experimental neurobiology*, 29(1):27–37, 2020.

[58] Dr.Ananya Mandal. Autism causes. https://www.news-medical.net/health/Autism-Causes.aspx.

[59] Jose O. Maximo, Elyse J. Cadena, and Rajesh K. Kana. The implications of brain connectivity in the neuropsychology of autism. *Neuropsychology review*, 24(1):16–31, 2014.

[60] Scott McLachlan, Kudakwashe Dube, Thomas Gallagher, Jennifer A Simmonds, and Norman Fenton. Realistic synthetic data generation: The ATEN framework. In *Proceedings of the 11th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2018)*, pages 497–523, 2019.

[61] Yasir Iqbal Mir. Improved thyroid disease prediction model using data mining techniques with outlier detection. In *Advanced Machine Learning Approaches in Cancer Prognosis: Challenges and Applications*, pages 129–161. 2021.

[62] Yasir Iqbal Mir and Sonu Mittal. Thyroid disease prediction using hybrid machine learning techniques: An effective framework. *International Journal of Scientific & Technology Research*, 9(2), 2020.

[63] Muhammad Faiz Misman, Azurah A Samah, Farah Aqilah Ezudin, Hairuddin Abu Majid, Zuraini Ali Shah, Haslina Hashim, and Muhamad Harun. Classification of adults with autism spectrum disorder using deep neural network. In *Proceedings of the 1st International Conference on Artificial Intelligence and Data Sciences (AiDAS)*, pages 29–34, 2019.

[64] Sharmila Banerjee Mukherjee. Autism spectrum disorders—diagnosis and management. *The Indian Journal of Pediatrics*, 84:307–314, 2017.

[65] Hajra Murtaza, Musharif Ahmed, Naurin Farooq Khan, Ghulam Murtaza, Saad Zafar, and Ambreen Bano. Synthetic data generation: State of the art in health care domain. *Computer Science Review*, 48:100546, 2023.

[66] NHS. Attention deficit hyperactivity disorder (adhd). https://www.nhs.uk/conditions/attention-deficit-hyperactivity-disorder-adhd/.

[67] Giannis Nikolentzos, Polykarpos Meladianos, Stratis Limnios, and Michalis Vazirgiannis. A degeneracy framework for graph similarity. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 2595–2601, 2018.

[68] Hidir Selcuk Nogay and Hojjat Adeli. Machine learning (ML) for the diagnosis of autism spectrum disorder (ASD) using brain imaging. *Reviews in the neurosciences*, 31(8):825–841, 2020.

[69] Thyroid Foundation of Canada. Prevalence of thyroid diseases in canada: An analysis. https://www.thyforlife.com/prevalence-of-thyroid-diseases/.

[70] Qiao Pan, Yuanyuan Zhang, Min Zuo, Lan Xiang, and Dehua Chen. Improved ensemble classification method of thyroid disease based on random forest. In *Proceedings of the 8th International Conference on Information Technology in Medicine and Education (ITME)*, pages 567–571, 2016.

[71] Alan Perotti, Paolo Bajardi, Francesco Bonchi, and André Panisson. Graphshap: Motif-based explanations for black-box graph classifiers. *arXiv preprint arXiv:2202.08815*, 2022.

[72] Esteban Piacentino, Alvaro Guarner, and Cecilio Angulo. Generating synthetic ECGs using GANs for anonymizing healthcare data. *Electronics*, 10(4):389, 2021.

[73] Nataša Pržulj. Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23(2):e177–e183, 2007.

[74] Shaik Razia, P Swathi Prathyusha, N Vamsi Krishna, and N Sathya Sumana. A comparative study of machine learning algorithms on thyroid disease prediction. *International Journal of Engineering and Technology*, 7(2.8):315–319, 2018.

[75] David Riaño and Alberto Fernández-Pérez. Simulation-based episodes of care data synthetization for chronic disease patients. In *Knowledge Representation for Health Care: HEC 2016 International Joint Workshop, KR4HC/ProHealth 2016*, pages 36–50. Springer, 2017.

[76] Fatemeh Saiti, Afsaneh Alavi Naini, Mahdi Aliyari Shoorehdeli, and Mohammad Teshnehlab. Thyroid disease diagnosis based on genetic algorithms using PNN and SVM. In *Proceedings of the 3rd IEEE International Conference on Bioinformatics and Biomedical Engineering*, pages 1–4, 2009.

[77] Caio Pinheiro Santana, Emerson Assis de Carvalho, Igor Duarte Rodrigues, Guilherme Sousa Bastos, Adler Diniz de Souza, and Lucelmo Lacerda de Brito. rs-

fMRI and machine learning for ASD diagnosis: a systematic review and meta-analysis. *Scientific reports*, 12(1):6030–6030, 2022.

[78] W Schiffmann, M Joost, and R Werner. Synthesis and performance analysis of multilayer neural network architectures. *Koblenz: Institute of Physics*, 1992.

[79] W Schiffmann, M Joost, and R Werner. Optimization of the backpropagation algorithm for training multilayer perceptrons. *Koblenz: Institute of Physics*, 1994.

[80] Rittika Shamsuddin, Barbara M Maweu, Ming Li, and Balakrishnan Prabhakaran. Virtual patient model: an approach for generating synthetic healthcare time series data. In *Proceedings of the IEEE International Conference on Healthcare Informatics (ICHI)*, pages 208–218, 2018.

[81] Peter K Sharpe, Helge E Solberg, Kjell Rootwelt, and Michael Yearworth. Artificial neural networks in diagnosis of thyroid function from in vitro laboratory tests. *Clinical chemistry*, 39(11):2248–2253, 1993.

[82] Faria Zarin Subah, Kaushik Deb, Pranab Kumar Dhar, and Takeshi Koshiba. A deep learning approach to predict autism spectrum disorder using multisite resting-state fmri. *Applied Sciences*, 11(8), 2021.

[83] Feyzullah Temurtas. A comparative study on thyroid disease diagnosis using neural networks. *Expert Systems with Applications*, 36(1):944–949, 2009.

[84] Rajat Mani Thomas, Selene Gallo, Leonardo Cerliani, Paul Zhutovsky, Ahmed El-Gazzar, and Guido van Wingen. Classifying autism spectrum disorder using the temporal statistics of resting-state functional mri data with 3d convolutional neural networks. *Frontiers in Psychiatry*, 11, 2020.

[85] Sana Tonekaboni, Shalmali Joshi, Melissa D McCradden, and Anna Goldenberg. What clinicians want: contextualizing explainable machine learning for clinical end use. In *Machine learning for healthcare conference*, pages 359–380, 2019.

[86] Amirsina Torfi and Edward A Fox. CorGAN: Correlation-capturing convolutional generative adversarial networks for generating synthetic healthcare records. *arXiv preprint arXiv:2001.09346*, 2020.

[87] Amirsina Torfi, Edward A Fox, and Chandan K Reddy. Differentially private synthetic medical data generation using convolutional GANs. *Information Sciences*, 586:485–500, 2022.

[88] Ebru Turanoglu-Bekar, Gozde Ulutagay, and Suzan Kantarcı-Savas. Classification of thyroid disease by using data mining models: a comparison of decision tree algorithms. *Oxford Journal of Intelligent Decision and Data Sciences*, 2:13–28, 2016.

[89] StackExchange Network Cross Validated. Using principal component analysis (pca) for feature selection. https://stats.stackexchange.com/questions/27300/using-principal-component-analysis-pca-for-feature-selection/27310#27310.

[90] Jason Walonoski, Mark Kramer, Joseph Nichols, Andre Quina, Chris Moesel, Dylan Hall, Carlton Duffett, Kudakwashe Dube, Thomas Gallagher, and Scott McLachlan. Synthea: An approach, method, and software mechanism for generating synthetic patients and the synthetic electronic health care record. *Journal of the American Medical Informatics Association*, 25(3):230–238, 2018.

[91] Zhenchen Wang, Puja Myles, and Allan Tucker. Generating and evaluating synthetic uk primary care data: preserving data utility & patient privacy. In *Pro-*

ceedings of the 32nd IEEE International Symposium on Computer-Based Medical Systems (CBMS), pages 126–131, 2019.

[92] Zhenchen Wang, Puja Myles, and Allan Tucker. Generating and evaluating cross-sectional synthetic electronic healthcare data: preserving data utility and patient privacy. *Computational Intelligence*, 37(2):819–851, 2021.

[93] Choong-Wan Woo, Luke J Chang, Martin A Lindquist, and Tor D Wager. Building better biomarkers: brain models in translational neuroimaging. *Nature neuroscience*, 20(3):365–377, 2017.

[94] Fan Yang, Zhongping Yu, Yunfan Liang, Xiaolu Gan, Kaibiao Lin, Quan Zou, and Yifeng Zeng. Grouped correlational generative adversarial networks for discrete electronic health records. In *Proceedings of the International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 906–913, 2019.

[95] Guang Yang, Qinghao Ye, and Jun Xia. Unbox the black-box for the medical explainable ai via multi-modal and multi-centre data fusion: A mini-review, two showcases and beyond. *Information Fusion*, 77:29–52, 2022.

[96] Jae-Seung Yun, Jung Min Bae, Ki-Jo Kim, Yu Seok Jung, Gyong Moon Kim, Hyung-Rae Kim, Jun-Seok Lee, Seung-Hyun Ko, Seon-Ah Cha, and Yu-Bae Ahn. Increased risk of thyroid diseases in patients with systemic lupus erythematosus: A nationwide population-based study in korea. *PloS one*, 12(6):e0179088, 2017.

[97] Guoqiang Peter Zhang and Victor L Berardi. An investigation of neural networks in thyroid function diagnosis. *Health Care Management Science*, 1(1):29–37, 1998.