Master's Thesis

Predicting Thrombosis with Machine Learning

Khurshid Abbas

Master in Applied Computer Science

Faculty of Computer Sciences
Østfold University College
Halden
May 27, 2021

Master's Thesis

Predicting Thrombosis with Machine Learning

Khurshid Abbas

A thesis presented for the degree of
Master in Applied Computer Science

Faculty of Computer Sciences
Østfold University College
Halden
May 27, 2021

Abstract

This study presents a comparison of the performance of standard machine learning techniques in predicting thrombosis. The comparison is conducted on full and reduced variations of a clinical dataset.
The investigation demonstrates that XGBoost accomplishes the highest efficiency on the full dataset. It is followed by Random Forests, Support Vector Machines and Artificial Neural Networks. They score an accuracy of 94.56, 92.74, 92.14 and 88.82, respectively.
Random Forests yields the maximum performance on the reduced dataset, followed by XGBoost, Artificial Neural Networks and Support Vector Machines. They score an accuracy of 86.10, 84.29, 81.87 and 77.94, respectively.
Support Vector Machines produces the lowest number of false negatives on the full dataset followed by XGBoost, Artificial Neural Networks and Random Forests. They attain a recall score of 96.72, 93.44, 80.32 and 75.40, respectively.
XGBoost attains the best performance on the reduced dataset followed by Support Vector Machines, Random Forests and Artificial Neural Networks. They score 50.81, 49.18, 40.98 and 34.42, respectively, on the recall metric.

Keywords: Quantitative Comparison, Thrombosis Prediction, Decision Support, Random Forests, XGBoost, Support Vector Machines, Artificial Neural Networks.

Acknowledgements

I'm grateful to Dr. Lars Vidar Magnusson for his continued assistance, guidance and supervision. I appreciate his contributive and supportive presence that ensured an excellent experience throughout my master's thesis.
I'm thankful to Ruslan Sorano for his valuable assistance.

I'm also grateful to Dr. Monica Kristiansen Holone for her kindness and support.

Ultimately, I'm thankful to my parents and siblings for their unconditional love and enduring encouragement.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Modern society is experiencing an advancing wave of digitalisation and, computing technologies are being adopted into various aspects of present-day society. Digitalisation has introduced automated computing systems in travel, education, public administration, communication, hospitality, economic, healthcare and several more divisions of the social construct. The tendency towards digitalisation is generating enormous amounts of digital data. This untapped data is being utilised to benefit from the computing technologies such as machine learning. Machine learning models trained with this data are offering vital decision support to enhance the services in various areas of society, including healthcare.

1.1 Machine Learning in Health Care

The digitalisation of the healthcare sector has enabled the collection of vital clinical data. This increased availability of data has enabled the application of data analytics in the field. The clinical data is employed to train machine learning models that provide decision support in healthcare. Machine learning-based decision support is beneficial to medical specialists in diagnosing and curing various diseases, therefore research institutes and experts are conducting numerous studies in order to benefit from this opportunity.

Østfold Hospital Kalnes has conducted a similar study in collaboration with its international partners to investigate thrombosis. The research initiative investigates thrombosis patients and the potential application of machine learning in the domain. Thrombosis is a fatal ailment that is caused by the abnormal formation of thrombi inside blood vessels. It has been discussed thoroughly in the *Background* chapter. The study has generated a comprehensive dataset that describes thrombosis patients. Our study performs three investigations on this dataset. It examines the full and reduced variations of the dataset with the help of standard machine learning algorithms. It additionally analyses the false negatives produced during the predictive modelling on the dataset.

1.2 Research Objectives

This study aims to conduct a comparative investigation of the performance of standard machine learning algorithms. It will compare the performance of the algorithms on different variations of the dataset in predicting thrombosis. The dataset has been provided by the investigation conducted at Østfold Hospital Kalnes. It will employ two variations of the original dataset to compare the performance of standard machine learning algorithms. The full dataset variation includes all of the feasible features from the original dataset and, the reduced dataset variation contains a limited amount of patients' information. A thorough explanation of these variations has been presented in the *Experimental Setup* chapter.
It is particularly significant to identify a maximum number of patients from the mixed population to ensure that the subjects with the ailment are not disregarded. This necessitates that the predictive models should possess high scores on the recall metric. A high score on the recall metric ensures that the number of false negatives is minimised, therefore, this study additionally compares the performance of standard machine learning algorithms on reducing the number of false negatives.

The principal objectives of this study can be summarised with the help of the following research questions.

- **RQ I:** How do standard machine learning algorithms compare in performance on the full dataset?
- **RQ II:** How do standard machine learning algorithms compare in performance on the reduced dataset?
- **RQ III:** How do standard machine learning algorithms compare in performance on minimising the number of false negatives?

This study is structured into the following chapters. The *Introduction* chapter discusses the motivation and the goals of the study. The *Background* chapter presents a thorough explanation regarding thrombosis and machine learning algorithms. The *Related Work* chapter contains a comprehensive literature review concerning the investigation. A detailed description of the experiments has been provided in the *Experimental Setup* chapter. The *Results* chapter illustrates the outcomes obtained from the experiments, and the *Discussion* chapter reviews these outcomes. The final outcomes of the investigation are presented in the *Conclusion* chapter.

Chapter 2

Background

This chapter presents a comprehensive overview of thrombosis and machine learning.

2.1 Thrombosis

Thrombosis is the development of blood thrombi inside blood vessels. The formation of blood clots generally occurs due to the injuries sustained by blood vessels. The immune system of the human body initiates a healing process to stop the bleeding. The platelets and fibrin accumulate on the injured patches of the blood vessels to form blood clots. This blood clots formation (coagulation) can occasionally occur even when there is no injury to synthesise an embolus. Several serious health complications can occur if the embolus detaches from the blood vessel and travels to other vital organs such as the lungs [1]. There are two principal types of thrombosis, venous and arterial thrombosis. Venous thrombosis is relatively more prevalent than arterial thrombosis, and it can cause blockage of blood flow to different parts of the body. Pulmonary thrombosis is a dangerous medical complication that arises when venous thromboembolism migrates to the lungs. Arterial thrombosis may deprive the organs of oxygen-rich blood which can result in a stroke or general organ damage [1].
The inner membranes of blood vessels and the heart are in continuous contact with the blood due to the blood circulation. The coagulant characteristics of the blood can lead to the formation of clotting elements on the interior portions of the blood-rich organs, i.e. heart and blood vessels. These harmful clots are generally known as thrombus. A thrombus is usually formed by the accumulation of platelets, insoluble fibrin, red blood cells and white blood cells. Thrombosis can be further categorised into the white, red, mixed and transparent thrombus depending on the situation and characteristics of the thrombus [1].

2.1.1 Thrombotic Hazards

Thrombosis can have lethal consequences for human health. A study concluded that venous thrombosis may have an early case fatality rate of up to 6%, and 20% of patients can die within the first year of illness [2]. The fatalities may increase further since the recurrence rate can be as high as 5% [3]. Thrombosis often concludes with strokes, therefore, making it one of the most deadly ailments in the United States [4]. The mortality rate is interestingly maximum during the first few months of the illness and declines gradually over time [2].
Thrombosis is a lethal ailment that contributes significantly to fatalities around the world. The recurring nature of the disease aggravates the risk further that can jeopardise public health. A study conducted on 570 patients in Cambridge, UK concluded that the average rate of recurrence stands at 11%. The recurrence rate after the first unprovoked thrombosis was the highest at 19.4% and lowest after first provoked thrombosis at 0% [5]. A few more clinical studies concluded with high recurrence rates. The recurrence rates are significantly higher in men than in women, and they can be as high as 30.7% [6]. The incidence of thrombosis for the first time is contrarily higher in women than men. The diverging recurrence rates are not precisely justified [6][7].

2.1.2 Risk Factors

There is a wide range of risk factors that can lead to complications concerning thrombosis. They can be classified into two major categories, unprovoked and provoked. The unprovoked category consists of older age (>65 years), long haul travel, thrombophilia, obesity, smoking, hypertension and air pollution. The provoked category consists of little physical movement, trauma, oral contraceptives, cancer, critical sickness (pneumonia, heart failure) [8].
There are several other causes linked to the occurrence of thrombosis that includes genetic disorders, environmental factors, medical procedures [3]. Gender can affect the prevalence of thrombosis. A study performed in Tromsø, Norway demonstrated that the thrombosis incidence rate is higher in women until age 60. After the age of 60, the incidence rate is slightly higher in men than in women [2]. A study conducted in Sweden concluded that the incidence rates are similar for both males and females [9]. Age is an influential factor in general as well. The incidence rate increases steeply with the increasing age [3]. Ethnic backgrounds can influence the incidence of thrombosis to a significant extent. Several studies have concluded that the incidence rate can be lower in the Asian populations than in European counterparts [3]. National health insurance claims data from Taiwan presents an incidence rate that is ten times lower than the American and European incidence rates [10]. Several genetic risk factors can be contributive to the incidence of thrombosis. The deficiency of natural anticoagulants protein S, protein C and antithrombin are prominent genetic risk factors [3]. The acquired risk factors can be the leading reason for the incidence of thrombosis. There are various acquired risk factors, including, but not limited to, medical procedures, cancer, drugs, obesity, smoking, long-haul

travel and lack of exercise [3]. Cancer is one of the most influential acquired risk factors, raising the risk of venous thrombosis by 50 times during the first six months after the diagnosis [11].

2.1.3 Diagnosis

Thrombosis can be diagnosed with the help of several diagnosis techniques. Clinical probability assessment, Measurement of fibrin D-Dimer and Compression ultrasonography constitute the bulk of thrombosis diagnosis infrastructure [8].

Clinical probability assessment segregates the patients based on the associated high and low clinical probability. Patients with high clinical probability are treated with anticoagulants until the diagnosis is concluded with the help of Compression ultrasonography. The diagnosis for patients with low or intermediate clinical probability can be ruled out with the aid of a D-Dimer test [8].

Fibrin D-Dimer is a degradation product whose concentration rises for patients with venous thrombosis. D-Dimer test is remarkably efficient (95%) in classifying venous thrombosis patients. 500 μg/L is generally accepted as the standard threshold [8].

Compression ultrasonography proposes three different approaches. Only proximal veins that are above the calf are subject to the diagnosis during the first approach. This approach has a low yield rate, therefore, less effective. Proximal and distal veins are diagnosed with Compression ultrasonography during the second approach. This approach may initiate anticoagulation in some patients that can lead to bleeding. The third approach is administered by conducting single Compression ultrasonography on proximal veins. Patients with an intermediate or low clinical probability can be declared healthy if the results are negative. The patients with a high clinical probability go through further imaging [8].

2.1.4 Treatment

Thrombosis patients are generally treated with the help of Anticoagulant, Antiplatelet and Thrombolytic therapies [1].

The most fundamental treatment therapy is anticoagulation and, the majority of patients react positively to the treatment [8]. The thrombus consists of blood cells and fibrin as described earlier. They are found throughout the human body, especially in blood vessels and the heart. The coagulation system and platelets play a pivotal role in the development of thrombosis. Thrombin produces fibrin that forms blood clots together with platelets. Anticoagulant therapy targets the dangerous combination of platelets and the coagulation system [1]. Anticoagulant therapy may cause bleeding during the initial phases of the treatment, particularly in the older age. It can be avoided by halting alcohol consumption, personal care and withdrawing the use of concurrent drugs [8]. General anticoagulants include Heparins, Hua Falin and Fondaparinux [1][8].

Antiplatelet therapy is centred towards the platelets as the name suggests and it has proven to be effective. Platelets do not normally aggregate inside blood vessels to form clots. When blood vessels are injured, platelets can interact with the recently exposed collagen. It leads to the aggregation of platelets, therefore, blood clots. It is possible to control the aggregation of platelets with certain drugs to restrain thrombosis. Antiplatelet drugs decrease the adhesive characteristics of platelets to restrain the excessive aggregation of platelets inside blood vessels [1][12].

Thrombolytic therapy is another excellent approach to thrombosis treatment. Thrombolytic drugs produce fibrinolytic enzyme from plasminogen. It disintegrates the fibrin inside a thrombus to dissolve it. Reteplase, Alteplase, Streptokinase and Senecteplase are prominent thrombolytic drugs [1][13].

2.2 Machine Learning

We are going through the age of digitalisation. We are using computers and smartphones to assist ourselves in our daily lives which have digitalised the business processes. It includes but not limited to education, healthcare, shopping, personal communication and information sharing. While we use computers to accomplish various goals, it leads to the generation of valuable data. The creation of useful data has grown exponentially with the advent of smartphones. Naturally generated information generally comprises patterns and regularities. These patterns and regularities play a pivotal role in predictive modelling. We utilise specific computer algorithms to accommodate a computer in identifying these patterns. Computers can predict the outcomes of future incidents with the help of these predictive models once these patterns have been identified. For instance IBM Watson has been successful in recommending cancer treatments with an accuracy of up to 99% [14]. The accuracy and the reliability of machine learning models can be enhanced further with a continuous supply of quality data [14].

Machine learning is the process of training computers to predict future outcomes by gaining valuable knowledge from historical experience and information. Machine learning can be classified into three principal types, supervised, unsupervised and reinforcement learning [15].

2.2.1 Supervised Learning

Supervised learning is one of the most prominent types of machine learning. Supervised learning has a strong focus on labelled data. The machine learning algorithm is trained with the help of a training dataset. The training dataset contains training examples and each example comprises the inputs and the desired output. The algorithm iterates over a large number of samples and attempts to learn the patterns with the help of a supervisor. As the algorithm is provided with the correct output values, therefore, it can learn from the mistakes while attempting to learn the relationship between the input and the output. Once the model has processed enough examples to learn a sound mapping from the input to the output, it is evaluated against the test dataset. The accuracy of models in supervised learning is highly dependent on the number and quality of training examples. Classification and Regression are the leading types of supervised learning [15]. Figure 2.1 demonstrates the overall concept of supervised learning.

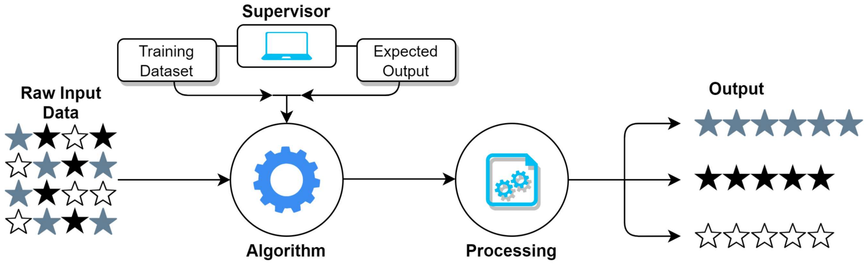


Figure 2.1: A visual representation of supervised learning in machine learning.

2.2.2 Unsupervised Learning

Unsupervised learning is principally associated with unlabelled data. It operates on a dataset that consists of only inputs and no specified outputs. The algorithm processes the bulk of data and discovers the hidden patterns and regularities. The groups or clusters identified by the algorithm are utilised to make informed decisions. The process has been illustrated in Figure 2.2. Unsupervised learning is particularly beneficial since a greater portion of the data available for processing is unlabelled. It is specifically helpful in discovering hidden but remarkably valuable patterns in unstructured data. Clustering is an excellent example of unsupervised learning [15].

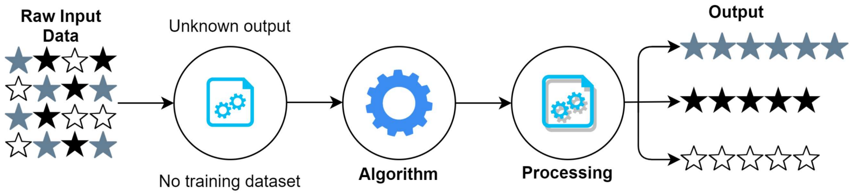


Figure 2.2: A visualisation of unsupervised learning in machine learning.

2.2.3 Reinforcement Learning

Reinforcement learning is moderately unrelated to supervised and unsupervised learning. It is employed when the aim is not a single entity but a set of accurate actions to achieve the goal (state). As the name suggests, a reinforcement learning algorithm is designed to learn from mistakes. There is naturally a higher frequency of errors during the initial phase of learning. The algorithm (agent) is continuously informed regarding the “good” and “bad” decisions, therefore, reinforced to make the correct decisions. The accuracy improves together with the process of learning. It has been described in Figure 2.3. Reinforcement learning is relatively more sensitive to unreliable inputs because it might lead to incorrect reinforcements. Computers learning to play video games against humans is a well-known example of reinforcement learning [15].

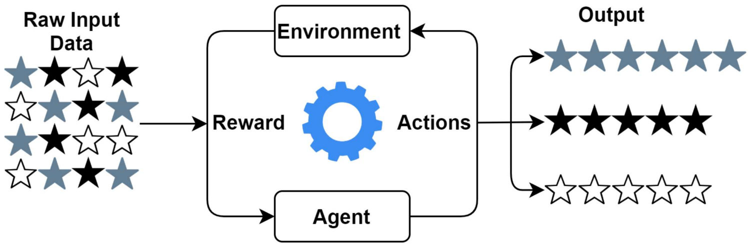


Figure 2.3: A visual representation of reinforcement learning in machine learning.

2.3 Machine Learning Algorithms

There are several algorithms employed in machine learning for the learning process, including, but not limited to, linear regression, logistic regression, decision trees, random forest, XGBoost and artificial neural networks.

2.3.1 Linear Regression

Regression is associated with the prediction of continuous numerals from independent variables. It principally relies on the cause and effect relationship between the variables and belongs to the supervised learning branch of machine learning. Linear regression is one of the many varieties of regressions where there is only one predictor variable. As the name suggests, there is a linear relationship between the predictor variable and the target variable or the dependent variable. This relationship can be expressed in the form of a linear equation. The equation consists of the independent variable, dependent variable and some parameters. These parameters are adjusted during the learning process in order to develop an equation that generates the best fitting line on all of the data points. It has been demonstrated graphically in Figure 2.4. It is possible to utilise multiple independent variables by employing Multiple Linear Regression [16].

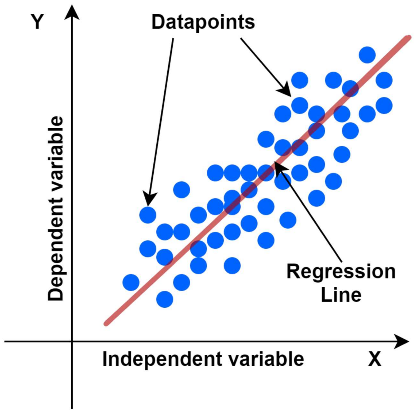


Figure 2.4: The process of fitting a regression line to the data-points using Linear Regression.

2.3.2 Logistic Regression

Logistic regression is a machine learning algorithm to perform classification. It is a supervised learning technique. Classification in machine learning is the process of differentiating the dataset into different groups or classes. Logistic regression is usually employed to perform binary classification in which the dataset is categorised into two categories. Logistic regression is influenced by the concept of probability and a sigmoid function is used to perform the cost analysis. The sigmoid function converts any input value into a value that lies between 0 and 1. A threshold is established during the learning process to differentiate the dataset into two classes depending on the probability output. Figure 2.5 presents a visual representation of logistic regression. Similar to linear regression, gradient descent is utilised to depreciate the cost estimation. Logistic regression is quite famous for its simplistic nature, however, other complex algorithms may outperform it easily [16].

2.3.3 Decision Trees

Decision trees are an excellent, supervised machine learning technique. They can be helpful in classification and regression. As the name suggests, these algorithms process the dataset to build decision trees. Decision trees consist of several if-else statements to perform decisions or predictions. The tree itself consists of the root node, decision nodes, branches and leaf nodes. The nodes where the tree splits into two branches are called decision nodes. The root node is the first decision node in the tree. Leaf nodes do not split further into branches, as illustrated in Figure 2.6. The position of the nodes and the splits are governed by the measure of impurity. Decision trees use entropy and Gini index as measures of impurity. Classification and Regression Tree (CART) is

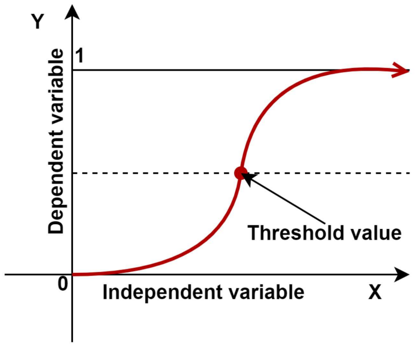


Figure 2.5: A visualisation of classification performed by Logistic Regression.

one of the most famous techniques to build decision trees. CART is easy to utilise and, it can handle both numerical and categorical data. Overfitting has always been a problem with decision trees since they are sensitive to the noise [16].

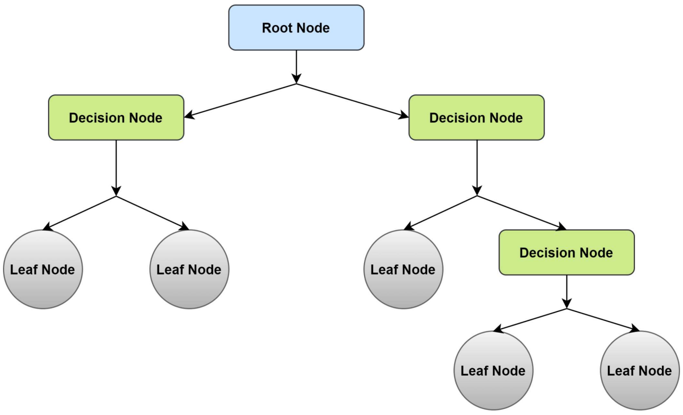


Figure 2.6: An illustration of a simple Decision Tree extending from a Root node to Leaf nodes.

2.3.4 Random Forests, Gradient Boosting

Random Forests is a supervised machine learning algorithm. This algorithm is constructed on the concept of "wisdom of crowds" that suggests, multiple models can predict better than a single model. Multiple decision trees are combined to generate an ensemble that provides a collective prediction. During the training process, each decision tree is trained on a random number of predictors and dataset samples. Randomness is introduced into the process to decorrelate the

trees included in a random forest which enhances the accuracy to a significant level. The principal parameter for Random Forests is “mtry”. It is the amount of randomly chosen predictors that are available at a specific split during the tree formation process. Random Forests can be utilised for classification and regression, therefore, ensembles are built out of classification trees or regressions trees accordingly [16]. Figure 2.7 illustrates a fundamental Random Forests configuration.
Gradient boosting is closely related to Random Forests. The models/trees are built sequentially while minimising the errors. It is accomplished by extending the influence of high-performance models. Gradient boosting employs gradient descent to minimise the anomalies in the contributing models. Gradient boosting has now evolved into Extreme gradient boosting that improves the performance further [16].

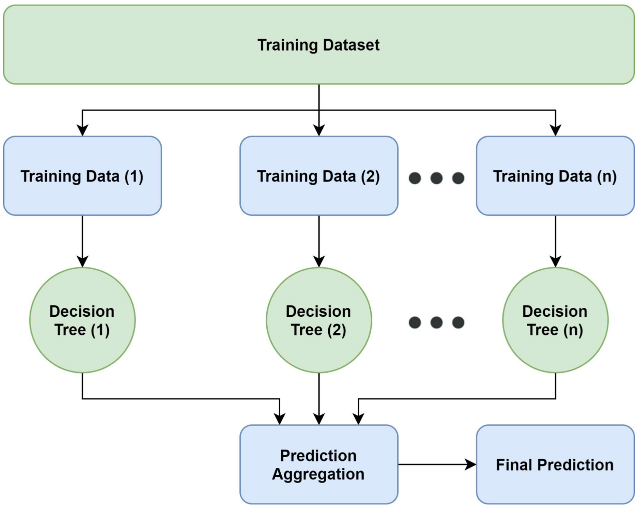


Figure 2.7: An illustration of the basic structure of the Random Forests algorithm.

2.3.5 Artificial Neural Networks

Artificial Neural Networks (ANNs) are one of the most commended among machine learning algorithms. They outperform decision tree-based algorithms readily where unstructured data is involved e.g. videos, images, text. ANNs belong to both supervised and unsupervised learning since they can process virtually all sorts of data, however, they are exceptionally helpful in discovering hidden patterns. ANNs find their applications in reinforcement learning as well. ANNs endeavour to imitate the biological brains. A typical ANN consists of an input layer, one or more hidden layers and an output layer. Each layer consists of several nodes that mimic neurons in a biological brain. These nodes are connected to one another with the help of edges, therefore, the layers of nodes are connected. An edge possesses a specific weight (importance) which is tuned during the training process. The node/neuron receives inputs and produces a single output which is forwarded to another node. A node produces the output by calculating the weighted sum of all inputs and then adding a bias to the sum, the final output is generated by applying an activation function to the final sum [16]. A fundamental neural network consisting of an input layer, a hidden layer and the output layer has been demonstrated in Figure 2.8.

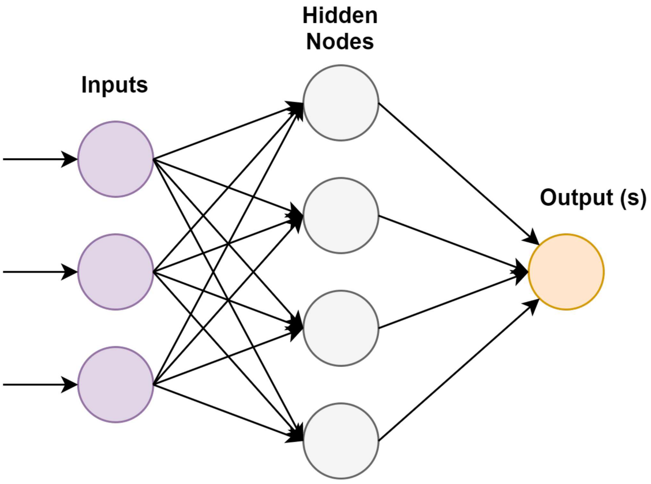


Figure 2.8: The representation of an Artificial Neural Network with singular input, hidden and output layers.

Chapter 3

Related Work

This chapter presents a thorough literature survey regarding the utilisation of machine learning in thrombosis diagnosis.

Artificial intelligence tools are growing in terms of reliability and accuracy due to the availability of adequate data and extensive processing power. The application of such tools inside the healthcare sector is gaining pace, however, the particular utilisation of artificial intelligence on thrombosis is reasonably confined. There are a few studies that have addressed the application of artificial intelligence tools to cure or diagnose thrombosis.

3.1 Literature Survey

A systematic literature survey was conducted to discover potentially relevant studies. The databases incorporated in the primary literature survey consisted of ACM Digital Library, IEEE Xplore, Elsevier and Google Scholar. The academic information covering the predictive analysis aspect of thrombosis is moderately limited. The survey was, therefore, conducted with comparatively lesser strict filters. The literature survey reviewed various topics related to this study. It included the preliminary introduction to thrombosis, conventional treatment techniques for thrombosis and the general application of machine learning in healthcare, particularly concerning thrombosis.

3.2 Machine Learning and Thrombosis

This section presents a number of studies that have investigated the application of machine learning for the diagnosis and treatment of thrombosis.

Database	Search Keywords	Search Fields	No. of Matches
ACM Digital Library	thrombosis, machine, learning, artificial, intelligence	title, abstract	≈ 5
IEEE Xplore	thrombosis, machine, learning, artificial, intelligence	title, abstract	≈ 7
Google Scholar	thrombosis, machine, learning, artificial, intelligence	full text	≈ 7
Elsevier - ScienceDirect	thrombosis, machine, learning, artificial, intelligence	full text	≈ 5

Table 3.1: A concluding summary of the literature review.

Machine Learning to Predict Venous Thrombosis in Acutely Ill Medical Patients

An investigation conducted by Nafee et al. [17] has assessed the performance of machine learning models against the IMPROVE (International Medical Prevention Registry on Venous Thromboembolism) score. IMPROVE is an integer-based risk assessment model that is used to identify high-risk thrombosis patients. It completes the risk assessment by systematically analysing the thrombosis risk factors. The data for analysis originated from a phase 3 clinical trial that comprised 7513 patients. It illustrated the personal, ethnic, clinical and historical treatment aspects of the patients. The patients included in the trial were hospitalised for critical illness. A total of 39 machine learning models were built which included Random Forests, Extreme Gradient Boosting, Classification Trees and Bayesian Logistic Regression-based models. The predictions of different models were combined with the help of the Ensemble Learning method to generate a more accurate and unified prediction. The weights for the candidate models were determined with the help of cross-validation. In total two super learners were developed depending on the number of predictors utilised for the learning process. IMPROVE score was calculated for each patient and, the performance was compared to the machine learning models. The c-statistic score for the IMPROVE method stood at 0.59 as compared to the machine learning models, which scored 0.69 and 0.68, therefore, outperformed the IMPROVE score. It has been illustrated in Figure 3.1. To summarise, the machine learning models performed better than the integer-based IMPROVE score in predicting thrombosis among critically ailing patients [17].

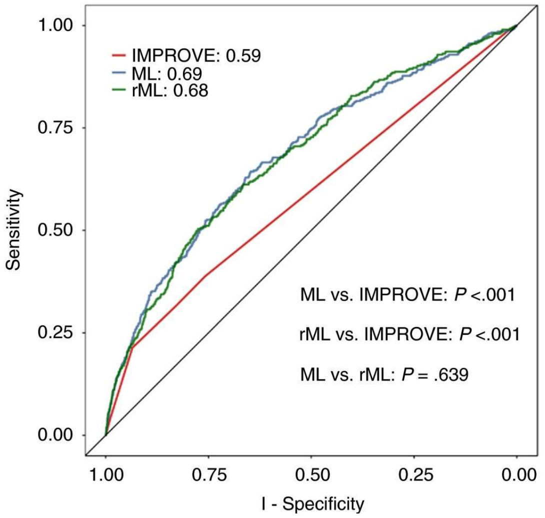


Figure 3.1: The comparison of the performance of machine learning models and IMPROVE score [17].

Machine Learning Approaches for Risk Assessment of Peripherally Inserted Central Catheter-Related Vein Thrombosis in Hospitalised Patients with Cancer

Liu et al. [18] also conducted a study on hospitalised cancer patients. The objective of the study was to predict the future incidence of peripherally inserted central venous catheter (PICC) thrombosis in cancer patients with the help of machine learning. The introduction of PICCs is prevalent among cancer patients who required frequent chemotherapy sessions. PICC aided treatment can provoke thrombosis formation among the patients without explicit prior symptoms, which may endanger the patient's life. It is possible to evade critical thrombosis formation by administering anticoagulants, but it can lead to bleeding that can be dangerous for cancer patients. Hence timely identification of high-risk patients can aid medical experts in conducting effective treatment. A total of 348 patients were admitted to the study. The patients were monitored for 30 days after the introduction of PICC. During the monitoring period, patients were continuously diagnosed for PICC related thrombosis using colour Doppler ultrasonography. The attributes collected for the training purpose included the patient's clinical details, medical history, patient's family's thrombosis record, patient's diet, genetic information, demographic circumstances, cancer treatment details and the inserted catheter data. The study implemented Random Forests and Least Absolute Shrinkage and Selection Operator (LASSO) to build models with high accuracy. LASSO was introduced to perform the predictor selection and, RF was utilised to classify the patients with a high risk of PICC induced thrombosis. There was a 50%-50% random split between the training and the testing data. Several different models were subsequently built by forming different combinations of RF, LASSO and Seeley. Seely is a PICC-thrombosis assessment criterion that has been widely utilised across the field. The RF models achieved the best performance in identifying high-risk patients as compared to the Seeley criterion that performed the worst by identifying all patients as negative. Machine learning models achieved scores of 0.7733, 0.7869, 0.7833 and 0.7717 as compared to the Seely criterion that scored 0.5 on the AUC scale. The performance of different models has been compared in Figure 3.2. AUC (area under the curve) is the measure of the performance of a classifier in classifying different classes. The accuracy of a model is directly proportional to the AUC score. The study concluded by establishing that machine learning-based models can efficiently outperform currently accepted criteria for PICC-induced thrombosis assessment [18].

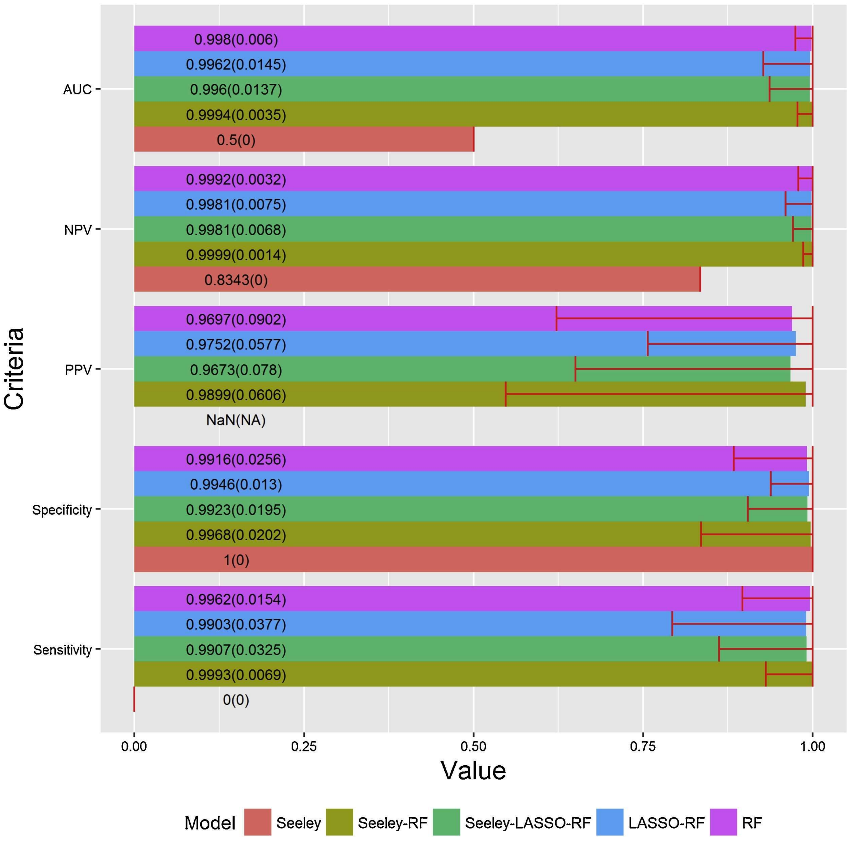
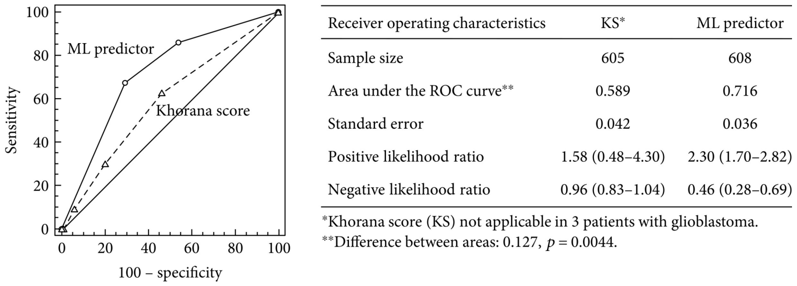


Figure 3.2: The performance of different machine learning models in predicting PICC thrombosis by Liu et al. [18].

Validation of a Machine Learning Approach for Venous Thromboembolism Risk Prediction in Oncology

Ferroni et al. [19] has conducted a similar research. The study intended to perform the risk assessment for thrombosis in cancer patients treated with chemotherapy, with the help of machine learning. The original dataset consisted of 1433 patients. The patients were monitored for approximately ten months. During this period, medical, demographic and biochemical data of patients were collected to utilise in the training process. The machine learning model was a combination of kernel machine learning techniques and random optimisation. The kernel machine learning model combined multiple Support Vector Machines. The training set consisted of 70% of the original dataset, similarly, the testing dataset constituted of the remaining 30%. The model was trained in five different learning sessions that included five distinct random optimisations. The testing dataset was employed to calculate the accuracy of the model and later compared it to the Khorana score. Khorana score is a risk assessment model utilised in similar circumstances. The final comparison confirmed that the machine learning model exceeded Khorana score in performance. The machine learning model scored 0.716 on the AUC scale as compared to the Khorana score model that scored 0.589. The standard error for the machine learning model and the Khorana score model stood at 0.036 and 0.042, respectively. The comparison has been presented in Figure 3.3. The study established that a machine learning-based risk assessment model can help identify potential thrombosis in cancer patients [19].



Receiver operating characteristics	KS*	ML predictor
Sample size	605	608
Area under the ROC curve**	0.589	0.716
Standard error	0.042	0.036
Positive likelihood ratio	1.58 (0.48–4.30)	2.30 (1.70–2.82)
Negative likelihood ratio	0.96 (0.83–1.04)	0.46 (0.28–0.69)

*Khorana score (KS) not applicable in 3 patients with glioblastoma.
**Difference between areas: 0.127, $p = 0.0044$.

Figure 3.3: The comparison of the performance of the machine learning model and Khorona score by Ferroni et al. [19].

Random Forest Active Learning for AAA Thrombus Segmentation in Computed Tomography Angiography Images

Maiora et al. [20] also attempted to demonstrate the utilisation of machine learning tools in thrombosis related complications. Abdominal Aortic Aneurysm (AAA) is the enlargement of the aorta blood vessel at a specific point. Aorta supplies blood to the body and, its deformation may cause the formation of thrombus. The analysis of this complication is generally performed with the help of 3D Computerised Tomography Angiography (CTA) which is a visualisation technology. The identification of this specific type of thrombus is particularly challenging because it resembles the surrounding aortic tissues. This study endeavours to develop an active learning system to separate the CTA images. The segmentation of such images is usually influenced by the noise present in the data, however, the proposed system would require minimal human intervention to achieve the outcome. The study has utilised a hybrid approach to attain the segmentation task. It consists of active learning and Random Forests. According to the authors, active learning minimises the intervention of human inputs, data anomalies and helps kick start learning with a small dataset. Random Forests offers better accuracy, swift learning and adaptive nature to the incremental datasets. The goal of the classifier is to segment the images into the target region and the surrounding background. The system starts training with the initial dataset and, it returns to the user for manual labelling for uncertain images. Once manually labelled, these samples are also included in the training dataset to improve accuracy. The training is accomplished in iterations while each iteration adds five samples to the training dataset. The study illustrates that the performance of the model stabilises when it tuned to implement 80 trees with a depth parameter of 20, as displayed in Figure 3.4. The study demonstrates that accuracy up to 0.99 is achieved after at most 4 iterations. The accuracy attained on the test images is also above 0.98 in all cases. The research concluded that the proposed solution can be employed to effectively differentiate the thrombus in the aorta from its surrounding tissues [20].

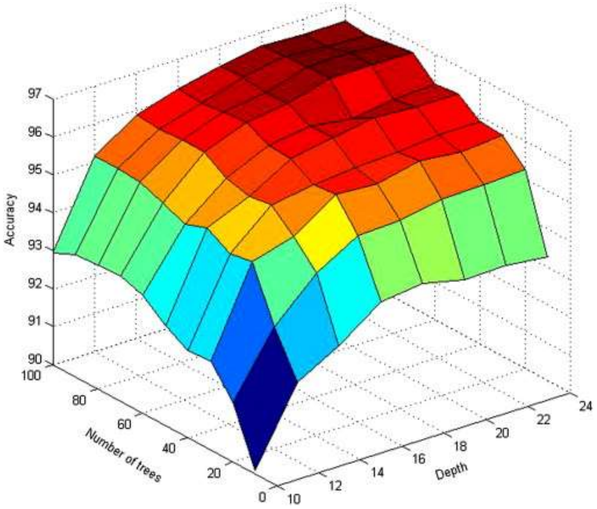


Figure 3.4: The segmentation accuracy of Random Forests over a range of depth and number of trees [20].

Sequential vs. Batch Machine-Learning with Evolutionary Hyperparameter Optimization for Segmenting Aortic Dissection Thrombus

Morariu et al. [21] have conducted similar research on thrombus formation in the aorta that is comparable to the study conducted by Maiora et al. [20]. Thrombus formation in the aorta can be lethal, therefore, expeditious diagnosis is crucial. The lumen of the surrounding tissues is very similar to the thrombus and an abundance of similar structures in the abdominal area complicates the diagnosis further. The diagnosis of such complication is generally performed with the help of 3D Computerised Tomography Angiography that produces blurred dissection images. The study has provided some mathematical background to highlight the blurry boundaries, however, it is not directly related to our work. This study trains the classifier in three different approaches. Firstly, to train the model on all of the datasets. Secondly, to train the model using the result from the previous image (slice by slice). Lastly, to train multiple classifiers that are responsible for a specific area in the image. There are two machine learning algorithms employed in the study, kNN and SVM. Hyperparameter tuning for kNN includes the determination of the optimal number of nearest neighbours. It was determined by decreasing the error with ten-fold cross-validation and Euclidean distance was applied as the distance function. The hyperparameters for the SVM algorithm were tuned with the help of evolutionary algorithms. They include hyperparameters *Sigma* and *C*. Grid search was not applied in this case for its time-consuming nature. The study concludes that SVM outperforms kNN algorithm in terms of classification accuracy and the stability criterion [21]. Figure 3.5 demonstrates a comparison between kNN and SVM algorithm in terms of Recall (RC), Precision (PR) and Dice similarity coefficient (DSC).

	kNN	SVM
DSC	74.15 ± 11.74	81.93 ± 05.65
PR	65.32 ± 15.41	76.60 ± 07.35
RC	89.24 ± 06.26	89.30 ± 06.82

Figure 3.5: The comparison of the performance of kNN and SVM algorithms over Dice Similarity Coefficient, Precision and Recall performed by Morariu et al. [21].

The use of Artificial Neural Network Analysis can Improve the Risk-Stratification of Patients Presenting with Suspected Deep Vein Thrombosis

Willan et al. [22] utilised Artificial Neural Networks to stratify patients at the risk of deep vein thrombosis. The patient cohort consisted of 11490 cases over a period of 7 years. The study illustrates that the D-Dimer test and Well's score are generally employed to exclude patients with deep vein thrombosis. The effectiveness of these techniques is limited and, machine learning tools can be used in addition to these tools to enhance the outcome. The study included only those patients in the Artificial Neural Networks analysis who possessed a comprehensive Well's score, D-Dimer test and an ultrasound scan. A total of 7080 patients were eligible for the final analysis. This dataset was divided into subsets of 5270 & 1810 for training and testing datasets, respectively. The dataset mainly consisted of age, sex, D-Dimer result and the components of Well's score. The study attempted to resolve the binary classification problem that was to conclude if a patient has thrombosis. An artificial Neural Network was constructed to accomplish this goal. It consisted of an input layer that was constituted by 13 nodes, a hidden layer that contained 8 nodes, and the output layer with a single node. A Support Vector Machines model and a Random Forests model were also trained to compare the performance with Artificial Neural Networks. The study concluded that Artificial Neural Networks outperformed the other two algorithms. The model was able to eliminate patients with deep vein thrombosis without the aid of an ultrasound scan, therefore, it proved to be superior to the D-Dimer test and Well's score. The Artificial Neural Networks model proposed only 62% of patients for an ultrasound scan as compared to the D-Dimer test whose peak performance was 87%. The performance of the model has been illustrated in Figure 3.6.

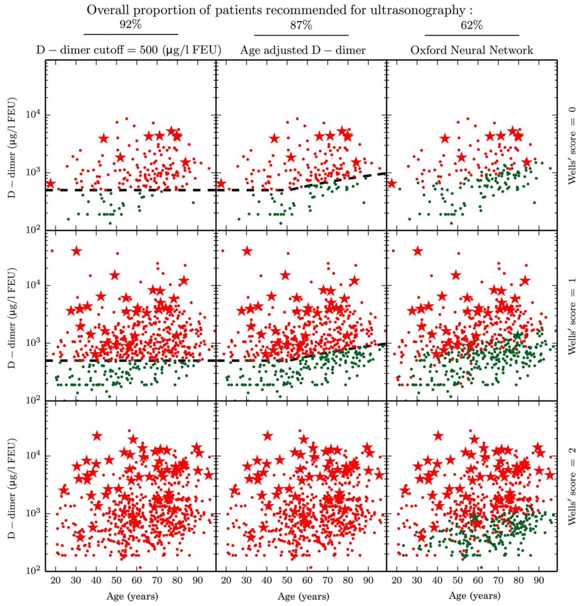


Figure 3.6: The performance of the ANN model against the D-Dimer test in classifying thrombotic patients [22].

Vilhena et al. [23] have also employed Artificial Neural Networks to identify patients who are susceptible to thrombosis. The final machine learning model was able to classify patients with an accuracy, sensitivity and specificity higher than 95%.
Yang et al. [24] have built a machine learning model based on Random Forests to predict venous thromboembolism. The results were compared against the Padua linear model that is utilised for risk assessment of thrombosis. The machine learning model scored 0.815 on the AUC scale and outperformed the Padua model that scored 0.789.
Semiz et al. [25] have trained another machine learning model based on Logistic Regression to detect pump thrombosis in Left Ventricular Assist Devices (LVADs). LVADs generally fail due to the development of pump thrombosis in the device. The sensitivity, accuracy and precision of the model stood at 90.9%, 88.9% and 83.3%, respectively.

3.3 Comparing Machine Learning Techniques

Machine learning is a vast domain of computer science. It consists of several algorithms that can be employed to train predictive models. This section presents a literature review primarily focused on the selection of machine learning techniques. It examines the utilisation of machine learning in the directly associated studies to thrombosis. It additionally presents several studies that are entirely concentrated on comparing a range of machine learning techniques.

3.3.1 Related Studies

Nafee et al. [17] developed a number of machine learning models that mainly consisted of Random Forests, Extreme Gradient Boosting and Classification Trees. The predictions of different models were combined with the help of the Ensemble Learning method to generate a unified prediction. The study conducted by Liu et al. [18] has also utilised the Random Forests algorithm to predict the future incidence of Peripherally Inserted Central Venous Catheter (PICC) thrombosis in cancer patients. The study employed Least Absolute Shrinkage and Selection Operator (LASSO) to perform the predictor selection. Maiora et al. [20] has also relied on Random Forests to perform machine learning analysis of complications associated with Abdominal Aortic Aneurysm (AAA). The Random Forests models are trained in close collaboration with active learning. Yang et al. [24] also utilised the Random Forests algorithm to predict venous thromboembolism.
A study conducted by Ferroni et al. [19] has employed kernel machine learning techniques to perform the risk assessment for thrombosis in cancer patients treated with chemotherapy. The kernel machine learning model combined multiple Support Vector Machines. The study has additionally considered Random Optimisation. Morariu et al. [21] have conducted a research to predict aortic dissection thrombus. There are two machine learning algorithms employed in

the research, kNN and SVMs. The hyperparameters for the SVMs algorithm were tuned with the help of evolutionary algorithms. Hyperparameter tuning for kNN was performed by decreasing the error with ten-fold cross-validation.
An investigation performed by Willan et al. [22] has utilised Artificial Neural Networks to stratify patients at the risk of deep vein thrombosis. The neural network implemented in the study consisted of an input layer that was constituted by 13 nodes, a hidden layer that contained 8 nodes, and the output layer with a single node. Vilhena et al. [23] have also employed Artificial Neural Networks to identify patients who are susceptible to thrombosis.
Semiz et al. [25] have implemented machine learning models with help of Logistic Regression to detect pump thrombosis in Left Ventricular Assist Devices (LVADs).

3.3.2 Comparative Studies

An Empirical Comparison of Supervised Learning Algorithms

Caruana et al. [26] have conducted an empirical comparison between different machine learning algorithms. The algorithms incorporated in the study consisted of Artificial Neural Networks, Support Vector Machines, Logistic Regression, Random Forests, Decision Trees, boosted and bagged trees. The comparative study calibrated the hyperparameters for the involved algorithms to an optimal level and, it comprised 11 binary classification problems. The final comparison was conducted with the help of numerous performance metrics. It includes accuracy, lift, area under the curve, average precision, cross-entropy and root-mean-square error. Boosted trees exhibited the best performance, followed by Random Forests, Support Vector Machines and Artificial Neural Networks.

Performance Comparison of Feed-Forward Neural Networks Trained with Different Learning Algorithms for Recommender Systems

Hassan et al. [27] have explored Artificial Neural Networks from the perspective of the optimisation algorithms. The study concentrated on training neural networks with the aid of different learning algorithms to build recommender systems. These learning algorithms consisted of the Delta Rule algorithm (Adaline), the Backpropagation algorithm, Levenberg-Marquardt algorithm, Genetic Algorithm and Simulated Annealing algorithm. The study employed mean square error as the evaluation criteria. It concluded that the Adaline algorithm outperformed the other algorithms involved in the study and, it required a significantly lower number of iterations during the training process.

Performance Comparison of Machine Learning Algorithms and Number of Independent Components Used in fMRI Decoding of Belief vs. Disbelief

A study performed by Douglas et al. [28] has compared the performance of six machine learning algorithms. These algorithms included K-Star, Naïve Bayes, Support Vector Machines, Decision Trees, AdaBoost and Random Forests. A neuroimaging dataset of informed participants was employed for training the classifiers. The hyperparameters for each algorithm were tuned individually to produce the optimal results. Random Forests outperformed the remainder of the algorithms by producing an accuracy of 92%. AdaBoost followed closely by scoring 91%.

An Empirical Performance Comparison of Machine Learning Methods for Spam E-mail Categorization

Chih-Chin et al. [29] has also pursued a comparative study that ranks the performance of different machine learning algorithms. The aged study comprised of Naïve Bayes, Support Vector Machines, K-Nearest Neighbour and Term Frequency-Inverse Document Frequency. The machine learning algorithms were evaluated individually over a spam email segregation task. Naïve Bayes and Support Vector Machines performed reasonably well. They scored an accuracy percentage of up to 93% and 92% while segregating different sections of an email, respectively.

Performance Comparison of Intrusion Detection Systems and Application of Machine Learning to Snort System

Shah et al. [30] have investigated the application of machine learning algorithms in intrusion detection systems. This study examined the performance of Suricata and Snort intrusion detection systems to identify suspicious traffic on computer networks. A combination of Suricata and Support Vector Machines were investigated to produce the most reliable outcomes. The study concluded that an optimised version of Support Vector Machines utilising the Firefly algorithm yielded the most promising results. The false-negative rate stood at 2.2%, and a false positive rate of 8.6% was recorded.

A Performance Comparison of Machine Learning Algorithms for Arced Labyrinth Spillways

A study conducted by Salazar et al. [31] has examined the application of machine learning algorithms to optimise the geometric variables while designing arced labyrinth spillways. Labyrinth weirs are generally employed to optimise the construction of dams and reduce construction costs. The study compared the performance of Random Forests and Artificial Neural Networks to determine the discharge coefficient for the weir. A weir maintains the flow of water in dam spillways. The performance of both algorithms was compared by employing metrics such as mean error, root mean squared error, mean absolute error and mean absolute percentage error. The study concluded that Artificial Neural Networks outperformed Random Forests over the standard comparison criteria, however, the performance of Artificial Neural Networks is profoundly dependent on the initialisation of weights during training.

3.4 Summary

After thoroughly examining the utilisation of machine learning algorithms in Related Studies and Comparative Studies presented in the previous sections, it is concluded that this study will utilise Random Forests, XGBoost, Artificial Neural Networks and Support Vector Machines for the upcoming experimentation.

Chapter 4

Experimental Setup

This chapter presents a thorough description of the experimental setup for the investigation.

4.1 Dataset Description

The data has been collected under *RI Schedule* research initiative by Østfold Hospital Kalnes, Norway. There are roughly 1600 (after removing duplicates) thrombosis patients admitted to the study. The dataset consists of approximately 195 attributes that describe a patient's circumstances. The dataset demonstrates the personal characteristics, clinical details, prohibited medications, prescribed therapies (medications), medical history, respective thrombosis risk factors, Well's score, physical health indicators, D-Dimer and personal ultrasound tests of the patients. The primary target attribute in this research is ultrasound results which conclude the thrombosis diagnosis. Once additional data is available, the specific location of the relapse can be considered as well.

RI Schedule Dataset					
Samples			Attributes		
Total	Positive	Negative	Total	Numerical	Categorical
1800	364	1436	195	166	29

Table 4.1: A description of the original *RI Schedule* dataset.

The original dataset from Østfold Hospital Kalnes consisted of 195 attributes and 1800 patients, including some duplicates. These attributes can be divided into several broad categories as follows. The features describing the fundamental characteristics of the patients are included in the *baseline* category. They consist of a patient's age, sex, medical history, medication and Well's score that is calculated by a thorough physical examination. It is focused on the thrombotic symptoms. The *risk factors* category consists of the thrombotic history of a patient, pregnancy status and cancer. It involves genetic risk factors as well.

The *immobilisation* category generally comprises the features related to the physical movement of the patient. It includes features regarding neurological diseases, smoking and travel history as well.
Clinical symptoms is a broad category that encompasses clinical observations recorded in a clinic. These observations include pain and swelling in different parts of the body, blood pressure, bleeding, fever, excessive sweating, frostbite, various blood tests and D-Dimer test. Some of these examinations are repeated for patients with multiple visits. This category might include autopsy as well if a patient included in the study passes away. The patients are eventually diagnosed with the aid of ultrasonography. It provides a concluding diagnosis and, this attribute has been chosen to perform the predictive analysis. The resulting study is, therefore, a binary classification problem.

4.2 Dataset Preprocessing

Data preprocessing is a combination of techniques that are primarily related to the manipulation of the training data. The methods generally consist of conversion, addition, and truncating procedures [16]. Data preprocessing is performed before the modelling process and, it can influence the results significantly. It additionally provides a further extensive understanding of the dataset.

4.2.1 Preprocessing Tools and Initial Manipulation

A number of tools and software have been utilised for data preprocessing and for the study in general. Pandas is an open-source data analysis and manipulation tool in the Python programming language [32]. It has been thoroughly utilised throughout the study to perform essential manipulation of the data prior to the training process. Pandas is developed with the help of NumPy, NumPy is a seasoned Python framework that is employed extensively by the data science community to perform mathematical and statistical operations [33]. It is flexible and productive. Matplotlib is a Python library that can be utilised to draw knowledgeable visualisations [34]. Matplotlib can be implemented in combination with Pandas to produce swift yet remarkably helpful plots. This study has utilised Seaborn for some visualisations. Seaborn is a Python library that can produce powerful plots with the help of Matplotlib. It has been developed by *Michael Waskom* and the Seaborn development team.
The initial dataset was provided in two different files. One of the files contained the data points, but the data was unlabeled, i.e., without the titles of the attributes. The second file incorporated the labels (names) of the attributes. The titles (names) were assigned to the corresponding columns in the earlier file with the help of Pandas. The original labels were supplied in the Norwegian language, however, they were translated to the English language to yield an agile and easy-to-understand outcome.

4.2.2 Expunged Attributes and Data points

This study was performed in collaboration with the medical experts from Østfold Hospital Kalnes. They contributed essential information regarding the medical aspects of the study. It was subsequently suggested to extract two subsets from the principal dataset. The first subset comprised all feasible features from the original dataset, however, several dysfunctional and internal administrative features were removed. It was because they did not contribute to the outcome. This subset has been referred to as the *Full Dataset* in the upcoming literature. The features dropped from the Full dataset can be found in Appendix A.2.

The second subset essentially consisted of primary patient information, Well's score, basic information from the risk factors, immobilisation and clinical symptoms categories. It has been referred to as the *Minified Dataset* in the upcoming literature. The features truncated from the Minified dataset are presented in Appendix A.3.

The dataset comprised several duplicate patients as well. This problem was consulted with the medical experts, who supervised the process of data collection. It was concluded that the duplicates should be truncated except for the latest record. The dataset was therefore reduced to *1653* data points from *1800* instances originally.

4.2.3 Missing Values Management

The dataset comprised several features with missing values. These missing values can be substituted with appropriate alternatives. The alternatives generally include *mean*, *median* and *mode*. This study relied solely on *mode* given the nature of the missing values. Table 4.2 presents a summary of the management of the missing values and Figure 4.1 presents an overall appearance of the missing values in the dataset prior to the substitution.

Feature Name	No. of Missing Values	Substituting Method
Clinic Vital Blood Pressure Diastolic	05	Mode
Risk Factors P Pills Type	03	Mode
Clinic Blood Test Results GFR	03	Mode
Risk Factors HRT Type	02	Mode
Clinic Blood Test Results CRP	01	Mode

Table 4.2: The substitution procedures for missing values in different features.

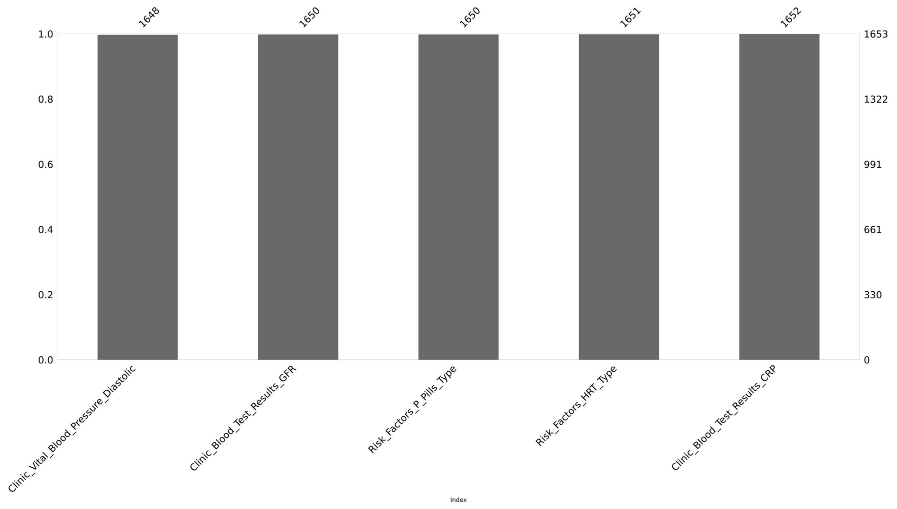


Figure 4.1: A graphical illustration of the missing values in the dataset.

4.2.4 Feature Transformations

The dataset incorporated a range of features. These features were related to various data types. The majority of the features were based on continuous numeric values, however, the dataset contained several categorical features as well. Most of the categorical features were truly categorical, i.e., based on discrete classes and some numeric features were disguised as categorical features due to rare occurrences of *strings* in the corresponding columns. These particular instances of unwanted *strings* were substituted with appropriate numbers and, the datatypes of these specific columns were altered accordingly. Table 4.3 demonstrates a summary of the operations conducted.

The final diagnosis is based on ultrasonography. This feature is represented with "UL_Proven_VTE". The patients are diagnosed with aid of the D-Dimer test as well prior to ultrasonography. Some Patients can therefore be declared non-thrombotic after performing the D-Dimer test exclusively. These patients are represented with *-1* in the ultrasonography feature. It has been illustrated in Figure 4.2. The patients with positive ultrasonography are represented with *1* and, those with negative results are assigned *0* label. The *-1* label has been substituted by *0* since both refer to non-thrombotic patients. The final feature has been described in Figure 4.3.

4.2.5 Final Operations

The final operations performed on the dataset during the preprocessing phase involved One-Hot Encoding, data splitting and Data Standardisation. One-Hot Encoding is a technique that is utilised to convert categorical data into numeric

Feature Name	Unusual Term	Substitute	Updated Datatype
Clinic Blood Test Results GFR	>60	60	Float
Clinic Blood Test Results HB	11,,7	11	Float
Fragmin Number Of Times	Ingen	-1	Integer
Fragmin Dose Field	Ukjent	-1	Integer
Klexane Number Of Times	Ingen	-1	Integer
Klexane Dose Field	Ukjent	-1	Integer

Table 4.3: The conversion of unusual terms in different features to numerical terms.

data. It accomplishes the goal by encoding a categorical feature into multiple binary features [35]. The number of binary features is associated with the number of classes in the original categorical feature. This technique was applied to the dataset to convert the outstanding categorical data into numeric data. Pandas library has been utilised to implement the procedure [32]. The dataset was subsequently split into two subsets, the training set and the testing set. The testing and training datasets comprised 20% and 80% of the original dataset, respectively. In order to ensure reproducibility, the splitting was stratified based on "UL_Proven_VTE" feature and employed a constant random state throughout the study. Overfitting is a common obstacle in machine learning where the model excessively adapts to the training set and the generalisation slides. k-Fold Cross-Validation was employed during the training process to eliminate potential overfitting. It partitions the training dataset into multiple subsets. The model is trained with all subsets except the first subset (first fold). It is used for evaluation. This process is repeated for the second subset, and so on [16]. It has been demonstrated in Figure 4.4. Data standardisation was performed with the help of Scikit-learn's *StandardScaler* [36]. It rescales the data to have a standard deviation of *1* [16]. Data standardisation can be significantly helpful by aiding some algorithms such as Artificial Neural Networks to coverage much faster.

4.3 Machine Learning Setup

The purpose of this investigation is to compare the performance of machine learning algorithms in accurately predicting thrombosis. As the target attribute suggests, it is a classification problem where the patients need to be segregated

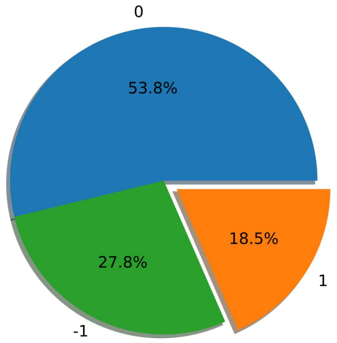


Figure 4.2: The distribution of classes in the original ultrasonography feature (target attribute).

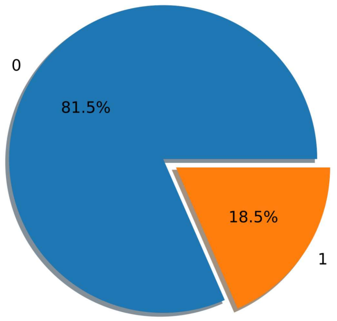
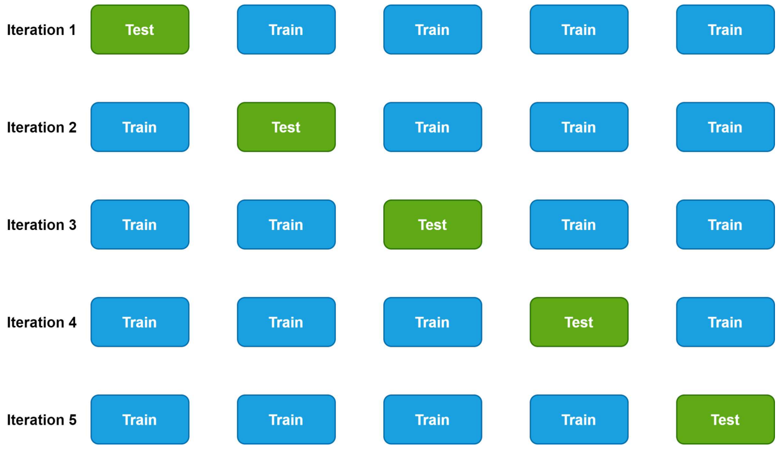


Figure 4.3: The distribution of classes in the manipulated ultrasonography feature (target attribute).

into different classes depending on the chance of occurrence of thrombosis. This section presents comprehensive information regarding the training and evaluation of machine learning models on the datasets described earlier. The machine learning algorithms have been selected with the help of a systematic selection procedure. It has been thoroughly described in the *Related Work* chapter.

4.3.1 Hyperparameter Tuning

Hyperparameter tuning shall be a crucial part of the upcoming training process. Hyperparameters are predefined numbers or beliefs that need to be provided to machine learning algorithms before the training process. Hyperparameters control the overall learning process during the training of a machine learning model [37]. For instance, "n_estimators" is a crucial hyperparameter for random forests algorithm. It controls the number of trees involved in the tree building and decision-making process. The learning rate, number of epochs or iterations, activation functions, hidden layers and several more are important



Iteration 1	Test	Train	Train	Train	Train
Iteration 2	Train	Test	Train	Train	Train
Iteration 3	Train	Train	Test	Train	Train
Iteration 4	Train	Train	Train	Test	Train
Iteration 5	Train	Train	Train	Train	Test

Figure 4.4: k-Fold Cross-Validation (5 Folds)

hyperparameters that need to be tuned prior to the learning process. A perfect combination of hyperparameter tuning can produce exceptional outcomes; therefore, Hyperparameter tuning has been granted a particular focus during this study to maximise the performance of the models.
There are numerous techniques to perform hyperparameter optimisation. This study has utilised two principal techniques to perform hyperparameter tuning. Grid search (Full factorial design) is an exhaustive hyperparameter searching method. It examines all possible combinations on the provided ranges of the corresponding hyperparameters [37]. Grid search can be notably computation-intensive on parameters with extensive searching spaces. In order to address this concern Bayesian optimisation has also been utilised to perform the tuning process. Bayesian optimisation is an excellent method to optimise hyperparameters. It employs the Bayes theorem to find the maxima or minima of a given function. Bayesian optimisation generates a surrogate probability model of the objective function and updates it regularly depending on the performance of hyperparameters [37]. It utilises the performance history of the previously evaluated hyperparameters to direct future exploration.

4.3.2 Random Forests

Random Forests is an excellent ensemble learning algorithm. It generates a number of decision trees to perform the predictive analysis. It has been discussed thoroughly in the *Background* chapter. The algorithm has been implemented with the help of Scikit-learn. Scikit-learn is a Python library that comprises several tools for machine learning and statistical modelling [36].

Model Structure and Hyperparameter Tuning

The structure of the model was established with the help of Grid Search [37]. The entire tuning process was performed while utilising k-Fold Cross-Validation (5 folds) in order to ensure a reliable outcome [16]. The tuning model utilised *accuracy* as the evaluation metric. The hyperparameter tuning included the following hyperparameters.

- **"n_estimators":** It represents the number of trees utilised for formulating a prediction. The searching space for this hyperparameter was between 1 and 500. The tuning produced 112 as the optimal value for this hyperparameter on the Full dataset. *(Full Dataset: 112, Minified Dataset: 74)*

- **"max_features":** This hyperparameter defines the number of features to consider while seeking the best split. The searching concluded with 164, between the range of 1 and the maximum number of features, for the Full dataset. *(Full Dataset: 164, Minified Dataset: 71)*

- **"bootstrap":** This binary hyperparameter can be utilised to enable or disable bootstrapping (a randomisation technique). It was concluded that enabling bootstrap sampling produces better results. *(Full Dataset: True, Minified Dataset: True)*

- **"max_depth":** As the name suggests, it controls the maximum depth of a tree. The searching space stretched between 1 and 30. It also included "no limit over the depth of the tree" in the searching space. It concluded with 17 as the outcome for both datasets. *(Full Dataset: 17, Minified Dataset: 17)*

- **"min_samples_leaf":** It ensures the minimum number of samples at a leaf node. A split shall not be considered if it leaves fewer than the defined number of samples in the subsequent branches. It was searched between 0 & 10 and, 1 was returned as the optimal result. *(Full Dataset: 1, Minified Dataset: 1)*

- **"min_samples_split":** This hyperparameter defines the minimum number of samples required to split a node in the tree. The tuning process considered the range between 1 & 10, it was concluded with 2. *(Full Dataset: 2, Minified Dataset: 2)*

- **"class_weight":** It is often challenging to acquire desirable results with imbalanced datasets. This hyperparameter can be utilised to increment the importance of minority classes. This hyperparameter has been utilised to increase the weight of thrombotic patients. Scikit-learn provides a function to calculate the respective weights of the classes depending on the frequency of their occurrence. This function is termed *balanced* which utilises this formula, *n_samples / (n_classes * np.bincount(y))* [36]. Keras documentation presents a similar approach that has been described in the Artificial Neural Networks section. *(Full Dataset: balanced, Minified Dataset: balanced)*

- **"criterion":** It is employed to select the function to measure the quality of a split. It offers two alternatives, Gini impurity and Entropy information gain. The tuning concluded with Gini impurity as the preferred alternative. *(Full Dataset: gini, Minified Dataset: gini)*
- **"random_state":** Random Forests utilise randomisation techniques frequently, for instance, bootstrapping and sampling of features. It may lead to slightly different results for each training session. This difficulty can be resolved by providing a randomisation seed to the model. "random_state" can be utilised to establish the initial seed. This study generally employs *333* for random states. *(Full Dataset: 333, Minified Dataset: 333)*

The rest of the hyperparameters utilised the default values implemented by Scikit-learn library [36].

4.3.3 XGBoost

XGBoost is a machine learning library that is based on the gradient boosting framework. It offers exceptional efficiency and performance on structured datasets [38]. Gradient Boosting has been presented in detail in the *Background* chapter.

Model Structure and Hyperparameter Tuning

XGBoost library comprises a Scikit-learn wrapper that renders it highly compatible with the Scikit-learn optimisation framework [38]. Grid Search has been employed for the hyperparameter optimisation of XGBoost models [37]. It utilises k-Fold Cross-Validation (5 folds) to maximize generalisation. The hyperparameter tuning model employed *accuracy* as the evaluation metric and it included the following hyperparameters.

- **"learning_rate":** Gradient boosted trees can be expeditious in learning that might cause overfitting. The learning process can be slowed down with the help of this hyperparameter. It was tuned between 0.01 and 0.50 with a step size of 0.01. It settled to 0.10 for the Full dataset. *(Full Dataset: 0.10, Minified Dataset: 0.19)*
- **"n_estimators":** This hyperparameter is equivalent to the number of boosting rounds since each boosting round increments the number of trees by one [38]. The searching space spanned between 1 and 250 that concluded with 31 for the Full dataset. *(Full Dataset: 31, Minified Dataset: 123)*
- **"max_depth":** The depth of a tree is controlled by this hyperparameter. Deeper trees might learn quicker and capture more details but, they are

susceptible to overfitting. This difficulty can be overcome with the aid of this hyperparameter. The searching concluded with 7, between the range of 1 and 10, for the Full dataset. *(Full Dataset: 7, Minified Dataset: 4)*

- **"min_child_weight":** This hyperparameter controls the sum of instance weight in leaf nodes. The tree-building process will terminate if the partition of a node results in a leaf node with the sum of instance weight smaller than what this hyperparameter specifies. The tuning process considered the range between 0 and 10. It concluded with 2 for the Full dataset. *(Full Dataset: 2, Minified Dataset: 3)*

- **"subsample":** Subsampling can be helpful to avoid overfitting. Depending on the value of this hyperparameter, XGBoost randomly subsamples training instances prior to the tree building process. The process is repeated during each boosting iteration. The searching space stretched between 0 and 1 with a step size of 0.10. The concluding outcome was 0.9 for both datasets. *(Full Dataset: 0.9, Minified Dataset: 0.9)*

- **"colsample_bytree":** It is concerned with the subsampling of the data features during the construction of a decision tree. This process is iterative and takes place during every boosting iteration. The hyperparameter was searched between 0 and 1 with a step size of 0.10. The tuning process concluded with 0.7 as the outcome, for the Full dataset. *(Full Dataset: 0.7, Minified Dataset: 1.0)*

- **"scale_pos_weight":** This hyperparameter is utilised to control the weights of minority and majority classes in an imbalanced dataset. It has been employed to increase the weight of thrombotic patients. XGBoost documentation recommends a function to calculate the respective weights of the classes depending on the frequency of their occurrence. The weights can be calculated by employing the following formula, *sum(negative instances) / sum(positive instances)* [38]. *(Full Dataset: 4.418032786885246, Minified Dataset: 4.418032786885246)*

- **"objective":** This parameter is utilised to establish the problem type and the learning objective, generally known as the objective function. Given the nature of the problem in this study, *binary:logistic* has been provided as the learning objective. *(Full Dataset: binary:logistic, Minified Dataset: binary:logistic)*

- **"seed":** The hyperparameter is employed to provide a launchpad for the forthcoming randomisation during the training process. It ensures that the results are reproducible. This study employs *0* as the seed for experiments related to XGBoost. *(Full Dataset: 0, Minified Dataset: 0)*

- **"tree_method":** XGBoost offers several tree construction algorithms. This study employed *hist* for its faster performance and a slightly better accuracy during the Grid Search experiments [38]. *(Full Dataset: hist, Minified Dataset: hist)*

The rest of the hyperparameters utilised the default values implemented by XGBoost library [38].

4.3.4 Support Vector Machines

Support Vector Machines (SVMs) are a highly capable set of supervised learning methods. SVMs are robust and offer plenty of flexibility. SVMs can perform both classification and regression. They are generally utilised for classification problems. SVMs are particularly effective in high dimensional spaces. SVMs can be altered with the help of kernel functions that enables them to process a wide array of datasets [16].

Model Structure and Hyperparameter Tuning

Scikit-learn presents an outstanding implementation of Support Vector Machines. It allows a pliable hyperparameter optimisation with the help of features such as multiple kernel functions and regularisation tuning [36]. As discussed in the previous sections, Grid Search has been employed for the hyperparameter optimisation [37]. The tuning procedure was performed while utilising k-Fold Cross-Validation (5 folds) [16]. The tuning model utilised *accuracy* as the evaluation metric. The hyperparameter tuning included the following hyperparameters.

- **"kernel":** This hyperparameter is employed to choose the kernel type in a model. Scikit-learn has built-in implementations of *linear, poly, rbf, sigmoid* and precomputed kernel functions. The tuning procedure included all of the alternatives available. It concluded with *sigmoid* for the Full dataset. *(Full Dataset: sigmoid, Minified Dataset: linear)*

- **"C":** It controls the regularisation during the training process. The value of this hyperparameter is inversely proportional to the strength of regularisation and directly proportional to the cost of misclassification. The searching space extended between 0.1 and 50 that concluded with 1 as the outcome for the Full dataset. For the Minified dataset, the final searching space ranged between 0.01 and 0.2 with a step size of 0.001. *(Full Dataset: 1, Minified Dataset: 0.050999999999999997)*

- **"gamma":** This hyperparameter is utilised to adjust the kernel coefficient that controls the sensitivity between the feature vectors. It is effective in *rbf*, *poly* and *sigmoid* kernel functions. Scikit-learn has listed two rules of thumb to choose the value of this hyperparameter, "scale" *(1 / (n_features * X.var())* and "auto" *(1 / n_features)* [36]. It was tuned between 0.002 and 0.004 with a step size of 0.000001. The optimal outcome was 0.002745000000000099 for the Full dataset. *(Full Dataset: 0.002745000000000099, Minified Dataset: Not applicable)*

- **"coef0":** This hyperparameter represents the independent term in a kernel function. It influences the performance of a model that employs *poly* and *sigmoid* kernels. The tuning process encompassed a range from -0.1 to 0.1 with a step size of 0.001 and, it concluded the search with 0.08400000000000016 as the outcome. *(Full Dataset: 0.08400000000000016, Minified Dataset: Not applicable)*

- **"tol":** It represents the tolerance for the stopping criterion. The processing time can be decreased if the value of this hyperparameter is increased, however, it affects the performance negatively. The tuning extended between 0.00001 and 0.001, the step size was 0.00001. The final outcome was 0.00001. *(Full Dataset: 0.00001, Minified Dataset: 0.00001)*
- **"class_weight":** As explained earlier, this hyperparameter is beneficial in imbalanced datasets to adjust the weight of different classes. It has been employed to increase the weight of the minority class that is thrombotic patients. Scikit-learn provides a function to calculate the respective weights of the classes depending on the frequency of their occurrence. This function is termed *balanced* which utilises this formula, *n_samples / (n_classes * np.bincount(y))* [36]. *(Full Dataset: balanced, Minified Dataset: balanced)*
- **"random_state":** The hyperparameter is utilised to give a launchpad to the upcoming randomisation during the training process. It helps in reproducing the outcomes. This study generally employs *333* for random states in experiments related to Scikit-learn. *(Full Dataset: 333, Minified Dataset: 333)*

The rest of the hyperparameters utilised the default values implemented by Scikit-learn library [36].

4.3.5 Artificial Neural Networks

Artificial Neural Networks (ANNs) are an extraordinary family of machine learning algorithms. ANNs attempt to replicate the human brain, which has an enormous amount of interconnected neurons. ANNs are exceptionally reliable to process unstructured data and possess a high degree of flexibility [16]. They have been reviewed in the *Background* chapter.

Model Structure and Hyperparameter Tuning

This study utilises Keras to implement the experiments related to Artificial Neural Networks. Keras is a high-level machine learning library that provides a wide variety of Artificial Neural Networks implementations. It is uncomplicated, flexible and robust [39]. The study implements several feedforward neural network models with multiple hidden layers. These models have been optimised with the help of Bayesian optimisation [37]. The Bayesian optimisation model employed *accuracy* as the evaluation metric. The maximum number of trials for the tuning models were limited to 250, while each trial consisted of 50 iterations. The hyperparameter tuning procedure utilised a validation split of 20% to validate the results during the optimisation process.

- **"Model Type":** This hyperparameter is utilised to determine the fundamental model type. This study has employed the *Sequential* model for the experiments associated with ANNs. The *Sequential* model is implemented in the form of a plain stack of layers and each layer supports a single input and output tensor.

- **"Layers and Neurons":** An Artificial Neural Network consists of a number of layers that can be classified into input, hidden and output layers. These are the building blocks of an artificial neural network. This study has utilised an input layer with suitable shapes depending on the input data. There are several deeply connected dense hidden layers in the models that contain a number of neurons. The number of hidden layers and corresponding neurons has been optimised to yield optimum results. The output layer, which is a dense layer, consists of a single neuron.

- **"Activation Functions":** This hyperparameter is helpful in selecting a suitable activation function for a neuron. In ANNs, activation functions calculate the concluding output of a neuron provided an input. There are numerous activation functions available in Keras library [39]. This study tunes the hidden layers with Rectified Linear Activation (ReLU), Logistic (Sigmoid) and Hyperbolic Tangent (Tanh) activation functions. The output layer comprises Logistic (Sigmoid) activation function.

- **"Optimisers and Learning Rate":** ANNs comprise various optimisers that govern the training process. Optimisers are a particular kind of algorithms that can be employed to adjust the weights inside an artificial neural network to diminish the loss. This study has tuned models by utilising Adam, RMSprop and Adagrad optimisers. The learning rate for the corresponding optimisers has been additionally tuned between 0.0001 and 0.1 with the help of Bayesian optimisation.

- **"Loss":** This hyperparameter is employed to choose the loss function in an Artificial Neural Network. It calculates the severity of prediction error during the training process that is termed *loss*. The model improves iteratively to diminish the loss. Given the nature of the problem, this study practices "binary_crossentropy" as the loss function.

- **"Epochs":** It regulates the passage of the dataset through the Artificial Neural Network. A single epoch is concluded when the entire dataset traverses forward and backwards through the Artificial Neural Network. The Bayesian optimisation model performs 50 epochs for each trial. The final model tunes the number of epochs with the help of early stopping. It is assessed over the validated accuracy score with the highest patience of 10 epochs.

- **"Seeds and Random States":** It is used to provide random seeds to the models. This practice can be incredibly effective in reproducing results and a stable training procedure. This study has seeded the models with *333* as the random state.
- **"Class Weights":** Class weights can help adjust the importance of different classes in imbalanced datasets. This hyperparameter has been utilised to increase the weight of thrombotic patients. The weights regarding various classes have been calculated with the help of Keras documentation, Scikit-learn provides an identical function [39] [36]. *Class Weights (0: 0.6131725417439703, 1: 2.709016393442623)*

Table 4.4 has demonstrated a summary of the hyperparameter optimisation for the Full dataset. It presents information regarding the model type, optimising algorithms and corresponding learning rates (LR). It additionally provides the number of hidden layers, corresponding neurons and activation functions (AF) in each layer. The number of epochs has been provided according to the respective activation function, in identical order.

Sequential					
Optimiser	LR	Layers	Neurons	AF	Epochs
Adagrad	0.0421	5	208, 383, 512, 121, 16	ReLU, Sigmoid, Tanh	09, 01, 06
RMSprop	0.0331	1	326	ReLU, Sigmoid, Tanh	03, 09, 13
Adam	0.0431	1	122	ReLU, Sigmoid, Tanh	02, 07, 18

Table 4.4: The structure of the ANNs models obtained from Bayesian Optimisation for the Full dataset.

Table 4.5 illustrates similar information for the Minified dataset.

Sequential					
Optimiser	LR	Layers	Neurons	AF	Epochs
Adagrad	0.0801	3	71, 373, 241	ReLU, Sigmoid, Tanh	08, 01, 07
RMSprop	0.0351	1	16	ReLU, Sigmoid, Tanh	02, 10, 08
Adam	0.0101	3	446, 250, 234	ReLU, Sigmoid, Tanh	01, 21, 14

Table 4.5: The structure of the ANNs models obtained from Bayesian Optimisation for the Minified dataset.

The rest of the hyperparameters utilised the default values implemented by Keras library [39].

Chapter 5

Results

This chapter presents the results from the experiments conducted during the investigation.

5.1 Random Forests

This section presents the performance of Random Forests algorithm on the Full dataset and the Minified dataset.

5.1.1 Confusion Matrices

The confusion matrices are manifested below to describe the performance of the Random Forests algorithm on the testing data. The algorithm has classified the thrombotic patients in the Full dataset with 15 false negatives and 09 false positives. The number of true negatives and true positives stands at 261 and 46, respectively, for the Full dataset. The performance of the algorithm on the Minified dataset concluded with 36 false negatives, 10 false positives, 260 true negatives and 25 true positives.

The Minified dataset comprises a higher amount of false negatives and false positives and a lower number of true negatives and true positives than the Full dataset. The results have been presented in Figures 5.1 and 5.2.

5.1.2 Precision Recall Curves

The precision-recall curves are given below to describe the performance of the Random Forests algorithm on the testing data. The precision-recall curve for the Full dataset encompasses a comparatively higher area than the Minified dataset's precision-recall curve. The optimal F1 score for the Full dataset is positioned relatively higher and farther right on the graph as compared to the optimal F1 score for the Minified dataset. A gradual increment in the recall

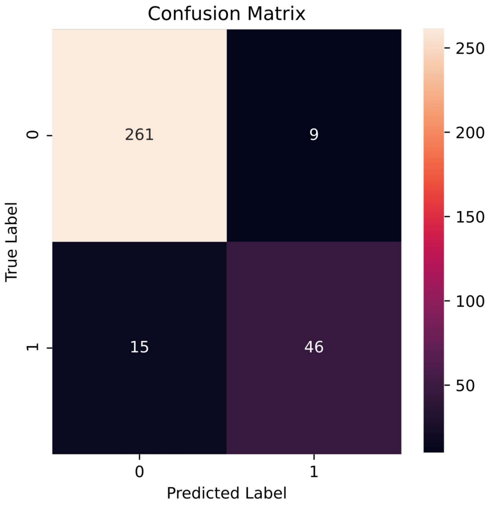


Figure 5.1: The confusion matrix for Random Forests algorithm performance on testing data (Full Dataset).

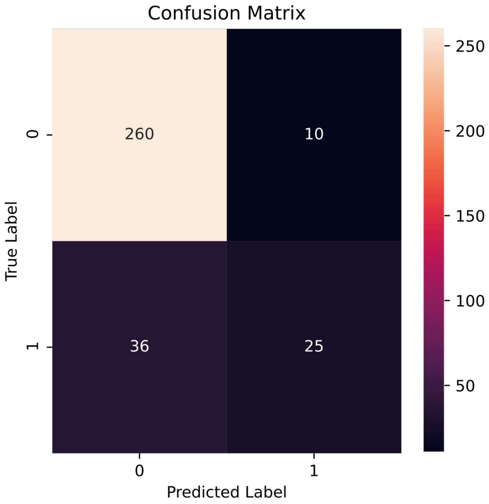


Figure 5.2: The confusion matrix for Random Forests algorithm performance on testing data (Minified Dataset).

results in a continuous decrease in the precision for both datasets. The decrement in the precision is comparatively higher for the Minified dataset. The precision-recall curves have been presented in Figures 5.3 and 5.4.

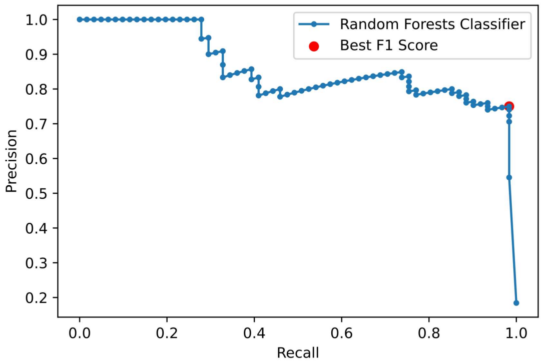


Figure 5.3: The precision-recall curve for Random Forests algorithm performance on testing data (Full Dataset).

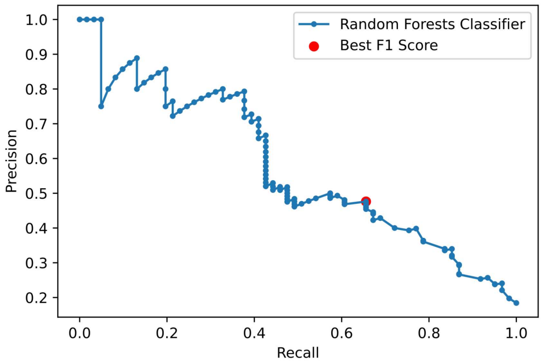


Figure 5.4: The precision-recall curve for Random Forests algorithm performance on testing data (Minified Dataset).

Table 5.1 exhibits a summary of the performance of the Random Forests algorithm on the testing and training data. All results are calculated with respect to the minority class that is thrombotic patients.
The algorithm generally performs better on the training data than the testing data as represented in the table. The performance on the training data is reasonably stable for both the Full dataset and the Minified dataset. The testing data yields alternating performance for the Full dataset and the Minified dataset. The algorithm achieves comparatively better scores in terms of all considered metrics on the Full dataset than the Minified dataset.

	Testing Data				
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
Full Dataset	92.74	83.63	75.40	79.31	85.70
Minified Dataset	86.10	71.42	40.98	52.08	57.02
	Training Data				
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
Full Dataset	99.92	99.59	100.0	99.79	99.99
Minified Dataset	99.92	99.59	100.0	99.79	100.0

Table 5.1: The comparison of the performance of Random Forests algorithm on each dataset.

5.2 XGBoost

This section presents the performance of XGBoost algorithm on the Full dataset and the Minified dataset.

5.2.1 Confusion Matrices

The confusion matrices are manifested below to describe the performance of XGBoost algorithm on the testing data. The algorithm has classified the thrombotic patients in the Full dataset with 04 false negatives and 14 false positives. The number of true negatives and true positives stands at 256 and 57, respectively, for the Full dataset. The performance of the algorithm on the Minified dataset concluded with 30 false negatives, 22 false positives, 248 true negatives and 31 true positives.
The Minified dataset contains a higher amount of false negatives and false positives and a lower number of true negatives and true positives than the Full dataset. The results have been presented in Figures 5.5 and 5.6.

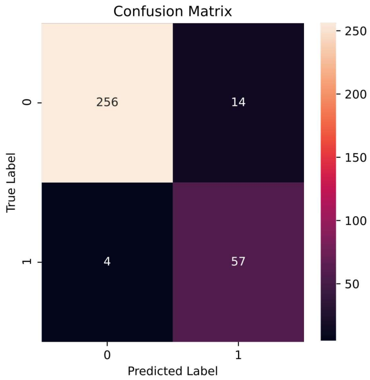


Figure 5.5: The confusion matrix for XGBoost algorithm performance on testing data (Full Dataset).

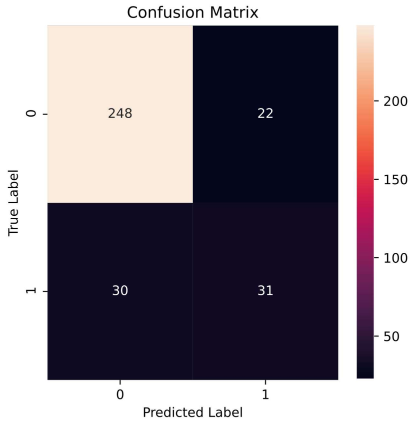


Figure 5.6: The confusion matrix for XGBoost algorithm performance on testing data (Minified Dataset).

5.2.2 Precision Recall Curves

The precision-recall curves are given below to describe the performance of XGBoost algorithm on the testing data. The precision-recall curve for the Full dataset encompasses a comparatively higher area than the Minified dataset's precision-recall curve. The optimal F1 score for the Full dataset is positioned relatively higher and farther right on the graph as compared to the optimal F1 score for the Minified dataset. A gradual increment in the recall results in a continuous decrease in the precision for both datasets. The decrement in the precision is comparatively higher for the Minified dataset. The precision-recall curves have been presented in Figures 5.7 and 5.8.

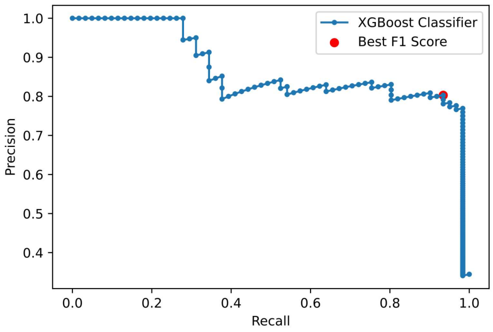


Figure 5.7: The precision-recall curve for XGBoost algorithm performance on testing data (Full Dataset).

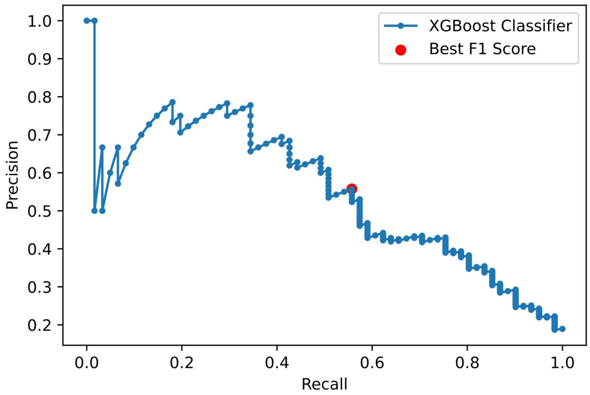


Figure 5.8: The precision-recall curve for XGBoost algorithm performance on testing data (Minified Dataset).

Testing Data					
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
Full Dataset	94.56	80.28	93.44	86.36	86.61
Minified Dataset	84.29	58.49	50.81	54.38	54.19
Training Data					
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
Full Dataset	98.33	91.72	100.0	95.68	99.98
Minified Dataset	99.84	99.18	100.0	99.59	99.99

Table 5.2: The comparison of the performance of XGBoost algorithm on each dataset.

Table 5.2 exhibits a summary of the performance of XGBoost algorithm on the testing and training data. All results are calculated with respect to the minority class that is thrombotic patients.
The algorithm generally performs better on the training data than the testing data as represented in the table. The performance on the training data is reasonably stable for both the Full dataset and the Minified dataset. The testing data yields alternating performance for the Full dataset and the Minified dataset. The algorithm achieves comparatively better scores in terms of all considered metrics on the Full dataset than the Minified dataset.

5.3 Support Vector Machines

This section presents the performance of Support Vector Machines (SVMs) algorithm on the Full dataset and the Minified dataset.

5.3.1 Confusion Matrices

The confusion matrices are presented below to describe the performance of Support Vector Machines algorithm on the testing data. The algorithm has classified the thrombotic patients in the Full dataset with 02 false negatives and 24 false positives. The number of true negatives and true positives stands at 246 and 59, respectively, for the Full dataset. The performance of the algorithm on the Minified dataset concluded with 31 false negatives, 42 false positives, 228 true negatives and 30 true positives.
The Minified dataset comprises a higher amount of false negatives and false positives and a lower number of true negatives and true positives than the Full dataset. The results have been presented in Figures 5.9 and 5.10.

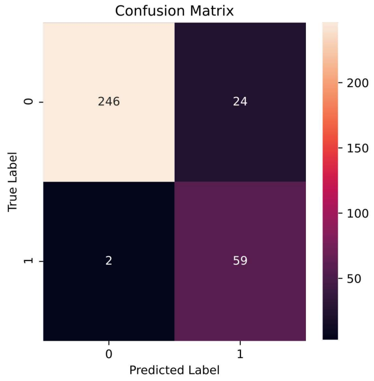


Figure 5.9: The confusion matrix for SVMs algorithm performance on testing data (Full Dataset).

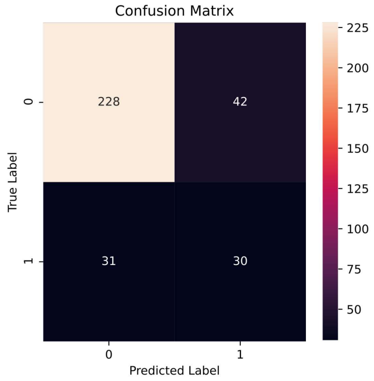


Figure 5.10: The confusion matrix for SVMs algorithm performance on testing data (Minified Dataset).

5.3.2 Precision Recall Curves

The precision-recall curves are illustrated below to describe the performance of Support Vector Machines algorithm on the testing data. The precision-recall curve for the Full dataset encompasses a comparatively higher area than the Minified dataset's precision-recall curve. The optimal F1 score for the Full dataset is positioned relatively higher and farther right on the graph as compared to the optimal F1 score for the Minified dataset. A gradual increment in the recall results in a continuous decrease in the precision for both datasets. The decrement in the precision is comparatively higher for the Minified dataset. The precision-recall curves have been presented in Figures 5.11 and 5.12.

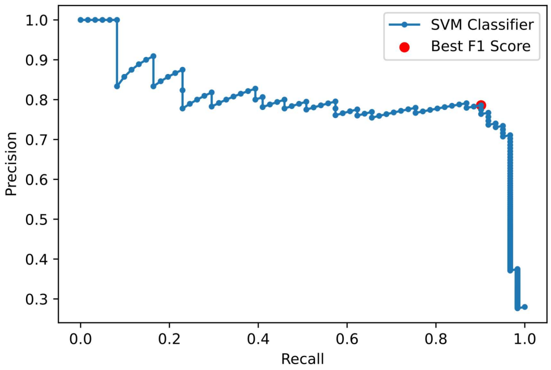


Figure 5.11: The precision-recall curve for SVMs algorithm performance on testing data (Full Dataset).

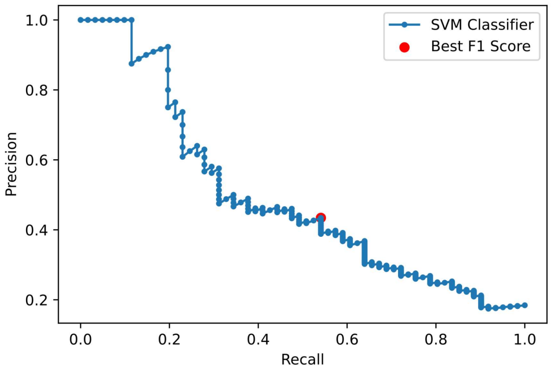


Figure 5.12: The precision-recall curve for SVMs algorithm performance on testing data (Minified Dataset).

Testing Data					
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
Full Dataset	92.14	71.08	96.72	81.94	79.65
Minified Dataset	77.94	41.66	49.18	45.11	49.07
Training Data					
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
Full Dataset	90.92	67.22	99.18	80.13	85.23
Minified Dataset	86.61	60.06	81.96	69.32	76.37

Table 5.3: The comparison of the performance of SVMs algorithm on each dataset.

Table 5.3 exhibits a summary of the performance of Support Vector Machines algorithm on the testing and training data. All results are calculated with respect to the minority class that is thrombotic patients.

The algorithm generally performs better on the training data than the testing data as represented in the table. The performance on the training data is reasonably stable for both the Full dataset and the Minified dataset. SVMs models generate a few exceptions contrary to the previous models. The algorithm scores higher accuracy, precision and F1 score on the testing data as compared to the training data for the Full dataset. The testing data yields alternating performance for the Full dataset and the Minified dataset. The algorithm achieves comparatively better scores in terms of all considered metrics on the Full dataset than the Minified dataset.

5.4 Artificial Neural Networks

This section presents the performance of Artificial Neural Networks (ANNs) algorithm on the Full dataset and the Minified dataset. Table 5.4 compares the performance of various models based on different optimising algorithms and activation functions, for the Full dataset. Table 5.5 presents outcomes for the Minified dataset. “AC” denotes the accuracy and “AUC (PR)” represents the area under the precision-recall curve. The models with the most reliable performance have been discussed further.

The information presented in tables 5.4 and 5.5 has been compared to choose a model for each dataset. The comparison considered the accuracy and the area under the precision-recall curve. The chosen activation functions, optimisers and scores have been reported in the bold text in the tables.

As described in table 5.4, Adagrad, RMSprop and Adam optimisers have produced competitive accuracy scores with the help of different activation functions. RMSprop scored the highest accuracy of 89.12% with the combination of Sigmoid function. The combination of Adagrad and Tanh activation function yielded a slightly lower accuracy than RMSprop, however, it scored higher on the AUC(PR) metric. The combination of Adagrad optimiser and Tanh activation function has been utilised to build the final model for the Full dataset.

	Full Dataset					
	ReLU		Sigmoid		**Tanh**	
	AC	AUC (PR)	AC	AUC (PR)	AC	AUC (PR)
Adagrad	88.82	64.50	81.57	16.97	**88.82**	**77.79**
RMSprop	88.51	67.92	89.12	75.54	84.89	63.09
Adam	88.51	71.46	87.00	59.78	86.70	63.69

Table 5.4: The comparison of the performance of different optimisers and activation functions for the ANNs models on the Full dataset.

	Minified Dataset					
	ReLU		**Sigmoid**		Tanh	
	AC	AUC (PR)	AC	AUC (PR)	AC	AUC (PR)
Adagrad	79.45	37.21	81.57	16.49	75.83	51.90
RMSprop	79.45	33.80	80.36	51.09	80.36	43.52
Adam	83.38	46.93	**81.87**	**50.79**	76.13	39.97

Table 5.5: The comparison of the performance of different optimisers and activation functions for the ANNs models on the Minified dataset.

Table 5.5 exhibits a thorough comparison of the performance of various optimising algorithms and activation functions on the Minified dataset. Adam optimiser and ReLU activation function have scored the highest accuracy of 83.38%. The combination of Adam optimiser and Sigmoid activation function, however, produced a higher AUC(PR) score with insignificantly lower accuracy. The Minified dataset employs the aggregate of Adam optimiser and Sigmoid activation function to generate the final model.

5.4.1 Confusion Matrices

The confusion matrices are presented below to describe the performance of the Artificial Neural Networks based models on the testing data. The algorithm has classified the thrombotic patients in the Full dataset with 12 false negatives and 25 false positives. The number of true negatives and true positives stands at 245 and 49, respectively, for the Full dataset. The performance of the algorithm on the Minified dataset concluded with 40 false negatives, 20 false positives, 250 true negatives and 21 true positives.

The Minified dataset comprises a higher amount of false negatives and true negatives and a lower number of false positives and true positives than the Full dataset. The results have been presented in Figures 5.13 and 5.14.

5.4.2 Precision Recall Curves

The precision-recall curves are presented below to describe the performance of the Artificial Neural Networks based models on the testing data. The precision-recall curve for the Full dataset encompasses a comparatively higher area than

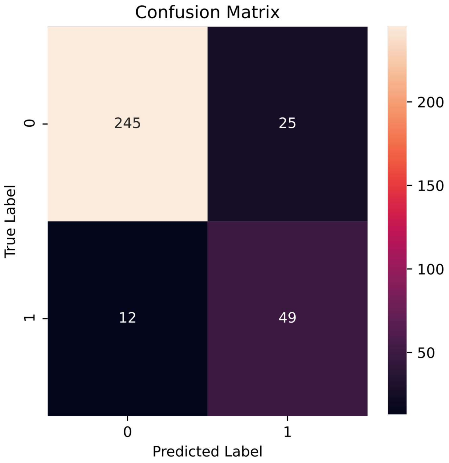


Figure 5.13: The confusion matrix for ANNs model performance on testing data (Full Dataset).

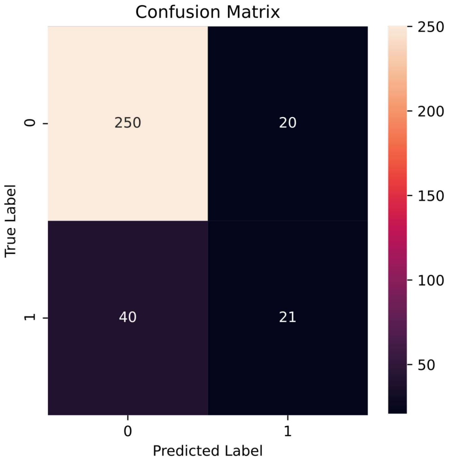


Figure 5.14: The confusion matrix for ANNs model performance on testing data (Minified Dataset).

the Minified dataset's precision-recall curve. The optimal F1 score for the Full dataset is positioned relatively higher and farther right on the graph as compared to the optimal F1 score for the Minified dataset. A gradual increment in the recall results in a continuous decrease in the precision for both datasets. The decrement in the precision is comparatively higher for the Minified dataset. The precision-recall curves have been presented in Figures 5.15 and 5.16.

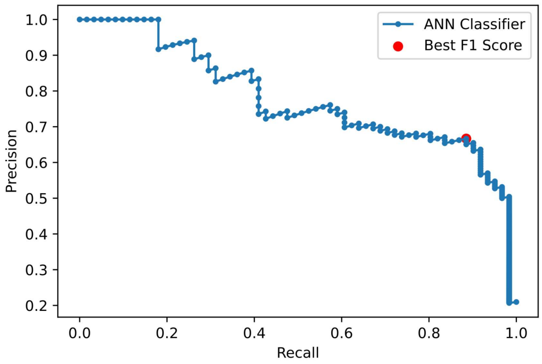


Figure 5.15: The precision-recall curve for ANNs model performance on testing data (Full Dataset).

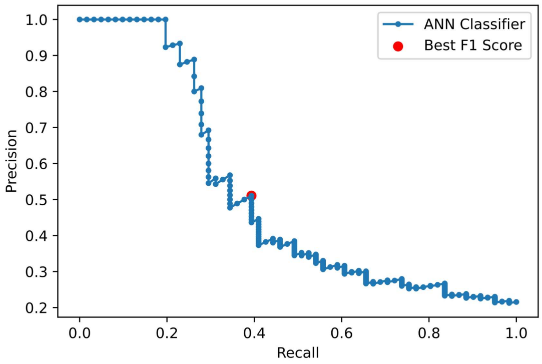


Figure 5.16: The precision-recall curve for ANNs model performance on testing data (Minified Dataset).

Table 5.6 exhibits a summary of the performance of Artificial Neural Networks models on the testing and training data. All results are calculated with respect to the minority class that is thrombotic patients.
The algorithm generally performs better on the training data than the testing data as represented in the table. The performance on the training data is reasonably stable for both the Full dataset and the Minified dataset with the exceptions of recall and AUC(PR). The testing data yields alternating performance for the Full dataset and the Minified dataset. The algorithm achieves comparatively better scores in terms of all considered metrics on the Full dataset than the Minified dataset.

Testing Data					
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
Full Dataset	88.82	66.21	80.32	72.59	77.79
Minified Dataset	81.87	51.21	34.42	41.17	50.79
Training Data					
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
Full Dataset	94.09	76.77	97.54	85.92	94.55
Minified Dataset	91.60	82.12	69.67	75.38	81.89

Table 5.6: The comparison of the performance of ANNs algorithm on each dataset.

Chapter 6

Discussion

This chapter contains the discussion of the results presented in the *Results* chapter. It is structured according to the research questions presented in the *Introduction* chapter.

6.1 Research Question I

How do standard machine learning algorithms compare in performance on the full dataset?

This study has implemented numerous machine learning models that are based on Random Forests (RF), XGBoost (XGB), Support Vector Machines (SVMs) and Artificial Neural Networks (ANNs). These models have been trained on the Full dataset to differentiate between thrombotic and non-thrombotic patients. Table 6.1 presents a summary of the outcomes of investigations conducted earlier on the Full dataset.
Tree-based algorithms, i.e., Random Forests and XGBoost, have generally performed comparatively more reliable than Support Vector Machines and Artificial Neural Networks. XGBoost has scored the highest accuracy and AUC(PR) scores on the testing data, while Random Forests has scored slightly lower. Random Forests holds the highest spot in the comparative performance on the training data and, XGBoost lies at the second spot.
Support Vector Machines and Artificial Neural Networks have secured lower accuracy and AUC(PR) than Random Forests and XGBoost on both testing and training data. Support Vector Machines has outperformed Artificial Neural Networks in reliability on the testing data, and there is an opposite outcome for the training data.
XGBoost surpassed all other algorithms in terms of performance on the F1 Score for the testing data. It was followed by Support Vector Machines, Random Forests and Artificial Neural Networks. Random Forests scored the highest on the same metric for training data. XGBoost, Artificial Neural Networks and

Support Vector Machines secured the second, third and fourth places, respectively.
The models have performed comparatively better on the training data as compared to the testing data. This behaviour is conceivably prompted by the imbalanced classes in the dataset that can trigger insignificant overfitting. Another plausible rationale behind the phenomenon is the fact that training data is not entirely unseen information to the models, contrary to the testing data that is exclusively uncomprehended.

Testing Data					
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
RF	92.74	83.63	75.40	79.31	85.70
XGB	94.56	80.28	93.44	86.36	86.61
SVMs	92.14	71.08	96.72	81.94	79.65
ANNs	88.82	66.21	80.32	72.59	77.79
Training Data					
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
RF	99.92	99.59	100.0	99.79	99.99
XGB	98.33	91.72	100.0	95.68	99.98
SVMs	90.92	67.22	99.18	80.13	85.23
ANNs	94.09	76.77	97.54	85.92	94.55

Table 6.1: The performance comparison of various algorithms on the Full dataset.

Precision-Recall curves can be exceptionally knowledgable for the datasets with imbalanced classes. Figure 6.1 illustrates the precision-recall curves for various algorithms that have been employed for this study. The illustration utilises contrasting colours to distinguish between curves of different algorithms. The figure represents the performance of models on testing data from the Full dataset.
The curves for Random Forests and XGBoost are prominently analogous and, XGBoost encompasses a slightly larger area under the curve. The curves for Support Vector Machines and Artificial Neural Networks follow separate trajectories over varying thresholds, however, they do not possess radically distinct areas under the curve. The curve for Support Vector Machines secures a slightly larger area under the curve as compared to the curve for Artificial Neural Networks. All algorithms generally provide a robust equilibrium between precision and recall.

XGBoost has yielded the most reliable efficiency in classifying thrombotic and non-thrombotic patients on solely uncomprehended testing data from the Full dataset. Artificial Neural Networks has produced the least efficient outcomes.

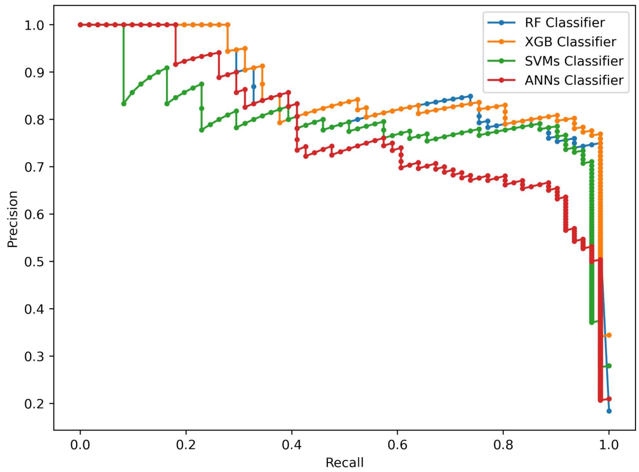


Figure 6.1: The precision-recall curves of various algorithms utilising testing data from the Full dataset.

6.2 Research Question II

How do standard machine learning algorithms compare in performance on the reduced dataset?

This section presents the performance of standard machine learning algorithms on the Minified dataset. These algorithms have been trained on the Minified dataset to differentiate between thrombotic and non-thrombotic patients. Table 6.2 presents a summary of the outcomes of investigations conducted earlier on the Minified dataset.

The tree-based algorithms, i.e., Random Forests and XGBoost, have performed comparatively better than Support Vector Machines and Artificial Neural Networks, comparable to the previous section. Random Forests has scored the highest accuracy and AUC(PR) scores on the testing data, while XGBoost has scored slightly lower than Random Forests. Random Forests outperformed all algorithms in the comparative performance on the training data and, XGBoost lies at the second spot.

Support Vector Machines and Artificial Neural Networks have achieved lower accuracy and AUC(PR) than Random Forests and XGBoost on both testing and training data. Artificial Neural Networks has outperformed Support Vector Machines in accuracy and AUC(PR) on the testing data and the training data. Support Vector Machines have scored the lowest of all algorithms.

XGBoost exceeded all other algorithms in terms of performance on the F1 Score

for the testing data. It was followed by Random Forests, Support Vector Machines and Artificial Neural Networks. Random Forests scored the highest on the same metric for training data. XGBoost, Artificial Neural Networks and Support Vector Machines secured the second, third and fourth places, respectively. It is identical to the previous section, the Full dataset.
The models have performed comparatively better on the training data as compared to the testing data, which is similar to the previous section, the Full dataset. This behaviour is conceivably prompted by the imbalanced classes in the dataset that can trigger insignificant overfitting. Another plausible rationale behind the phenomenon is the fact that training data is not entirely unseen information to the models, contrary to the testing data that is exclusively uncomprehended.

Testing Data					
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
RF	86.10	71.42	40.98	52.08	57.02
XGB	84.29	58.49	50.81	54.38	54.19
SVMs	77.94	41.66	49.18	45.11	49.07
ANNs	81.87	51.21	34.42	41.17	50.79
Training Data					
	Accuracy	Precision	Recall	F1 Score	AUC (PR)
RF	99.92	99.59	100.0	99.79	100.0
XGB	99.84	99.18	100.0	99.59	99.99
SVMs	86.61	60.06	81.96	69.32	76.37
ANNs	91.60	82.12	69.67	75.38	81.89

Table 6.2: The performance comparison of various algorithms on the Minified dataset.

Precision-Recall curves can be exceptionally knowledgable for the datasets with imbalanced classes. Figure 6.2 illustrates the precision-recall curves for various algorithms that have been employed for this study. The illustration utilises contrasting colours to distinguish between curves of different algorithms. The figure represents the performance of models on testing data from the Minified dataset.
The curves for Support Vector Machines and Artificial Neural Networks are prominently analogous and, Artificial Neural Networks encompasses a slightly larger area under the curve. The curves for Random Forests and XGBoost follow somewhat separate trajectories over varying thresholds, however, they do not possess radically distinct areas under the curve. The curve for Random Forests secures a slightly larger area under the curve as compared to the curve for XGBoost. The overall performance of different algorithms is low as compared to the previous section. Random Forests scores the highest on the AUC(PR) metric, while Support Vector Machines scores the lowest.

Random Forests has yielded the most reliable efficiency in classifying thrombotic and non-thrombotic patients on solely uncomprehended testing data from the Minified dataset. Support Vector Machines has produced the least efficient outcomes.

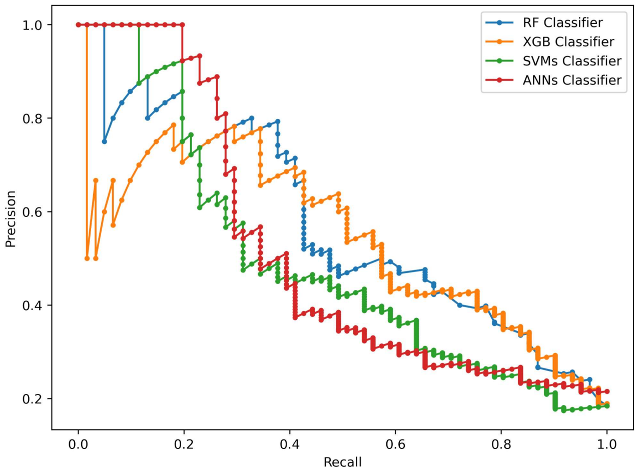


Figure 6.2: The precision-recall curves of various algorithms utilising testing data from the Minified dataset.

6.3 Research Question III

How do standard machine learning algorithms compare in performance on minimising the number of false negatives?

It can be remarkably significant to identify patients from a mixed population. It necessitates that the predictive models should possess high scores on the recall metric. A high recall score ensures that the number of false negatives is reduced to the minimum. This study performed a comparative investigation of the performance on recall metric for Random Forests, XGBoost, Support Vector Machines and Artificial Neural Networks. Table 6.3 exhibits a thorough comparison of the recall scores across different datasets.

Support Vector Machines scores the highest recall on the Full dataset when it is evaluated against the testing data. It implies that Support Vector Machines identifies the maximum number of thrombotic patients. XGBoost, Artificial Neural Networks and Random Forests secure the second, third and fourth positions, respectively. The performance on the training data is comparatively higher in resemblance to the previous trend. Random Forests and XGBoost

Full Dataset				
	RF	XGB	SVMs	ANNs
Testing Data	75.40	93.44	96.72	80.32
Training Data	100.0	100.0	99.18	97.54
Minified Dataset				
	RF	XGB	SVMs	ANNs
Testing Data	40.98	50.81	49.18	34.42
Training Data	100.0	100.0	81.96	69.67

Table 6.3: The comparison of the Recall scores of different algorithms on each dataset.

acquire the first and second spots while Support Vector Machines and Artificial Neural Networks hold the third and fourth positions, respectively.

The performance of the algorithms on the Minified dataset is comparatively lower than the performance on the Full dataset. XGBoost scores the highest recall on the Minified dataset when evaluated against the testing data. Support Vector Machines, Random Forests and Artificial Neural Networks secure the second, third and fourth positions, respectively. The performance on the training data is comparatively higher in resemblance to the previous trend. Random Forests and XGBoost acquire the first and second spots while Support Vector Machines and Artificial Neural Networks hold the third and fourth positions, respectively.

The algorithms included in the study generally perform better on the Full dataset in reducing the number of false negatives as compared to the performance on the Minified dataset. It can be explained by the lower amount of data contained in the Minified dataset.

Chapter 7

Conclusion

This chapter presents the conclusions from the investigation. It is separated into Conclusions and Future Work sections.

This study has conducted a comparative investigation to compare the performance of standard machine learning algorithms on the *RI Schedule* dataset. The study includes Random Forests, XGBoost, Support Vector Machines and Artificial Neural Networks in the quantitative comparison. It has been organised into three separate investigations that examine the performance on the Full dataset, Minified dataset and the reduction of false negatives.

7.1 Conclusions

The study concluded that XGBoost yields the most reliable performance in distinguishing between thrombotic and non-thrombotic patients on the testing data for the Full dataset. It scored 94.56 on the accuracy metric. The second, third and fourth places are held by Random Forests, Support Vector Machines and Artificial Neural Networks, respectively. They scored 92.74, 92.14 and 88.82, respectively, on the accuracy metric. Random Forests, XGBoost, Artificial Neural Networks and, Support Vector Machines secure the first, second, third and fourth positions, respectively, on the training data. They obtained scores of 99.92, 98.33, 94.09 and 90.92, respectively, on the accuracy metric. The models performed relatively better on the training data as compared to the testing data.

Random Forests has been the most efficient algorithm in separating thrombotic and non-thrombotic patients on the testing data for the Minified dataset. It scored 86.10 on the accuracy metric. It is followed by XGBoost, Artificial Neural Networks and Support Vector Machines in the descending order of performance. They scored 84.29, 81.87 and 77.94, respectively, on the accuracy metric. The ranking of the models for the training data is identical, i.e., Random Forests, XGBoost, Artificial Neural Networks and Support Vector Machines. They obtained scores of 99.92, 99.84, 91.60 and 86.61, respectively, on the accuracy metric. The models performed relatively better on the training data as compared to the testing data.

The study has employed the recall metric to indicate the number of false negatives when classifying thrombotic and non-thrombotic patients. Support Vector Machines obtained the highest recall on the testing data for the Full dataset by scoring 96.72. It is followed by XGBoost, Artificial Neural Networks and Random Forests. They scored 93.44, 80.32 and 75.40, respectively. XGBoost lead the performance ranking for the testing data of the Minified dataset by scoring 50.81. Support Vector Machines, Random Forests and Artificial Neural Networks secured the second, third and fourth positions by scoring 49.18, 40.98 and 34.42, respectively. The performance ranking on the training data for both Full and Minified datasets is identical, i.e., Random Forests, XGBoost, Support Vector Machines and Artificial Neural Networks in the descending order of the recall performance. They scored 100.0, 100.0, 99.18 and 97.54, respectively, on the Full dataset. The respective scores for the Minified dataset are 100.0, 100.0, 81.96 and 69.67. The models generally perform poorly on the Minified dataset as compared to the performance on the Full dataset.

7.2 Future Work

The Minified dataset has been designed to comprise only essential information regarding the patients. It assists medical practitioners in performing a prediction without requiring most of the clinical tests. It implies that reliable performance on the Minified dataset can be remarkably beneficial in real-world applications. The conclusions from this study illustrate that the predictive models have a comparatively weaker performance on the Minified dataset. This decline in performance can be further investigated.

The study has developed predictive models that can be utilised in the healthcare sector to predict thrombosis. It necessitates that the number of false negatives is minimised, which can be achieved by maximising the score on the recall metric. The performance of the recall metric can be enhanced with the assistance of dedicated hyperparameter tuning and custom objective functions.

The overall performance of the models can be improved and validated further with the help of additional clinical data.

Bibliography

[1] M. Yang and T. Tan, "Formation of thrombosis and its potential diagnosis and treatment with optoacoustic technology," in *Proceedings of the Third International Conference on Medical and Health Informatics 2019*, ser. ICMHI 2019, New York, NY, USA: Association for Computing Machinery, 2019, pp. 1–5.

[2] I. A. Næss, S. Christiansen, P. Romundstad, S. Cannegieter, F. R. Rosendaal, and J. Hammerstrøm, "Incidence and mortality of venous thrombosis: A population-based study," *Journal of Thrombosis and Haemostasis*, vol. 5, no. 4, pp. 692–699, 2007.

[3] F. R. Rosendaal, "Causes of venous thrombosis," *Thrombosis Journal*, vol. 14, no. 1, p. 24, 2016.

[4] R. L. Bick, "Introduction to thrombosis: Proficient and cost-effective approaches to thrombosis," *Hematology/Oncology Clinics of North America*, vol. 17, no. 1, pp. 1–8, 2003.

[5] T. Baglin, R. Luddington, K. Brown, and C. Baglin, "Incidence of recurrent venous thromboembolism in relation to clinical and thrombophilic risk factors: Prospective cohort study," *The Lancet*, vol. 362, no. 9383, pp. 523–526, 2003, ISSN: 0140-6736.

[6] P. A. Kyrle, E. Minar, C. Bialonczyk, M. Hirschl, A. Weltermann, and S. Eichinger, "The risk of recurrent venous thromboembolism in men and women," *New England Journal of Medicine*, vol. 350, no. 25, pp. 2558–2563, 2004, PMID: 15201412.

[7] S. C. Christiansen, W. M. Lijfering, F. M. Helmerhorst, F. R. Rosendaal, and S. C. Cannegieter, "Sex difference in risk of recurrent venous thrombosis and the risk profile for a second event," *Journal of Thrombosis and Haemostasis*, vol. 8, no. 10, pp. 2159–2168, 2010.

[8] S. Z. Goldhaber and H. Bounameaux, "Pulmonary embolism and deep vein thrombosis," *The Lancet*, vol. 379, no. 9828, pp. 1835–1846, 2012, ISSN: 0140-6736.

[9] M. Nordström, B. Lindblad, D. Bergqvist, and T. Kjellström, "A prospective study of the incidence of deep-vein thrombosis within a defined urban population," *Journal of Internal Medicine*, vol. 232, no. 2, pp. 155–160, 1992.

[10] C.-H. Lee, C.-L. Cheng, L.-J. Lin, L.-M. Tsai, and Y.-H. K. Yang, "Epidemiology and predictors of short-term mortality in symptomatic venous thromboembolism," *Circulation Journal*, vol. 75, no. 8, pp. 1998–2004, 2011.

[11] J. W. Blom, C. J. M. Doggen, S. Osanto, and F. R. Rosendaal, "Malignancies, prothrombotic mutations, and the risk of venous thrombosis," *JAMA*, vol. 293, no. 6, pp. 715–722, Feb. 2005, ISSN: 0098-7484.

[12] Antiplatelet Trialists' Collaboration *et al.*, "Collaborative overview of randomised trials of antiplatelet therapy prevention of death, myocardial infarction, and stroke by prolonged antiplatelet therapy in various categories of patients," *BMJ*, vol. 308, no. 6921, pp. 81–106, 1994, ISSN: 0959-8138.

[13] P. W. Sholar and W. R. Bell, "Thrombolytic therapy for inferior vena cava thrombosis in paroxysmal nocturnal hemoglobinuria," *Annals of Internal Medicine*, vol. 103, no. 4, pp. 539–541, 1985, PMID: 4037558.

[14] F. Jiang, Y. Jiang, H. Zhi, Y. Dong, H. Li, S. Ma, Y. Wang, Q. Dong, H. Shen, and Y. Wang, "Artificial intelligence in healthcare: Past, present and future," *Stroke and Vascular Neurology*, vol. 2, no. 4, pp. 230–243, 2017, ISSN: 2059-8688.

[15] E. Alpaydin, *Introduction to Machine Learning*. Cambridge, MA, USA: MIT Press, 2020, ISBN: 9780262043793.

[16] M. Kuhn and K. Johnson, *Applied Predictive Modeling*, ser. SpringerLink : Bücher. New York, NY, USA: Springer, 2013, ISBN: 9781461468493.

[17] T. Nafee, C. M. Gibson, R. Travis, M. K. Yee, M. Kerneis, G. Chi, F. AlKhalfan, A. F. Hernandez, R. D. Hull, A. T. Cohen, R. A. Harrington, and S. Z. Goldhaber, "Machine learning to predict venous thrombosis in acutely ill medical patients," *Research and Practice in Thrombosis and Haemostasis*, vol. 4, no. 2, pp. 230–237, 2020.

[18] S. Liu, F. Zhang, L. Xie, Y. Wang, Q. Xiang, Z. Yue, Y. Feng, Y. Yang, J. Li, L. Luo, and C. Yu, "Machine learning approaches for risk assessment of peripherally inserted central catheter-related vein thrombosis in hospitalized patients with cancer," *International Journal of Medical Informatics*, vol. 129, pp. 175–183, 2019, ISSN: 1386-5056.

[19] P. Ferroni, F. M. Zanzotto, N. Scarpato, S. Riondino, F. Guadagni, and M. Roselli, "Validation of a machine learning approach for venous thromboembolism risk prediction in oncology," *Disease Markers*, vol. 2017, Sep. 2017, ISSN: 0278-0240.

[20] J. Maiora, B. Ayerdi, and M. Graña, "Random forest active learning for aaa thrombus segmentation in computed tomography angiography images," *Neurocomputing*, vol. 126, pp. 71–77, 2014, Recent trends in Intelligent Data Analysis Online Data Processing, ISSN: 0925-2312.

[21] C. A. Morariu, M. Thomas, J. Pauli, D. S. Dohle, and K. Tsagakis, "Sequential vs. batch machine-learning with evolutionary hyperparameter optimization for segmenting aortic dissection thrombus," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, 2016, pp. 1189–1194.

[22] J. Willan, H. Katz, and D. Keeling, "The use of artificial neural network analysis can improve the risk-stratification of patients presenting with suspected deep vein thrombosis," *British Journal of Haematology*, vol. 185, no. 2, pp. 289–296, 2019.

[23] J. Vilhena, H. Vicente, M. R. Martins, J. Grañeda, F. Caldeira, R. Gusmão, J. Neves, and J. Neves, "An artificial intelligence approach to thrombophilia risk," *International Journal of Reliable and Quality E-Healthcare*, vol. 6, no. 2, pp. 49–69, 2017, ISSN: 2160-9551.

[24] Y. Yang, X. Wang, Y. Huang, N. Chen, J. Shi, and T. Chen, "Ontology-based venous thromboembolism risk factors mining and model developing from medical records," in *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2018, pp. 1669–1672.

[25] B. Semiz, S. Hersek, M. B. Pouyan, C. Partida, L. Blazquez-Arroyo, V. Selby, G. Wieselthaler, J. M. Rehg, L. Klein, and O. T. Inan, "Detecting suspected pump thrombosis in left ventricular assist devices via acoustic analysis," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 7, pp. 1899–1906, 2020.

[26] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proceedings of the 23rd International Conference on Machine Learning*, ser. ICML '06, Pittsburgh, Pennsylvania, USA: Association for Computing Machinery, 2006, pp. 161–168, ISBN: 1595933832.

[27] M. Hassan and M. Hamada, "Performance comparison of feed-forward neural networks trained with different learning algorithms for recommender systems," *Computation*, vol. 5, no. 3, 2017, ISSN: 2079-3197.

[28] P. Douglas, S. Harris, A. Yuille, and M. S. Cohen, "Performance comparison of machine learning algorithms and number of independent components used in fmri decoding of belief vs. disbelief," *NeuroImage*, vol. 56, no. 2, pp. 544–553, 2011, Multivariate Decoding and Brain Reading, ISSN: 1053-8119.

[29] L. Chih-Chin and T. Ming-Chi, "An empirical performance comparison of machine learning methods for spam e-mail categorization," in *Fourth International Conference on Hybrid Intelligent Systems (HIS'04)*, 2004, pp. 44–48.

[30] S. A. R. Shah and B. Issac, "Performance comparison of intrusion detection systems and application of machine learning to snort system," *Future Generation Computer Systems*, vol. 80, pp. 157–170, 2018, ISSN: 0167-739X.

[31] F. Salazar and B. M. Crookston, "A performance comparison of machine learning algorithms for arced labyrinth spillways," *Water*, vol. 11, no. 3, 2019, ISSN: 2073-4441.

[32] W. McKinney, "Data Structures for Statistical Computing in Python," in *Proceedings of the 9th Python in Science Conference*, S. van der Walt and J. Millman, Eds., 2010, pp. 56–61.

[33] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del R'ıo, M. Wiebe, P. Peterson, P. G'erard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020.

[34] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[35] H. Alkharusi, "Categorical variables in regression analysis: A comparison of dummy and effect coding," *International Journal of Education*, vol. 4, pp. 202–210, Apr. 2012.

[36] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[37] M. Feurer and F. Hutter, "Hyperparameter optimization," in *Automated Machine Learning: Methods, Systems, Challenges.* Cham: Springer International Publishing, 2019, pp. 3–33, ISBN: 978-3-030-05318-5.

[38] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16, San Francisco, California, USA: Association for Computing Machinery, 2016, pp. 785–794, ISBN: 9781450342322.

[39] F. Chollet *et al.*, *Keras*, `https://keras.io`, 2015.

Appendix A

Dataset Features

A.1 Original Features of *RI Schedule* Dataset

Table A.1: A complete list of the features in the *RI Schedule* dataset.

Features	
Patient ID	Inclusion Month
Weight KG	Height
Wells Score Fastlege	Exclusion Criteria HB
Sex	Age Inclusion
Exclusion Criteria Investigation 2 Hours	Exclusion Criteria Klexane
Exclusion Criteria Active Cancer Disease	Exclusion Criteria Suspect Lung Embolism
Exclusion Criteria Active Bleeding	Exclusion Criteria Threatening Circulation
Exclusion Criteria Improper Print	Exclusion Criteria Logical Factors
Exclusion Criteria Comorbidities	Exclusion Criteria Do Not Want Printed
Exclusion Criteria GFR	Exclusion Criteria HB
Contraindications Rivaroxaban Significant Risk Factor	Contraindications Rivaroxaban Current Treatment
Contraindications Rivaroxaban Hepatic Disease	Contraindications Rivaroxaban Pregnancy
Exclusion Criteria Current Treatment	BaselineWellsScore Nurse Active Cancer
BaselineWellsScore Nurse Paralysis	BaselineWellsScore Nurse Bed
BaselineWellsScore Nurse Tenderness	BaselineWellsScore Nurse Swelling Throughout UEX

BaselineWellsScore Nurse Swelling Leg	BaselineWellsScore Nurse Pitting Edema
BaselineWellsScore Nurse Venous Collateral	BaselineWellsScore Nurse Alternative Diagnosis
BaselineWellsScore Nurse Total Score	BaselineWellsScore Doctor Active Cancer
BaselineWellsScore Doctor Paralysis	BaselineWellsScore Doctor Bed
BaselineWellsScore Doctor Tenderness	BaselineWellsScore Doctor Swelling Throughout UEX
BaselineWellsScore Doctor Swelling Leg	BaselineWellsScore Doctor Pitting Edema
BaselineWellsScore Doctor Venous Collateral	BaselineWellsScore Doctor Alternative Diagnosis
BaselineWellsScore Doctor Total Score	Klexane Fragmin
Klexane Dose Field	Klexane Number Of Times
Fragmin Dose Field	Fragmin Number Of Times
Ongoing Anticoagulation If Yes Specify	Risk Factors
Risk Factors Previous DVT Le	Risk Factors Previous DVT Date1 Relative 21628
Risk Factors Previous DVT Date2 Relative 21628	Risk Factors Previous DVT Date3 Relative 21628
Risk Factors Previous Le Date1 Relative 21628	Risk Factors Previous Le Date2 Relative 21628
Risk Factors DVT Le First Degree Relatives	Risk Factors P Pills Type
Risk Factors HRT Type	Risk Factors Active Cancer Last 6 Months
Risk Factors Cancer Type	Risk Factors Myeloproliferative Disease
Risk Factors Thrombophilia	Risk Factors Thrombophilia APC
Risk Factors Thrombophilia Factor V Heterozygous	Risk Factors Thrombophilia Antithrombin
Risk Factors Thrombophilia Protein C	Risk Factors Thrombophilia Protein S
Risk Factors Thrombophilia Antiphospholipid	Risk Factors Thrombophilia Prothrombin
Risk Factors Pregnancy	Risk Factors Birth Last 12 Weeks
Risk Factors Immobilization Specify Operation	Risk Factors Immobilization Other Surgery Specify
Risk Factors Immobilization Thromboprophylaxis	Risk Factors Immobilization LMVH

Risk Factors Immobilization Rivaroxaban	Risk Factors Immobilization Apixaban
Risk Factors Immobilization Dabigatran	Risk Factors Immobilization Thromboprophylaxis How Many Days
Risk Factors Immobilization Trauma	Risk Factors Immobilization Neurological Disease
Risk Factors Immobilization Physical Inactivity	Risk Factors Immobilization Air Travel
Risk Factors Immobilization Car Train	Risk Factors Smoker
Clinic Symptoms Duration	Clinic Symptoms Pain
Clinic Symptoms Swelling	Clinic Symptoms Erythema
Clinic Vital Temp	Clinic Vital Pulse
Clinic Vital Blood Pressure Systolic	Clinic Vital Blood Pressure Diastolic
Clinic Measurement Leg Knee Left	Clinic Measurement Leg Knee Right
Knee Difference V H	Clinic Measurement Leg Ankle Left
Clinic Measurement Leg Ankle Right	Ankle Difference V H
Clinic Suspected Active Bleeding	Clinic Suspected Active Bleeding Bruises
Clinic Suspected Active Bleeding Hematuria	Clinic Suspected Active Bleeding Epistaxis
Clinic Suspected Active Bleeding Blood in Feces	Clinic Suspected Infection
Clinic Suspected Infection Fever	Clinic Suspected Infection Sweat
Clinic Suspected Infection Frostbite	Clinic Suspected Infection Reduced General Condition
Clinic Suspected Infection Suspected Erysipelas	Clinic Suspected Infection Suspected Erysipelas High Fever
Clinic Suspected Infection Suspected Erysipelas Well Defined	Clinic Blood Test Results HB
Clinic Blood Test Results Platelets	Clinic Blood Test Results CRP
Clinic Blood Test Results GFR	Clinic Blood Test Results Creatinine
Clinic Blood Test Results D Dimer	Clinic Blood Test Results ASAT
Clinic Blood Test Results ALAT	Clinic Blood Test Results Bilirubin
Clinic Rivaroxaban Received Not Received	Clinic Rivaroxaban Number Of Tablets
Clinic Nurse Assessment	Clinic Doctor Assessment
Time Course Supervised Legevakt Fastlege Relative 21628	Time Course Arrival Emergency Room Relative 21628
Date Visit 1 Relative 21628	Visit 1 Measurement Leg Knee Left

Visit 1 Measurement Leg Knee Right	Difference Leg Knee V H
Visit 1 Measurement Leg Ankle Left	Visit 1 Measurement Leg Ankle Right
Difference Leg Ankle V H	Visit 1 HB
Visit 1 D Dimer	Date Visit 2 Relative 21628
Visit 2 Bleeding 1 Treatment	Visit 2 New UL Examination
Visit 2 New UL Examination DVT Proven	Visit 2 Worsening Symptoms
Visit 2 Development Symptoms Pulmonary Embolism	Visit 2 Development Symptoms Pulmonary Embolism Specify
Visit 2 Development Symptoms Pulmonary Embolism Treatment	Visit 2 Bleeding
Visit 2 Bleeding 1 Date Relative 21628	Visit 2 Bleeding 1 Type
Visit 2 Bleeding 1 Localization	Visit 2 Bleeding 1 Location Specify
Visit 2 Bleeding 1 HB	Visit 2 Bleeding 2 Date Relative 21628
Visit 2 Bleeding 2 Type	Visit 2 Bleeding 2 Localization
Visit 2 Bleeding 2 Location Specify	Visit 2 Bleeding 2 Treatment
Visit 2 Bleeding 2 HB	Date Visit 3 Relative 21628
Control Performed	Visit 3 New VTE VTE 1 Time Relative 21628
Visit 3 New VTE VTE 1 DVT LE	Visit 3 New VTE Treatment
Visit 3 Deaths	Visit 3 Deaths Related To Bleeding
Visit 3 Deaths Unknown Cause	Visit 3 Deaths Death Cause
Visit 3 Deaths Autopsy	Visit 3 Proven VTE After Day 2
Visit 3 Deaths Related To Recurrence Of VTE	Start Of Anticoagulant Last 90 Days
If Yes What Medicine	Cause Of Startup
Visit 3 Malignancy Detected	Visit 3 Malignancy Detected Type
Date UL Relative 21628	UL Proven VTE
Date Relative 21628	Anticoagulation UFH Start Up Relative 21628
Anticoagulation UFH End Date Relative 21628	Anticoagulation LMWH Start Up Relative 21628
Anticoagulation LMWH End Date Relative 21628	Anticoagulation Xarelto Start Up Relative 21628
Anticoagulation Xarelto End Date Relative 21628	Anticoagulation Marevan Start Up Relative 21628
Anticoagulation Marevan End Date Relative 21628	Anticoagulation Eliquis Start Up Relative 21628
Anticoagulation Eliquis End Date Relative 21628	Anticoagulation Thrombolysis Start Up 21628

Anticoagulation Thrombolysis End Date 21628	Anticoagulation Other Start Up Relative 21628
Anticoagulation Other End Date Relative 21628	Anticoagulation Type
Continues With Ongoing Treatment	Continues With Ongoing Treatment Specify

A.2 Features Dropped From *Full Dataset*

Table A.2: A list of the features dropped from the Full dataset.

Features	
Wells Score Fastlege	Exclusion Criteria Investigation 2 Hours
Exclusion Criteria Klexane	Exclusion Criteria Active Cancer Disease
Exclusion Criteria Active Bleeding	Exclusion Criteria Threatening Circulation
Exclusion Criteria Improper Print	Exclusion Criteria Logical Factors
Exclusion Criteria Comorbidities	Exclusion Criteria Do Not Want Printed
Exclusion Criteria GFR	Exclusion Criteria HB
Contraindications Rivaroxaban Significant Risk Factor	Contraindications Rivaroxaban Current Treatment
Contraindications Rivaroxaban Hepatic Disease	Contraindications Rivaroxaban Pregnancy
Baseline Wells Score Nurse Active Cancer	Baseline Wells Score Nurse Paralysis
Baseline Wells Score Nurse Bed	Baseline Wells Score Nurse Tenderness
Baseline Wells Score Nurse Swelling Throughout UEX	Baseline Wells Score Nurse Swelling Leg
Baseline Wells Score Nurse Pitting Edema	Baseline Wells Score Nurse Venous Collateral
Baseline Wells Score Nurse Alternative Diagnosis	Baseline Wells Score Nurse Total Score
Clinic Nurse Assessment	Clinic Doctor Assessment
Time Course Supervised Legevakt Fastlege Relative 21628	Time Course Arrival Emergency Room Relative 21628
Control Performed	Anticoagulation UFH Start Up Relative 21628
Anticoagulation UFH End Date Relative 21628	Anticoagulation LMWH Start Up Relative 21628

Anticoagulation LMWH End Date Relative 21628	Anticoagulation Xarelto Start Up Relative 21628
Anticoagulation Xarelto End Date Relative 21628	Anticoagulation Marevan Start Up Relative 21628
Anticoagulation Marevan End Date Relative 21628	Anticoagulation Eliquis Start Up Relative 21628
Anticoagulation Eliquis End Date Relative 21628	Anticoagulation Thrombolysis Start Up 21628
Anticoagulation Thrombolysis End Date 21628	Anticoagulation Other Start Up Relative 21628
Anticoagulation Other End Date Relative 21628	Anticoagulation Type
Continues With Ongoing Treatment	Continues With Ongoing Treatment Specify

A.3 Features Dropped From *Minified Dataset*

Table A.3: A list of the features dropped from the Minified dataset.

Features	
Wells Score Fastlege	Exclusion Criteria Investigation 2 Hours
Exclusion Criteria Klexane	Exclusion Criteria Active Cancer Disease
Exclusion Criteria Active Bleeding	Exclusion Criteria Threatening Circulation
Exclusion Criteria Improper Print	Exclusion Criteria Logical Factors
Exclusion Criteria Comorbidities	Exclusion Criteria Do Not Want Printed
Exclusion Criteria GFR	Exclusion Criteria HB
Contraindications Rivaroxaban Significant Risk Factor	Contraindications Rivaroxaban Current Treatment
Contraindications Rivaroxaban Hepatic Disease	Contraindications Rivaroxaban Pregnancy
Baseline Wells Score Nurse Active Cancer	Baseline Wells Score Nurse Paralysis
Baseline Wells Score Nurse Bed	Baseline Wells Score Nurse Tenderness
Baseline Wells Score Nurse Swelling Throughout UEX	Baseline Wells Score Nurse Swelling Leg
Baseline Wells Score Nurse Pitting Edema	Baseline Wells Score Nurse Venous Collateral

Baseline Wells Score Nurse Alternative Diagnosis	Baseline Wells Score Nurse Total Score
Clinic Nurse Assessment	Clinic Doctor Assessment
Time Course Supervised Legevakt Fastlege Relative 21628	Time Course Arrival Emergency Room Relative 21628
Control Performed	Anticoagulation UFH Start Up Relative 21628
Anticoagulation UFH End Date Relative 21628	Anticoagulation LMWH Start Up Relative 21628
Anticoagulation LMWH End Date Relative 21628	Anticoagulation Xarelto Start Up Relative 21628
Anticoagulation Xarelto End Date Relative 21628	Anticoagulation Marevan Start Up Relative 21628
Anticoagulation Marevan End Date Relative 21628	Anticoagulation Eliquis Start Up Relative 21628
Anticoagulation Eliquis End Date Relative 21628	Anticoagulation Thrombolysis Start Up 21628
Anticoagulation Thrombolysis End Date 21628	Anticoagulation Other Start Up Relative 21628
Anticoagulation Other End Date Relative 21628	Anticoagulation Type
Continues With Ongoing Treatment	Continues With Ongoing Treatment Specify
Inclusion Month	Exclusion Criteria Current Treatment
Klexane Fragmin	Klexane Dose Field
Klexane Number Of Times	Fragmin Dose Field
Fragmin Number Of Times	Ongoing Anticoagulation If Yes Specify
Risk Factors	Risk Factors Previous DVT Date1 Relative 21628
Risk Factors Previous DVT Date2 Relative 21628	Risk Factors Previous DVT Date3 Relative 21628
Risk Factors Previous Le Date1 Relative 21628	Risk Factors Previous Le Date2 Relative 21628
Risk Factors Thrombophilia	Risk Factors Immobilization Thromboprophylaxis
Risk Factors Immobilization LMVH	Risk Factors Immobilization Rivaroxaban
Risk Factors Immobilization Apixaban	Risk Factors Immobilization Dabigatran
Risk Factors Immobilization Thromboprophylaxis How Many Days	Knee Difference V H

Ankle Difference V H	Clinic Suspected Active Bleeding
Clinic Suspected Active Bleeding Bruises	Clinic Suspected Active Bleeding Hematuria
Clinic Suspected Active Bleeding Epistaxis	Clinic Suspected Active Bleeding Blood in Feces
Clinic Rivaroxaban Received Not Received	Clinic Rivaroxaban Number Of Tablets
Date Visit 1 Relative 21628	Visit 1 Measurement Leg Knee Left
Visit 1 Measurement Leg Knee Right	Difference Leg Knee V H
Visit 1 Measurement Leg Ankle Left	Visit 1 Measurement Leg Ankle Right
Difference Leg Ankle V H	Visit 1 HB
Visit 1 D Dimer	Date Visit 2 Relative 21628
Visit 2 Bleeding 1 Treatment	Visit 2 New UL Examination
Visit 2 New UL Examination DVT Proven	Visit 2 Worsening Symptoms
Visit 2 Development Symptoms Pulmonary Embolism	Visit 2 Development Symptoms Pulmonary Embolism Specify
Visit 2 Development Symptoms Pulmonary Embolism Treatment	Visit 2 Bleeding
Visit 2 Bleeding 1 Date Relative 21628	Visit 2 Bleeding 1 Type
Visit 2 Bleeding 1 Localization	Visit 2 Bleeding 1 Location Specify
Visit 2 Bleeding 1 HB	Visit 2 Bleeding 2 Date Relative 21628
Visit 2 Bleeding 2 Type	Visit 2 Bleeding 2 Localization
Visit 2 Bleeding 2 Location Specify	Visit 2 Bleeding 2 Treatment
Visit 2 Bleeding 2 HB	Date Visit 3 Relative 21628
Visit 3 New VTE VTE 1 Time Relative 21628	Visit 3 New VTE VTE 1 DVT LE
Visit 3 New VTE Treatment	Visit 3 Deaths
Visit 3 Deaths Related To Bleeding	Visit 3 Deaths Unknown Cause
Visit 3 Deaths Death Cause	Visit 3 Deaths Autopsy
Visit 3 Proven VTE After Day 2	Visit 3 Deaths Related To Recurrence Of VTE
Start Of Anticoagulant Last 90 Days	If Yes What Medicine
Cause Of Startup	Visit 3 Malignancy Detected
Visit 3 Malignancy Detected Type	Date UL Relative 21628
Date Relative 21628	