A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF ÇANKIRI KARATEKİN UNIVERSITY

HEART DISEASE PREDICTION USING MACHINE LEARNING

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRONICS AND COMPUTER ENGINEERING

BY

MOHAMED ALI YOUSSOUF

ÇANKIRI

2024

HEART DISEASE PREDICTION USING MACHINE LEARNING

By Mohamed ALI YOUSSOUF

April 2024

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science

Advisor **:** Asst. Prof. Dr. Fuat TÜRK

Examining Committee Members:

Chairman **:** Asst. Prof. Dr. Selim BUYRUKOĞLU
Electronics and Computer Engineering
Çankırı Karatekin University

Member **:** Asst. Prof. Dr. Mustafa KARHAN
Electronics and Computer Engineering
Çankırı Karatekin University

Member **:** Asst. Prof. Dr. Fuat TÜRK
Computer Engineering
Kırıkkale University

Approved for the Graduate School of Natural and Applied Sciences

Prof. Dr. Hamit ALYAR
Director of Graduate School

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Mohamed ALI YOUSSOUF

ABSTRACT

HEART DISEASE PREDICTION USING MACHINE LEARNING

Mohamed ALI YOUSSOUF

Master of Science in Electronics and Computer Engineering

Advisor: Asst. Prof. Dr. Fuat TÜRK

April 2024

Heart disease is currently the leading cause of death worldwide and has become one of the most challenging diseases to diagnose. Consequently, the use of machine learning techniques for rapid disease diagnosis is recommended, which saves time for both doctors and patients. I conducted a comparative study on heart disease prediction using machine learning algorithms such as Logistic Regression, Random Forest, Decision Tree, SVM (Support Vector Machine), Naive Bayes (NB), and KNN (k-nearest neighbors). The accuracy rates obtained were 98%, 78%, 98%, 60%, 66%, and 63% for Random Forest, Logistic Regression, Decision Tree, KNN, NB, and SVM, respectively. This study explored more than just one approach to heart disease prediction. Additionally, I investigated ensemble learning techniques such as bagging, stacking, and boosting on a publicly available dataset of patients with heart disease. I found that combining these techniques significantly increased the accuracy of the predictions: 98% for stacking, 99% for bagging, and 97% for boosting. This demonstrates how integrating various techniques can enhance the precision of heart disease predictions.

2024, 54 pages

Keywords: Heart disease prediction, Random forest, Logistic regression, Support vector machine, Naive Bayes, K-nearest neighbors, Ensemble learning

ÖZET

MAKİNE ÖĞRENMESİ KULLANARAK KALP HASTALIĞI TAHMİNİ

Mohamed ALI YOUSSOUF
Elektronik ve Bilgisayar Mühendisliği, Yüksek Lisans
Tez Danışmanı: Dr. Öğr. Üyesi Fuat TÜRK
Nisan 2024

Kalp hastalıkları şu anda dünya genelinde ölümlerin önde gelen nedenleri arasında yer almaktadır ve teşhisi giderek daha karmaşık bir süreç haline gelmiştir. Bu nedenle, hastalıkların hızlı teşhisi için makine öğrenimi tekniklerinin kullanılması önerilmektedir. Bu yaklaşım, hem doktorlar hem de hastalar için zaman tasarrufu sağlamayı amaçlamaktadır. Kalp hastalığı tahminine yönelik olarak farklı makine öğrenimi algoritmalarını içeren karşılaştırmalı bir çalışma gerçekleştirdik. Bu algoritmalar arasında lojistik regresyon, rastgele orman, karar ağacı, destek vektör makinesi (DVM), Naive Bayes ve k-en yakın komşular (KYK) bulunmaktadır. Rastgele orman, lojistik regresyon, karar ağacı, KYK, Naive Bayes ve DVM'nin doğruluk oranları sırasıyla %98, %78, %98, %60, %66 ve %63'tür. Bu çalışma, yalnızca bir yönteme değil, kalp hastalığı tahminine yönelik çeşitli yaklaşımları inceledi. Ayrıca, kalp hastalığına sahip hastalar için kullanılabilir bir veri seti üzerinde bagging, stacking ve boosting gibi topluluk öğrenme tekniklerini keşfettim. Bu tekniklerin bir araya getirilmesinin, tahminlerin doğruluğunu önemli ölçüde artırdığını gözlemledik. Stacking, bagging ve boosting için sırasıyla %98, %99 ve %97 doğruluk oranları elde ettik, bu da farklı tekniklerin birleştirilmesinin kalp hastalığı tahminlerini daha kesin hale getirebileceğini göstermektedir.

2024, 54 sayfa

Anahtar Kelimeler: Kalp hastalığı tahmini, Rastgele orman, Lojistik regresyon, Destek vektör makinesi, Naive Bayes, K-en yakın komşular, Topluluk öğrenmesi

PREFACE AND ACKNOWLEDGEMENTS

I would like to thank my thesis advisor, Asst. Prof. Dr. Fuat TÜRK, for his patience, guidance, and understanding. Which supported me throughout this study. I deeply respect the trust he has placed in me and the valuable lessons he has taught me. I am grateful for her expertise and indispensable support, which enabled me to complete this study. I would also like to thank my family and friends for their interest and support. They have always understood the importance of family ties, have been a true source of inspiration for me, and have supported me in difficult times. In summary, I would like to express my deep gratitude to all those who contributed to the realization of this study.

Mohamed ALI YOUSSOUF
Çankırı-2024

CONTENTS

LIST OF ABBREVIATIONS

AI	Artificial intelligence
ANN	Artificial neural network
CHD	Chronic heart disease
DT	Decision tree
DL	Deep learning
KNN	K-Nearest neighbor
LR	Logistic regression
ML	Machine learning
NB	Naive bayes
RF	Random forest
SVM	Support vector machine

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

One of the biggest threats to human safety today is cardiovascular disease. Recently, research has brought attention to the treatment of cardiac diseases, which has been widely discussed in medical systems around the globe. According to (Sanz *et al.* 2020), heart disease ranks high among the world's leading killers. The World Health Organization estimates that 17.9 million people lost their lives to cardiovascular disease in 2022, making it the leading cause of mortality globally. More than 75% of the cardiac cases, which are the focus of this research, are from nations with low or middle incomes. Furthermore, cardiovascular disease accounts for one-third of all fatalities in individuals under the age of 70; heart attacks and strokes account for 85% of these deaths (WHO 2022).

1.1 Background

The proposed approach for prediction involves utilizing machine learning (ML), which allows systems to learn and improve from experience without explicit programming. This streamlined method offers an accurate way to assess the most indicative factors of a disease in a patient and make predictions based on those factors. Furthermore, ML enables personalized predictions for individuals by considering their specific characteristics and providing risk assessments. Overall, the use of ML provides an efficient and precise method for predicting cardiovascular disease.

In Turkey, approximately 250 thousand people are diagnosed with cardiovascular disease every year, and it is stated that approximately 150 thousand of them die. Heart disease accounts for about half of all deaths in the US and other developed countries (Medipol 2015). Cardiovascular illness has a significant impact on both human well-being and the financial aspects of nations, including healthcare expenses. The predominant cardiovascular disorders are mostly of microvascular etiology, including conditions such as coronary heart disease and cerebrovascular accidents. Following a prolonged period of

a poor lifestyle, cardiovascular illnesses may present clinically at a young age and continue to be present until old age.

The most important cardiac diseases include obesity, diabetes, family history, smoking, and high cholesterol (Nahar *et al.* 2013). Relying on risk variables to manually determine the chance of getting heart disease is difficult. To solve difficult problems, a variety of ML techniques have been developed recently (Zidan *et al.* 2020, Sagheer *et al.* 2019). Still, the most advanced ML will help us identify patterns and their useful knowledge.

ML, while extensively used in the field of medicine, is mostly employed for the purpose of forecasting cardiac disease. Researchers have shown a keen interest in using ML to identify illnesses due to its ability to reduce diagnostic time and exhibit high levels of accuracy and efficiency. The study focuses on using ML approaches to specifically detect cardiac illnesses, since they are now the primary cause of mortality. Accurate identification of cardiac disease is very beneficial in preserving lives. (Aljanabi *et al.* 2018). ML plays an important role in disease prediction (Obasi and Shafiq 2019).

Predict whether or not the patient suffers from a certain type of disease based on an effective learning technique (Sagheer *et al.* 2019, Ali *et al.* 2020).

1.2 Objective of Thesis

This study utilizes supervised learning techniques to predict the initial stage of cardiovascular disease. It employs various methodologies, such as logistic regression (LR), k-nearest neighbors (KNN), support vector machines (SVM), decision trees (DT), and random forests (RF), to determine whether the individuals under examination are affected by heart disease or are healthy. The performance of those algorithms has been examined by measuring ROC, precision, recall, F-score, and accuracy. In a notable addition to the exploration, ensemble learning methods, including boosting, bagging, and stacking, are integrated into our analysis. Ensemble learning, with its collaborative

approach of combining multiple models, presents a promising avenue for achieving higher accuracy in heart disease prediction.

A high and accurate result in predicting heart disease can be beneficial for patients, physicians, and public health institutions. Based on the benefits for patients, physicians, and public health institutions, the best algorithm and feature selection method can provide a more efficient and effective way to detect heart disease risk. High-risk heart disease patients, based on features that can be identified using the feature selection method from this study, can receive earlier treatment to reduce the severity of the illness. Physicians can optimize more advanced methods for detecting heart disease, and public health institutions can provide prevention advice for people at risk of heart disease.

1.3 Structure of Thesis

The ensuing sections of this document delve into a comprehensive literature review, elucidate the fundamentals of ML, present the proposed architecture, outline the methodology, and scrutinize experimental results while comparing various classification techniques. Ultimately, the concluding section encapsulates the key findings and implications derived from this critical exploration of heart disease prediction using a blend of ML and ensemble methods.

2. LITERATURE REVIEW

Gao *et al.* (2021) explores the use of ensemble learning techniques to improve the precision of predicting heart disease, highlighting the importance of machine learning in healthcare. The effectiveness of these ensemble techniques is evaluated against traditional machine learning algorithms using metrics such as accuracy, recall, precision, F-measure, and the ROC curve. The bagging ensemble method, which uses decision trees as the base learner, is highlighted for its superior performance, highlighting the ability of ensemble methods to manage intricate patterns in data. This study significantly enhances our understanding of machine learning applications in cardiology and sets a new standard for future research aimed at improving diagnostic procedures through artificial intelligence. They used Naive Bayes, DT, RF algorithms, and ensemble learning. As a result of the study, Naive Bayes 86.7%, RF 98.2%, DT 98.1%, SVM 85%, and KNN 98.1% accuracy rates were found, and the ensemble learning (bagging with DT) algorithm gave the highest result at 98.6%.

To diagnose cardiac disorders, (Sapra *et al.* 2021) used two datasets, the Z-Alizadesh Sani and the Cleveland heart disease dataset, that were trained by six ML algorithms: LR, DL, DT, RF, SVM, and ensemble learning (gradient boosted tree). In comparison to other algorithms, the results indicated that the gradient-boosted tree had the best accuracy, at 84%. The authors aim to provide a cost-effective, reproducible, non-invasive, rapid, and precise method for identifying heart disease using an ensemble method incorporating multiple classifiers. The Z-Alizadeh Sani coronary artery disease dataset is used for experimental validation. The ensemble approach, which combines predictions from multiple models, improves accuracy and reduces the likelihood of erroneous predictions. This method enhances the robustness of predictions, mitigating the weaknesses of individual models, leading to more reliable diagnostic outcomes. This advancement represents a significant step forward in medical diagnostics, offering a viable alternative to more invasive and expensive methods.

Atallah and Al-Mousa (2019) propose a majority vote ensemble approach for forecasting heart disease using cost-effective medical tests in local clinics. The method uses empirical

data from both healthy and diseased individuals to improve diagnoses. The methodology uses a hard voting ensemble model, which aggregates predictions from multiple machine learning models to determine the most likely outcome. This method reduces the likelihood of misdiagnosis and achieves an impressive accuracy rate of 90%. The cost-effective tests make this method accessible and practical in diverse healthcare environments, including areas with limited resources. The majority vote ensemble model represents a significant advancement in heart disease medical diagnostics, providing reliable early detection and potentially improving patient outcomes and management.

Reddy *et al.* (2019) aim to predict the risk of heart disease using feature selection techniques, focusing on the cleveland and statlog project heart datasets. The random forest method, a popular machine learning technique, is used to achieve accuracy between 90% and 95% in classification and feature selection tasks. The study identifies that including either 8 or 6 selected characteristics from the datasets is essential for optimal performance, as any further reduction in the number of features does not enhance or potentially degrade the performance of the prediction model. This finding indicates a threshold in the feature selection process beyond which the model's predictive power does not improve. This insight is crucial for developing efficient predictive models that are cost-effective and robust in their diagnostic capabilities. This work contributes to the ongoing efforts to improve cardiovascular disease diagnosis through intelligent data analysis and machine learning techniques, offering potential pathways to more accessible and affordable healthcare solutions.

Rajdhan *et al.* (2020) conducted a study on the effectiveness of machine learning algorithms in predicting heart disease using the Cleveland dataset. They used four algorithms: NB, DT, LR, and RF. The NB and LR models achieved an accuracy rate of 85.25%, while the Decision Tree model had an accuracy of 81.97%. The Random Forest algorithm outperformed the others with a 90.16% accuracy rate. The study highlights the effectiveness of ensemble methods like RF in handling complex datasets and extracting nuanced patterns. The RF algorithm's superior performance in this study suggests its potential as a robust tool for predictive analytics in healthcare settings, particularly for cardiovascular disease diagnosis.

Adhikari *et al.* (2022), used machine learning techniques to analyze data from the UCI dataset, a widely used dataset for cardiovascular disease research. They used Logistic Regression, SVM, DT, KNN, and NB to compare the effectiveness of different algorithms in predicting heart disease. DT algorithm was found to be the most accurate, demonstrating its potential in clinical applications where robust and interpretable models are necessary for effective diagnosis and patient management. This finding contributes to the ongoing discussion about the best machine learning approaches for medical diagnostics, suggesting that simpler, more interpretable models can sometimes outperform more complex ones, depending on the dataset and task requirements.

Diwakar *et al.* (2020) investigates the use of ML classification algorithms and image fusion techniques for the identification and treatment of cardiac disease. The text focuses on several ML classification algorithms, including SVM, DT, ANN, and ensemble methods. These techniques have been shown to be very useful in diagnosing cardiac disease. SVMs are highly acclaimed for their exceptional accuracy in binary classification tasks, rendering them well-suited for discerning between those who are in good health and those who suffer from heart disease. Incorporating image fusion into machine learning models is a notable breakthrough in improving diagnostic precision. Image fusion methods integrate data from multiple medical images, including echocardiograms, MRI, and CT scans, to provide a composite image that offers a more thorough assessment of the heart's state. This integration enables improved visibility of anomalies and enhances the ability of ML algorithms to make more educated classification decisions.

Patel *et al.* (2021) used longitudinal data from over 4,000 individuals over a 10-year period to predict the likelihood of developing Chronic Heart Disease (CHD) in the next decade. The dataset included variables like gender, age, diabetes status, and total cholesterol levels. Machine learning algorithms were used to train and predict the risk of developing CHD. The study aimed to assist healthcare institutions and organizations in predicting and suggesting preventive measures. The authors conducted a comparative analysis of machine learning techniques, including SVM, DT, ANN, and NB, using the Framingham dataset. They evaluated the performance of these algorithms under different scenarios, focusing on accuracy, recall, precision, and F-score. The study found that DT

method yielded the best level of accuracy. This research can help healthcare institutions and organizations forecast CHD in advance and recommend necessary preventative measures.

Srivastava *et al.* (2020) conducted a study on supervised machine learning methods to predict outcomes within a dataset. They used DT, KNN, SVM, and RF algorithms. The DT algorithm had an accuracy of 79%, which is lower than other methods due to its overfitting potential. KNN achieved a higher accuracy of 87%, which is useful for classification. SVM recorded an accuracy of 83%, which is particularly effective in high-dimensional spaces and cases where the number of dimensions exceeds the number of samples. The Random Forest algorithm, which builds multiple decision trees and merges them together, recorded an accuracy of 84%. The study highlights the importance of choosing the right algorithm based on the specific characteristics of the data and the prediction task. The differences in performance could also guide further investigations into optimizing or combining each algorithm to enhance predictive accuracy in future applications.

Almustafa (2020) found that several classification algorithms demonstrated a high level of accuracy in predicting cardiac disease. The study also noted that to predict heart disease effectively, it is feasible to use a reduced set of features instead of analyzing all the attributes in the dataset. These algorithms are robust and can handle complex health datasets, making them valuable tools in predictive health analytics. The study also suggested that using a reduced set of features instead of analyzing all available attributes can improve predictive accuracy without overfitting. Healthcare providers can focus on collecting relevant data and optimizing the diagnostic process, aligning with machine learning and data science trends.

Shouman *et al.* (2012) used 303 patient records and 14 patient attributes to predict heart disease in the Cleveland dataset from the UCI ML Repository. In their analysis, they used WEKA software and one of the supervised learning algorithms, one of the ML types, Naive Bayes, DT, KNN algorithm, and RF algorithms, and they found the highest accuracy result in the KNN algorithm with k=7 value.

Ali *et al.* (2021), compared their performances with LR, adaboosM1, multi-layer perceptron, KNN algorithm, DT, and RF algorithm, which are ML types, to evaluate the risk of heart disease from the data set. As a result, they found that the KNN algorithm, DT, and RF algorithm gave the most successful results.

Katarya and Meena (2021) predicted heart disease using a dataset from the UCI Repository. They used 14 out of 76 features in the dataset. They performed comparative analysis using ML algorithms such as LR, Naive Bayes, SVMs, KNN algorithm, DT, RF algorithm, ANNs, deep neural network, and multilayer perceptron algorithms. They used root mean square error, mean absolute error, precision, and sensitivity techniques to evaluate the accuracy of the results. As a result of their analysis, LR gave 93.40%, SVMs 92.30%, and ANNs 92.30%, but the RF algorithm gave the best results with an accuracy of 95.60%.

Bharti *et al.* (2021) received from UCI for heart disease prediction in Cleveland, 14 attributes out of 76 attributes in this dataset and they used ML algorithms. They used ML algorithms in their study and LR 83.3%, KNN algorithm 84.8%, SVMs 83.2%, RF algorithm 80.3%, DT 82.3%, and DL algorithm gave 94.2% accuracy rates. The most successful result was obtained by the DL technique (94.2%).

Dwivedi (2018) used ML algorithms to predict heart disease. He used the StatLog Heart Disease dataset from UCI and receiver operative characteristic (ROC) curve and calibration graphs to evaluate the performance of the algorithms. Among the algorithms he used in his study, the ANN gave 84%, SVMs 82%, LR 85%, KNN algorithm 80%, tree classifier algorithm 77%, and Naive Bayes 83%. LR gave the highest accuracy rate among the algorithms with 85%.

Mallela *et al.* (2021) used the Cleveland dataset, which includes 303 patient records obtained from the UCI Repository and 14 attributes for each record. They used Python programming language and ML algorithms such as KNN algorithm, DT, and RF algorithms and found the highest accuracy rate as 85.066% with the k-nearest algorithm k=12.

Jindal *et al.* (2021) used a heart disease dataset obtained from UCI for heart disease prediction and KNN algorithm, LR, and RF algorithms as algorithms in their study titled heart disease prediction using ML Algorithms. As a result of their study, they observed that the KNN algorithm and LR gave the best result with an accuracy rate of 88.5%. Table 2.1 shows Several studies summary on heart disease based machine learning.

Table 2.1 Several studies summary on heart disease prediction

AUTHORS	ALGORITHMS	ACCURACY
Gao *et al.* (2021)	NB, DT, RF algorithms, and ensemble learning	Naive Bayes 86.7%, RF 98.2%, DT 98.1%, SVM 85, KNN 98.1% accuracy rates were found, and the ensemble learning (bagging with DT) algorithm gave the highest result at 98.6%
Sapra *et al.* (2021)	LR, DL, DT, RF, SVM, and ensemble learning (gradient-boosted tree)	The finding showed that gradient boosted tree achieved the best accuracy of 84% compared to other algorithms
Atallah and Al-Mousa (2019)	Stochastic gradient descent, KNN, RF, LR, and voting ensemble learning	The voting ensemble learning model has achieved the best accuracy of 90%.
Reddy *et al.* (2019)	RF, SVM, NB, NN, and KNN	The findings indicate that RF achieved the highest level of performance.
Rajdhan *et al.* (2020)	NB, DT, LR, RF	Naive Bayes: 85.25%, DT: 81.97%, LR: 85.25%, RF: 90.16%
Adhikari *et al.* (2022)	LR, SVM, DT, KNN, GNB	DT: Highest Accuracy
Diwakar *et al.* (2020)	ANNs	Good results with ANNs
Patel *et al.* (2021)	SVM, LR, DT, KNN, RF	DT: Highest Accuracy
Srivastava *et al.* (2020)	DT, KNN, SVM, RF	79%, 87%, 83%, 84% accuracy respectively
Almustafa (2020)	Not specified	Different classification algorithms gave high accuracy.
Shouman *et al.* (2012)	NB, DT, K-NN, RF	K-NN (k=7): Highest Accuracy
Ali *et al.* (2021)	LR, AdaboostM1, MLP, K-NN, DT, RF	K-NN, DT, and RF gave the most successful results.
Katarya and Meena (2020)	LR, NB, SVM, K-NN, DT, RF, ANNs, Deep Neural Network, Multilayer Perceptron	RF: 95.60% accuracy
Bharti *et al.* (2021)	LR, K-NN, SVM, RF, DT, DL	DL: 94.2% accuracy
Dwivedi (2018)	ANN, SVM, LR, K-NN, Tree Classifier, Naive Bayes	LR: 85% accuracy
Jindal *et al.* (2021)	K-NN, LR, RF	K-NN, LR: 88.5% accuracy

3. GENERAL PARTS

3.1 Machine Learning

As shown in Figure 3.1, ML is a subset of artificial intelligence (AI) that enables computers to learn from existing data without explicit programming. Its essence lies in improving performance on a particular task by learning from previous experiences and data. ML algorithms can learn and adapt automatically, making them adept at dealing with complex tasks and evolving datasets. ML is a paradigm change in computing that enables systems to learn and adapt without explicit programming (Pradhan and Singh 2020).

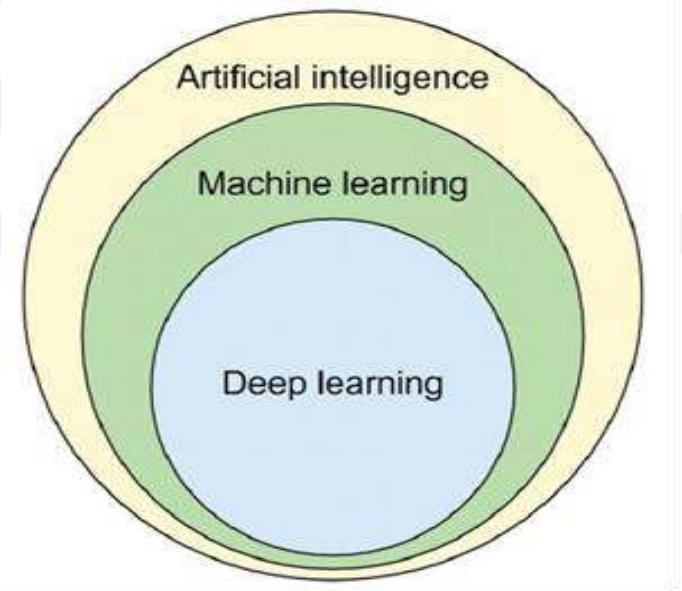

Figure 3.1 Artificial intelligence (Saint-Cirgue 2019)

There are three methods of ML: (Figure 3.2)

- Supervised learning
- Unsupervised learning
- Reinforcement learning

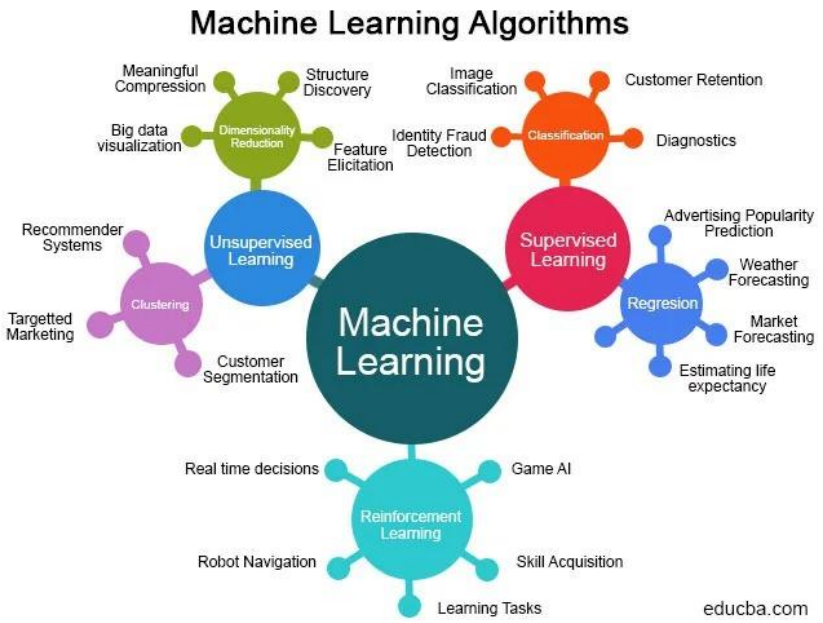

Figure 3.2 Machine learning methods (Saint-Cirgue 2019)

3.1.1 Supervised learning

Supervised learning is regarded as one of the pivotal models in machine learning. Its critical characteristic lies in leveraging correlated data between inputs and outputs. This categorization of data aids in classifying features effectively. Once the algorithm establishes the connection between inputs and outputs, it can forecast outcomes. Moreover, as additional data becomes available, the precision of the machine learning algorithm's outputs improves significantly (Saint-Cirgue 2019).

Supervised learning stands as one of the cornerstone models in ML, renowned for its vital role in understanding and predicting correlations between input and output data. This method is critical for jobs with labeled data because it allows the algorithm to learn from known cases and generalize to new, previously unseen occurrences. Supervised learning is a fundamental concept in ML that uses labeled data to train algorithms and make predictions on previously unseen data. It is based on a dataset where both input and output data are labeled, allowing the algorithm to learn underlying patterns and relationships. The explicit classification of input features and output labels is crucial for recognizing trends and translating input data to appropriate output categories. Once trained on labeled

data, the relationship between input and output, enabling it to make predictions on previously unknown data. The goal is to generalize well to previously unknown examples, making accurate predictions outside the training set. Supervised learning is widely used in applications such as image recognition, speech recognition, natural language processing, and predictive modeling.

There are two types of supervised learning: classification, which predicts categorical class labels, and regression, which predicts a continuous numerical output. In summary, supervised learning is a fundamental idea in ML that uses labeled data to train algorithms and make predictions on previously unseen data.

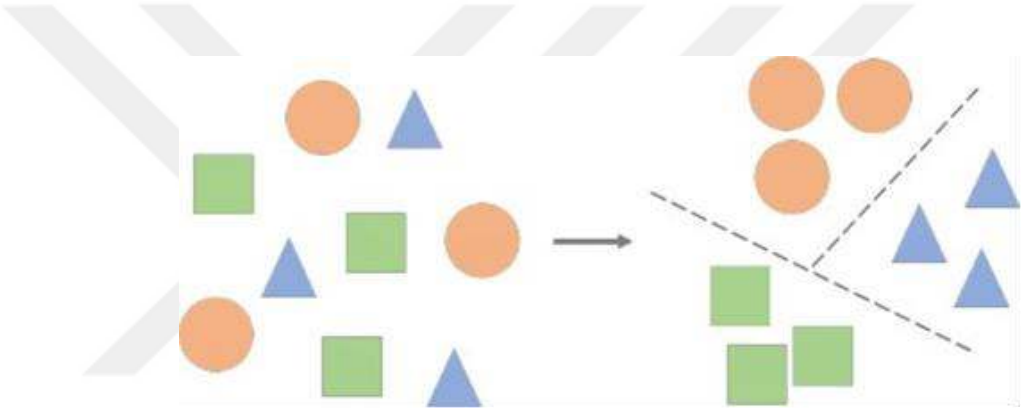

Figure 3.3 Supervised learning (Saint-Cirgue 2019)

3.1.2 Unsupervised learning

In unsupervised learning algorithms, data is not pre-classified into distinct categories or labels. Instead, the model discovers underlying patterns by analyzing the data itself. It identifies groups within the data based on attributes such as density, structure, and segmentation, among other characteristics (Saint-Cirgue 2019).

Unsupervised learning is a strong ML paradigm in which the algorithm extracts patterns and structures from unlabeled data without explicit supervision or predetermined categories. In contrast to supervised learning, unsupervised learning investigates the data's intrinsic structures, detecting links, similarities, and groupings.

Unsupervised learning is a method that uses unlabeled data without explicit labels or predetermined output categories to identify underlying patterns and structures in the data. It focuses on detecting correlations, dependencies, or similarities between data points without being told what to look for. Unsupervised learning characterizes data based on its inherent properties, such as densities, structures, segments, or other patterns. Common strategies include clustering, dimensionality reduction, and density estimation. Clustering involves clustering related data points together, while dimensionality reduction techniques aim to capture the most important features of the data while minimizing complexity. Generative models, such as generative adversarial networks and variational autoencoders, are used to learn the underlying distribution of data. Unsupervised learning has applications in anomaly detection, recommendation systems, and exploratory data analysis, especially when the data's underlying structure is unknown and the goal is to discover intrinsic patterns.

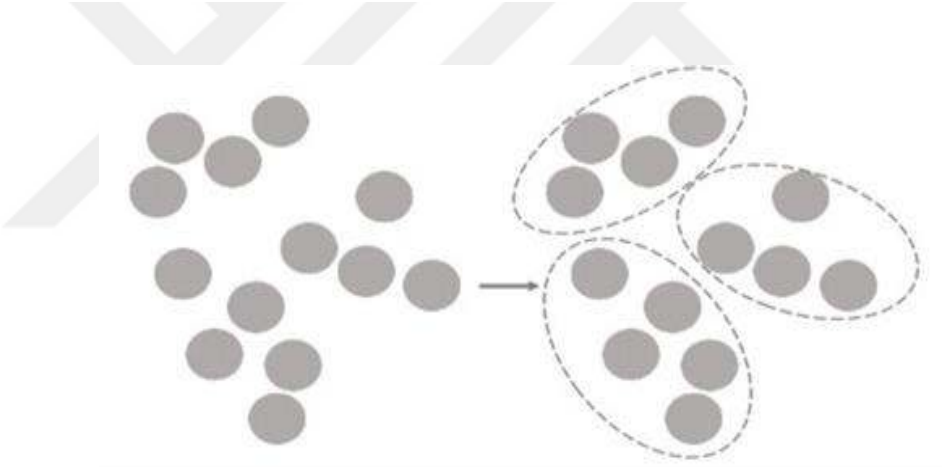

Figure 3.4 Unsupervised learning (Saint-Cirgue 2019)

3.1.3 Reinforcement learning

Reinforcement learning is a dynamic and adaptable branch of AI that allows machines to interact with their surroundings and learn optimal behaviors to achieve certain objectives. Unlike supervised learning, which uses explicit labels to guide the learning process, reinforcement learning shapes and refines an agent's behavior over time via a system of rewards and penalties. Reinforcement learning is a paradigm that focuses on teaching agents to make sequential judgments in dynamic situations. It involves agents interacting

with a changing environment, learning by acting, and watching the results. The goal is to teach machines or software agents to learn and adopt actions that lead to the achievement of certain goals, defined by a reward signal. Rewards-based learning is based on a system of rewards and punishments, where positive actions are rewarded, and undesirable behavior has repercussions (Saint-Cirgue 2019).

Sequential decision-making involves making a series of decisions over time, with each decision influencing future states and behaviors. Balancing exploration and exploitation is a fundamental difficulty in reinforcement learning, as agents must experiment to discover optimal methods while leveraging their current knowledge for short-term rewards. Reinforcement learning problems are often formulated as Markov Decision Processes, where the current state contains all relevant information and the agent's actions influence the transition between states.

Reinforcement learning has applications in various fields, including robots, gaming, autonomous systems, recommendation systems, and natural language processing. Notable applications include training game-playing agents and autonomous cars. In conclusion, reinforcement learning is an effective paradigm for teaching agents to make sequential judgments in dynamic situations, enabling them to learn and adapt independently (Figure 3.5).

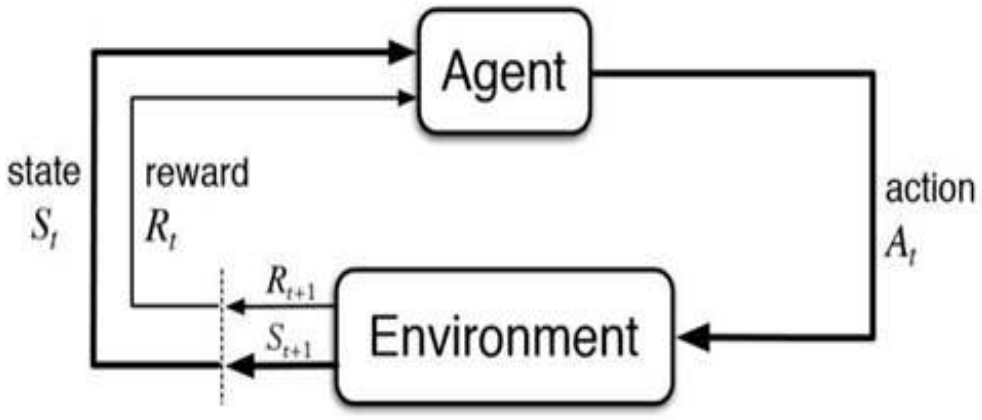

Figure 3.5 Reinforcement learning (Saint-Cirgue 2019)

3.2 Machine Learning Techniques

3.2.1 Randon forest structure

In Figure 3.6, the architectural structure of RF is explained simply.

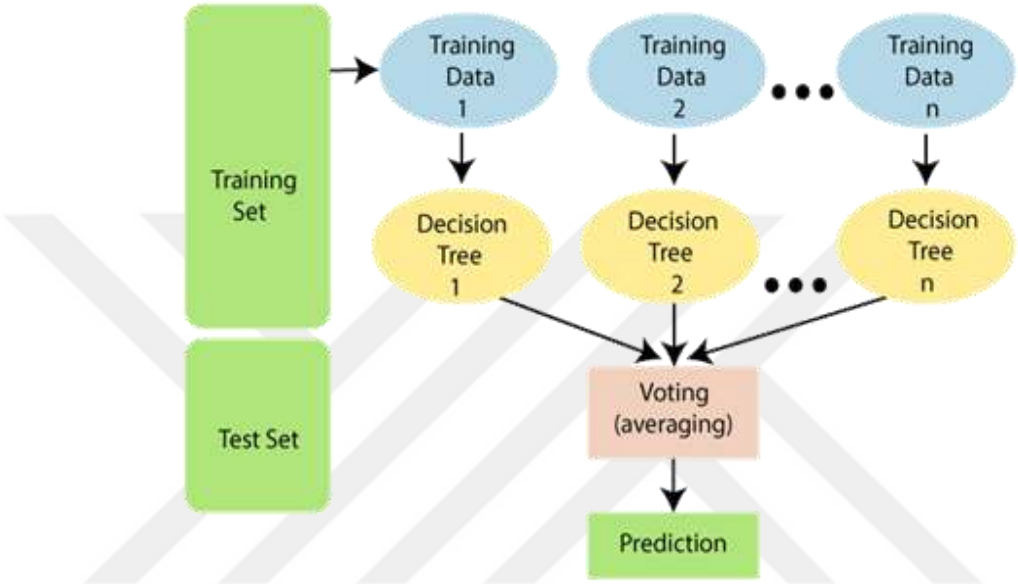

Figure 3.6 Random forest architecture (JavaTpoint 2021)

RF is a popular ML method in the field of supervised learning. It is a versatile tool that can be used for both classification and regression issues in ML and operates on ensemble learning principles. The core principle of ensemble learning is the deliberate combining of several classifiers to address complex problems and improve the model's overall performance (Çil and Güneş 2005).

In classification, RF excels in assigning input data points to predefined groups or classes. Simultaneously, it performs well in regression tasks, predicting continuous values based on input data. The strength of RF lies in its ability to build a multitude of DTs during the training phase. Each tree is constructed independently, utilizing a random subset of the available features and a random subset of the training data. This randomness injects diversity into the individual trees, mitigating the risk of overfitting and enhancing the model's robustness.

The ensemble nature of RF further contributes to its efficacy. During the prediction phase, the algorithm aggregates the predictions of each individual tree are combined using a voting mechanism for classification or an average strategy for regression. This ensemble strategy leverages the collective wisdom of multiple DTs, resulting in a more accurate and stable prediction.

RF is known for its resilience to noise in the data and its ability to handle large datasets with numerous features. Additionally, it provides a valuable feature importance ranking, aiding in the identification of influential predictors. RF is well-known for its tolerance to data noise and capacity to handle huge datasets with many features. It also provides a useful feature relevance ranking, which helps identify key predictors.

3.2.2 K-nearest neighbors

KNN is one of the most straightforward ML algorithms in the supervised learning arena. KNN is a versatile tool that uses supervised learning techniques to solve regression and classification issues (Çil and Güneş 2005).

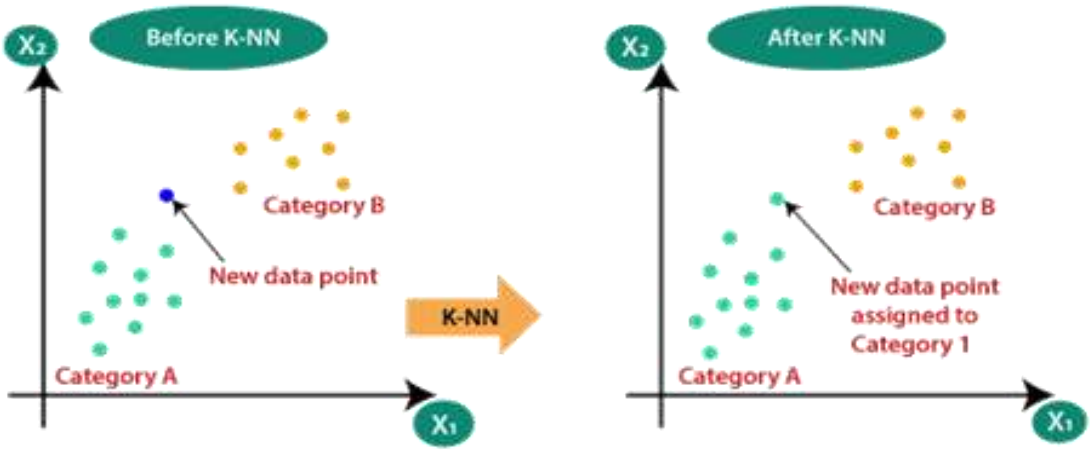

Figure 3.7 K-nearest neighbors (JavaTpoint 2021)

In terms of classification, KNN excels in labeling input data points based on their similarity to neighboring data points. It works on the notion that similar occurrences in

the feature space are likely to belong to the same class. During the training phase, KNN maintains the complete dataset, and when asked to predict the class of a new data point, it finds the KNNs in the feature space. The majority class among these neighbors is then allocated to the new data point. For regression tasks, KNN predicts a continuous value for a new data point by taking the average or weighted average of its KNNs' values. KNN's simplicity stems from the lack of an explicit training phase. The method saves the complete training dataset and conducts computations during prediction. This feature makes KNN extremely simple to implement and adapt to different types of data. However, it is important to remember that the efficacy of KNN is heavily influenced by the distance metric used to measure similarity, as well as the K value. While KNN is sensitive to local fluctuations in data, it may be less effective when dealing with irrelevant or noisy features.

3.2.2 Logistic regression

LR is commonly used for prediction and classification problems. LR is a widely used method in machine learning, particularly for problems involving prediction and classification. Contrary to its name, this is a classification method rather than a regression technique. LR is particularly useful for binary classification jobs in which the goal is to predict the likelihood of an instance belonging to a specific class.

The key properties and components of LR are:

- The LR algorithm uses the sigmoid function (logistic function) to turn the output into a range of 0 to 1. This range shows the chance that the instance belongs to the positive class.
- LR is intended specifically for binary classification issues with two possible outcomes or classes. However, it can be extended to handle multiclass classification by employing approaches such as one-vs-all or one-vs-one.
- The result of LR can be regarded as the likelihood of an instance belonging to the positive class. The final classification choice might be made based on a threshold (often 0.5).

- LR models the log odds of a probability, allowing for a linear relationship between input features and log odds. This makes it ideal for capturing complicated relationships in data.
- Regularization approaches, including L1 or L2 regularization, are methods that may be used in logistic regression (LR) to mitigate overfitting and enhance the ability to generalize to new data.
- LR defines a decision boundary in the feature space, separating instances of different classes. This boundary is determined based on the learned weights associated with each feature.

LR is favored for its simplicity, interpretability, and efficiency. It performs well when the relationship between features and the log odds of the response variable is approximately linear. Moreover, it serves as a baseline model in many machine-learning workflows and is often used as a benchmark for more complex algorithms.

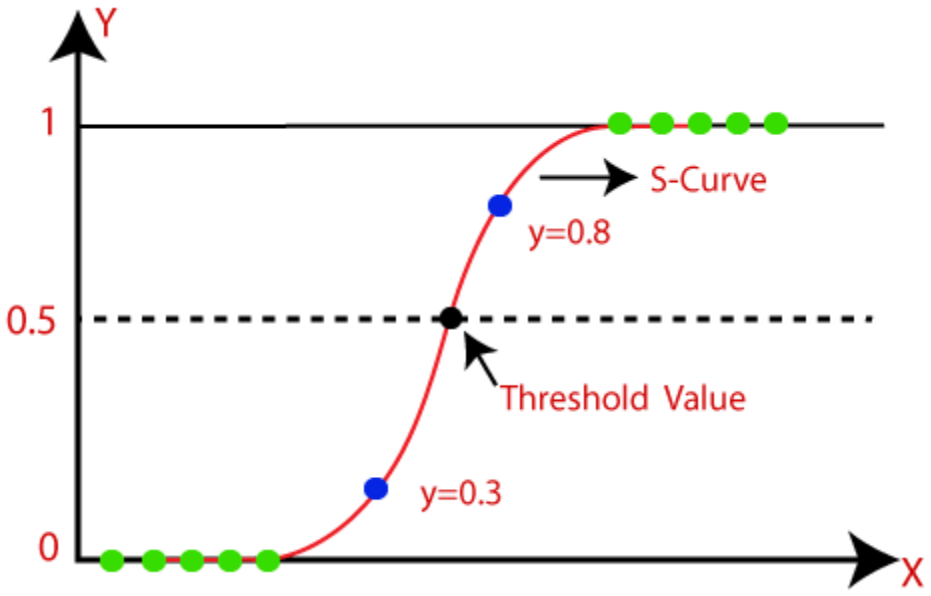

Figure 3.8 Logistic regression (Çil and Güneş 2005)

3.2.3 Decision tree

DT algorithm utilizes this labeled data to create a tree-like structure where data is split based on specific parameters, leading to a series of decision nodes, branches, and leaf

nodes. This hierarchical structure captures the decision-making process and aids in making predictions for new, unseen data (Çil and Güneş 2005). The essential components of a DT include:

- Decision nodes are points in the tree where the data is split based on specific features or attributes. These nodes represent decisions that lead to subsequent branches.
- Branches emanate from decision nodes and represent different outcomes or paths based on the conditions specified at the decision nodes. Each branch corresponds to a particular value or range of values of a feature.
- Leaf nodes are the terminal nodes of the DT. They represent the final decision or outcome. In the context of classification problems, each leaf node typically corresponds to a specific class label. For regression problems, leaf nodes might represent predicted numerical values.

To construct a DT, the data is recursively partitioned at each decision node, and the ideal feature and its accompanying condition are chosen to maximize the homogeneity of the resulting subsets. This guarantees that, as the tree grows, it catches patterns and relationships in the data, allowing it to make accurate predictions for future instances. DTs are useful because of their interpretability and ease of depiction. They provide a visible picture of the decision-making process, allowing people to follow the logical sequence of events. However, care must be taken to avoid overfitting, which occurs when the tree captures noise in the training data but does not transfer well to new data. Figure 3.9 shows the flowchart of DT (JavaTpoint 2021).

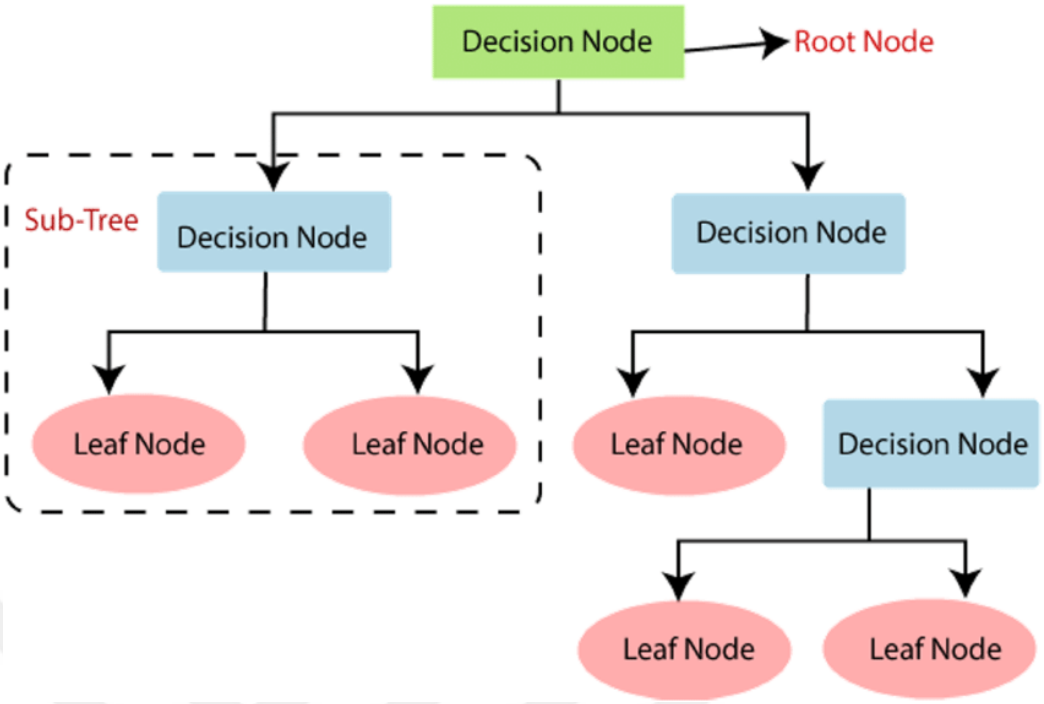

Figure 3.9 Decision tree (Deshpande *et al.* 2021)

3.2.5 Support vector machine

SVM is a popular supervised learning technique that excels in both classification and regression problems. The main premise of SVM is to determine the optimal hyperplane that best separates data points from distinct classes in the feature space (Ghosh *et al.* 2019). Figure 3.10 depicts the architectural structure of SVMs. In classification, SVM aims to create a decision border that optimizes the margin between classes, where the margin is the distance between the hyperplane and the nearest data points of each class. This margin maximization improves the algorithm's robustness and generalization to new data. The main traits and aspects of SVM are:

- The kernel approach enables SVM to effectively manage nonlinear decision boundaries.
- This entails transforming the input data into a higher-dimensional space where linear separation is more practical.
- Support vectors are the data points nearest to the decision border. They are vital in identifying the best hyperplane and, as a result, defining the decision bounds.

- The C parameter in SVM allows you to fine-tune the trade-off between having a smooth decision border and correctly classifying training points. A larger C value leads to a more accurate classification of training points, possibly at the expense of a more complex decision boundary.
- In the case of non-linear kernels, the gamma parameter influences the extent to which a single training example influences the model. A smaller gamma results in a more global influence, while a larger gamma focuses more on local details.

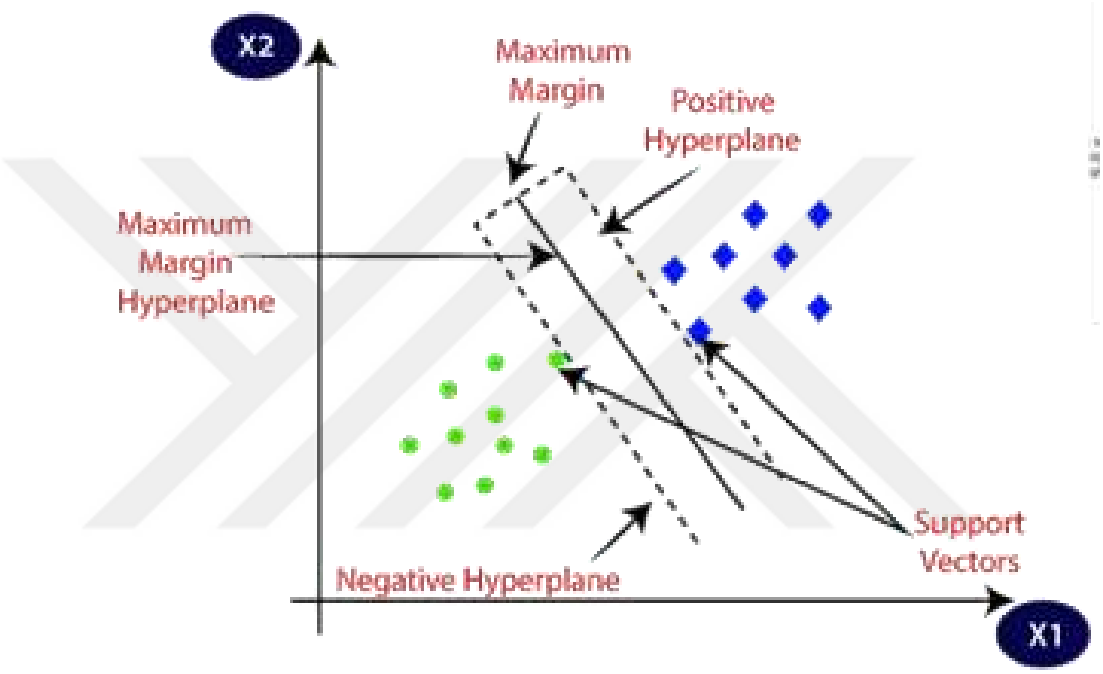

Figure 3.10 Support vector machine (JavaTpoint 2021)

3.2.6 Naive Bayes

Naive Bayes (NB) classifier is a probabilistic supervised learning method used in classification tasks, assuming feature independence. It calculates the probability of a given data point belonging to each class based on feature values and uses Bayes' theorem to determine the conditional probability of each class given the observed features. Naive Bayes classifiers can be categorized into Gaussian Naive Bayes for continuous features and Multinomial Naive Bayes for discrete features. Parameter considerations include Laplace Smoothing to handle unseen feature-label combinations in training data and hyperparameter tuning to impact the model's performance. While NB has few

hyperparameters, careful selection can significantly impact the model's performance (Di and Sordoni 2012).

3.3 Ensemble Learning Models

Ensemble learning, which combines the predictions of multiple base models, has been widely used in machine learning applications. The three main ensemble methods, bagging, boosting, and stacking, have been extensively studied and applied in various fields. These methods have been shown to improve predictive performance, particularly in high-dimensional and multi-class algorithms (Shobana & Nandhini, 2022).

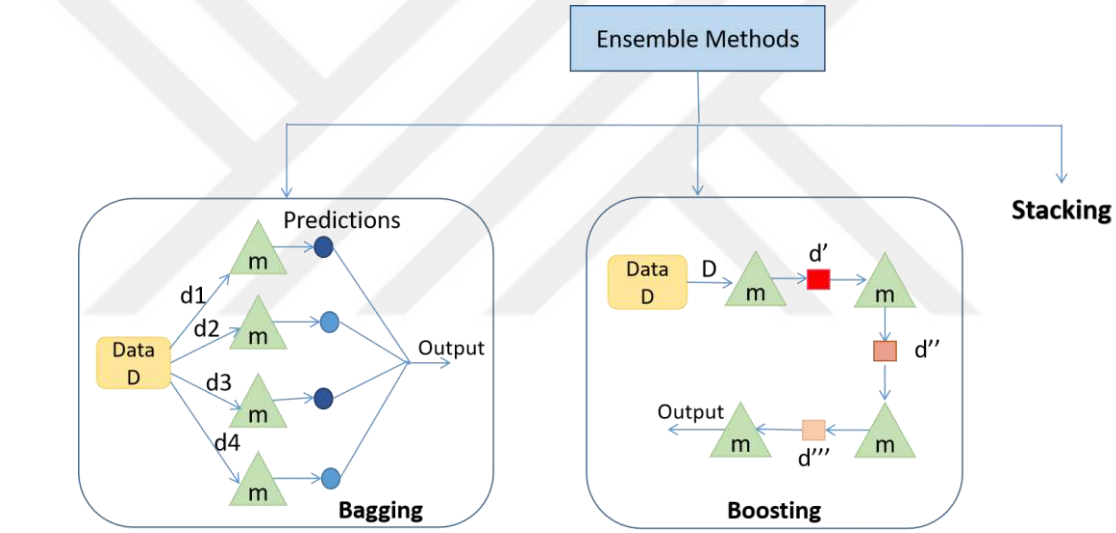

Figure 3.11 Ensemble methods types (Khandelwal 2021)

3.3.1 Boosting

Boosting is a powerful ML approach used to improve the performance of weak learners, which are models that perform only slightly better than random guessing (Mendonça *et al.* 2022). By combining a large number of these weak learners, boosting produces a robust model with considerably higher accuracy than any single weak learner. There are several boosting algorithms, all of which use a similar method. Notable examples are AdaBoost, Gradient Boosting, and XGBoost, with our work focusing on AdaBoost. AdaBoost, one of the early boosting algorithms, uses an iterative method to include weak

learners in a model. Each weak learner is trained using a weighted sample of data, with the weights changed based on the performance of previous weak learners (Alon *et al.* 2021). This strategy addresses individual models' weaknesses by focusing on areas where they struggle, hence improving overall predictive capacity. AdaBoost offers larger weights to misclassified data points, encouraging weak learners to focus on these instances and correct errors. This adaptive learning approach allows the model to continuously enhance its performance. One of AdaBoost's features is its ability to accommodate a wide range of base models, making it adaptable to varied datasets and problem domains (Møller *et al.* 2023). It is very good at mitigating bias and overfitting. AdaBoost was chosen for our investigation since it is effective and versatile. The algorithm's performance in ensemble learning scenarios, when multiple weak learners are merged, has consistently proved its ability to improve predicted accuracy. I wanted to improve the ML model's classification performance by thoroughly studying boosting principles and focusing on AdaBoost in our research. AdaBoost's iterative structure ensures that the model evolves dynamically, making it suitable for a wide range of datasets and leading to its extensive use in a variety of applications.

3.3.2 Bagging

Bagging, short for Bootstrap Aggregating, is an ensemble learning technique designed to enhance the accuracy and stability of predictive models. The core principle of bagging involves resampling the training data multiple times to create diverse subsets (Masoodi *et al.* 2022). A base model is then trained on each of these subsets, and their predictions are aggregated to formulate the final prediction. Resampling brings unpredictability and variation into the training process, reducing overfitting and enhancing model generalization performance. Bagging aids in the creation of a robust ensemble model that performs well across a variety of data patterns by training many models separately on different subsets. The RF method is one well-known bagging implementation. In RF, DTs are trained on multiple subsets of data, and their predictions are pooled using a voting system. This enhances the model's prediction ability by capitalizing on the diversity generated by bagging (Wang *et al.* 2024).

3.3.3 Stacking

Stacking is a powerful approach because it allows base models to compensate for each other's weaknesses, exploiting their respective strengths. This can lead to better overall predictive performance. Stacking, another complex ensemble learning strategy, works by allowing base models to adjust for each other's flaws while maximizing their particular strengths (Khandelwal, 2021). This collaborative technique tries to provide better predictive performance than solo models. Stacking involves training numerous diverse base models on the same dataset. Stacking, rather than depending on a simple average or voting process, introduces a meta-model that learns to successfully integrate the underlying models' predictions. The meta-model considers the strengths and shortcomings of each base model, utilizing their complementary nature to improve overall forecast accuracy. The adaptability of stacking stems from its capacity to adapt to the individual qualities of the data. By integrating a variety of base models, stacking excels at capturing intricate relationships within the information, making it an excellent solution for complex and nuanced prediction tasks (Figure 3.11).

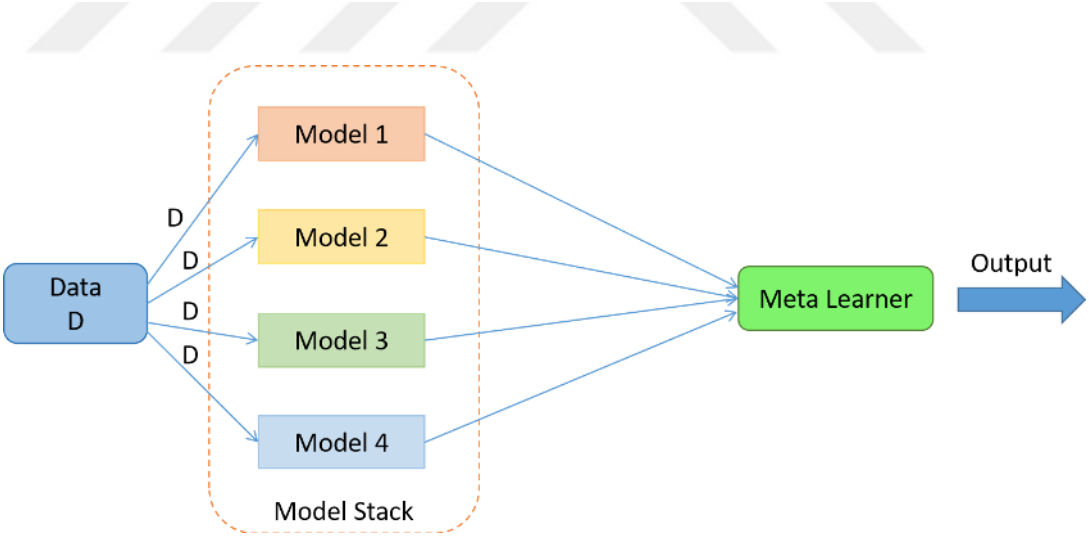

Figure 3.11 Ensemble stacking (Khandelwal 2021)

4. MATERIALS AND METHODS

4.1 Data Description

The heart disease dataset was collected from January 2019 to May 2019 at Zin Hospital in Erbil, Iraq. This dataset has the following attributes: age, sex, heart rate, systolic blood pressure, diastolic blood pressure, blood glucose level, CK-MB, and troponin negative or positive outputs. The gender column of data is normalized. Male=1, female=0. As for the output, positive=1 and negative=0 (Rashid *et al.* 2022).

Table 4.1 Dataset description

ATTRIBUTE	DESCRIPTION
Age	Age of the individual
Sex	Gender of the individual (Male=1, Female=0)
Heart Rate	Heart rate of the individual
Systolic Blood Pressure	Systolic blood pressure of the individual
Diastolic Blood Pressure	Diastolic blood pressure of the individual
Blood Glucose Level	Blood glucose level of the individual
CK-MB	CK-MB (Creatine Kinase-MB) level
Troponin	Troponin level (Negative=0, Positive=1)

4.2 Preprocessing

After evaluating the quality of the dataset, the next step involves preparing the data for pre-processing. This step, often referred to as data pretreatment and transformation, is essential as it substantially impacts the effectiveness of the deployed predictive models. At this stage, the data undergoes systematic processing to ensure cleanliness, consistency, and compatibility with the ML algorithms. Normalization and missing value imputation techniques, crucial for improving dataset quality, involve adjusting data scales to specific ranges, such as 0 to 1 or -1 to 1, to prevent the magnitude of the variables from influencing the machine learning algorithms in a biased manner.

4.3 Performance Evaluation Metrics

4.3.1 Confusion matrix

A confusion matrix, often referred to as an error matrix, is a structured table format created to accurately represent the effectiveness of a supervised learning system. The matrix's rows represent the occurrences of real classes, while the columns represent the instances of anticipated classes. The following table presents a confusion matrix for binary classification, which serves as the basis for generating other associated terminology and performance indicators.

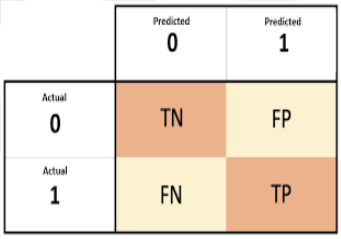

Figure 4.1 Confusion matrix (Çil and Güneş 2005)

- TN (True Negative): Values classified as Negative (0) in true values Demonstrate correct prediction of the outcome by negative class prediction.
- TP (True Positive): Values classified as Positive (1) in actual values Positive class prediction indicates a correct prediction.
- FP (False Positive): Values classified as Negative (0) in actual values Positive class indicates that the prediction was incorrectly predicted.
- FN (False Negative): Values classified as Positive (1) in actual values A negative class indicates that the prediction was incorrectly predicted.

4.3.2 Accuracy

It shows the accuracy of the values we predict in the data set. It is defined as the true positive values plus true negative values divided by true positive plus true negative plus false positive plus false negative. Equation (4.1) shows the calculations: of accuracy

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (4.1)$$

4.3.3 Precision

Indicates how many of the positive samples classified in the prediction field were correctly classified. Equation (4.2) shows the calculations: of precision:

$$\text{Precision} = \frac{TP}{TP+FP} \quad (4.2)$$

4.3.4 Recall

Sensitivity shows the ratio of correctly predicted positive values to the entire positive class (true positive and false negative) in actual values. A low sensitivity ratio indicates more false positives and less true positives, while a high sensitivity ratio indicates more true positives and less false positives. Equation (4.3) shows the calculations: of sensitivity or recall.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (4.3)$$

4.3.5 F1-score

F1 Score Returns the harmonic average of precision and Recall (sensitivity). Equation (4.4) shows the calculations: of the F1-score.

$$\text{F1-score} = 2 \cdot \frac{precision \cdot Recall}{precision+Recall} \quad (4.4)$$

4.4 Data Analysis and Interpretation

In the context of your study, data analysis and interpretation are critical for evaluating the performance of various supervised learning algorithms on the labeled dataset. The dataset is provided in CSV (comma-separated values) format, to use a set of classification methods for both training and testing. The classification techniques used are LR, RF, DT, KNN, SVM, Naïve Bayes (NB), and Ensemble Learning methods like bagging, boosting, and stacking. The Python programming language is used to create and compare the outcomes of various algorithms. Key Steps in data analysis and interpretation:

- The initial step is to load the labeled dataset in CSV format. This includes investigating the data's structure, dealing with missing values, and doing any necessary preparatory processes such as normalization or encoding categorical variables.
- The dataset is usually separated into two parts: a training dataset for training the models and a testing dataset for evaluating their performance. This ensures an unbiased evaluation of the models using new, previously unseen data.
- Each of the mentioned classification algorithms is written in Python. Scikit-learn is a popular library due to its ease of use and thorough implementations of many machine-learning techniques.
- Each classification algorithm is trained using data from the training set. To create predictions, the models understand the data's fundamental patterns and correlations.
- The testing dataset is used to assess the effectiveness of each model. Metrics including accuracy, precision, recall, F1-score, and confusion matrix can be used to determine how effectively each method generalizes to new data.
- The results of each algorithm are compared to determine strengths and shortcomings. Considerations include accuracy, computing efficiency, and model interpretability.
- The collective performance of ensemble learning methods (bagging, boosting, and stacking) is assessed. The combination of basic models and their effect on total predicted accuracy are evaluated.

5. RESULTS AND DISCUSSION

5.1 Results

In this section presents an evaluation of the performance of all implemented algorithms, encompassing metrics such as accuracy, precision, recall, confusion matrices, and AUC-ROC for each model.

5.1.1 Logistic regression

In the test data set where the LR algorithm was applied, predictions were made for 264 patients. Correct predictions were made for 72 people without heart disease and 133 people with heart disease. Incorrect predictions were made for 30 people without heart disease and 29 people with heart disease. LR algorithm applied model; the accuracy of the test data set was 78% (Figure 5.1).

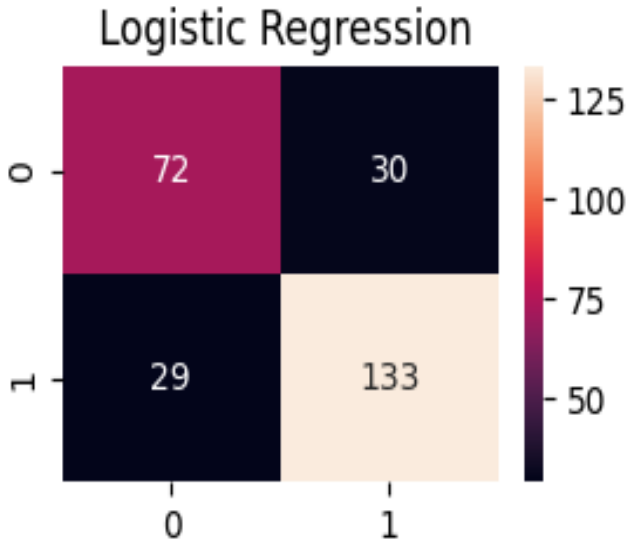

Figure 5.1 Confusion matrix of LR

Figure 5.2 shows the comparative performance of various models used in predicting cardiac disease. The figure presents essential assessment data, including accuracy,

precision, recall, and F1-score, for each model assessed. Performance is quantified using a percentage metric and offers a comprehensive evaluation of each model's accuracy in predicting cardiac disease. The values for each statistic are as follows: the model's accuracy is 78%, precision is 82%, recall is 82%, and the F1-score is 82%.

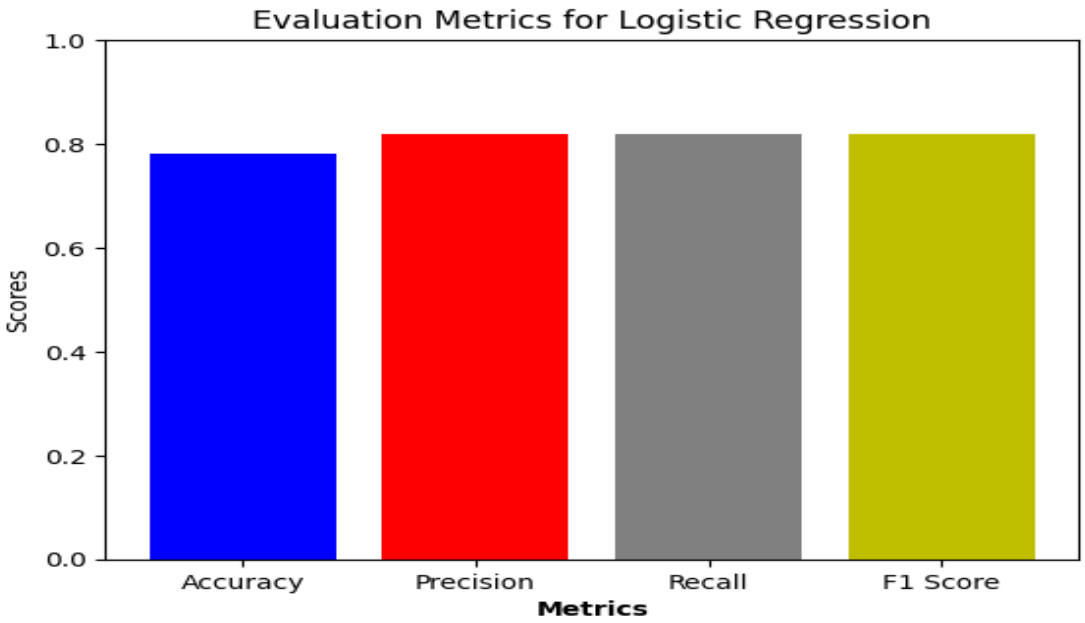

Figure 5.2 Evaluation metrics for LR

The ROC curve assesses the predictive capability of the LR model for heart disease by plotting the relationship between the false positive rate and the true positive rate at various threshold levels. The model's AUC is 0.86, which indicates high efficacy in predicting heart disease (Figure 5.3).

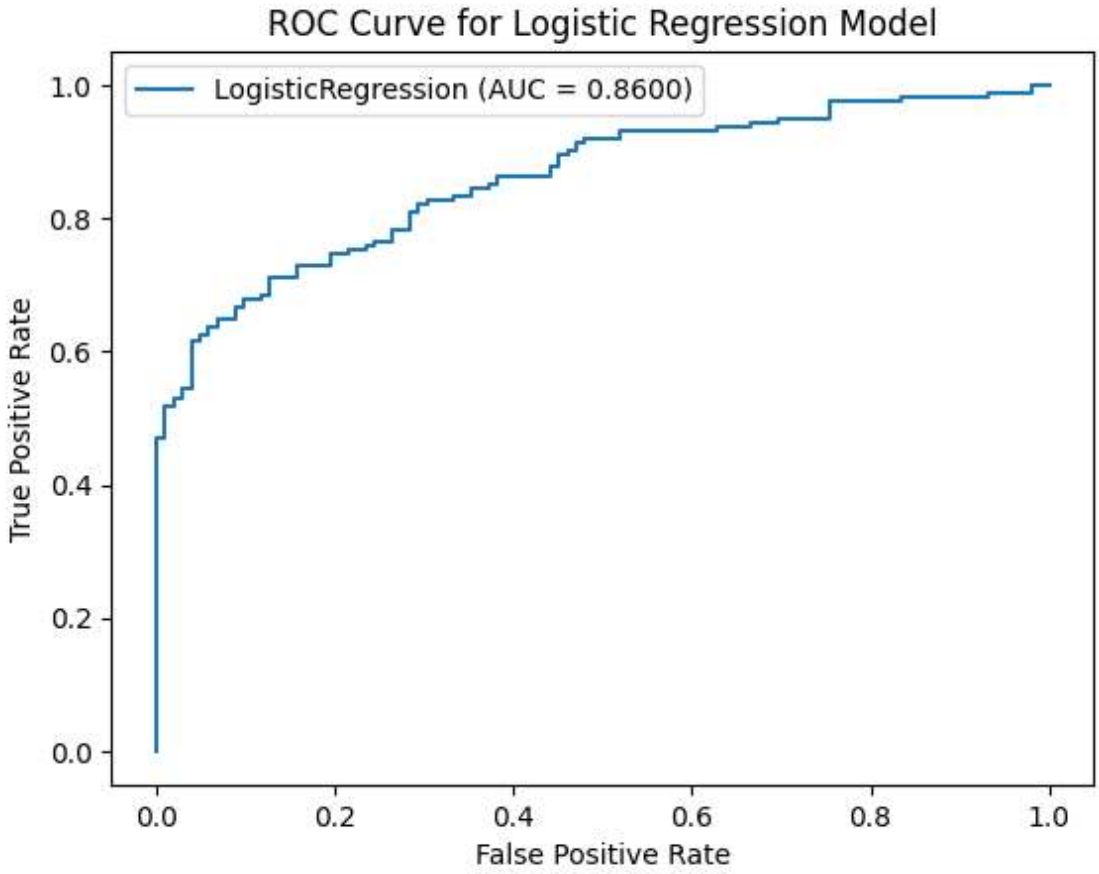

Figure 5.3 ROC curve for LR

5.1.2 Random forest

In the test dataset where the RF algorithm was applied, predictions were made for 264 patients. Correct predictions were made for 100 people without heart disease and 159 people with heart disease. Incorrect predictions were made for 2 people without heart disease and 3 people with heart disease. RF algorithm applied model; the accuracy of the test data set was 98% (Figure 5.4).

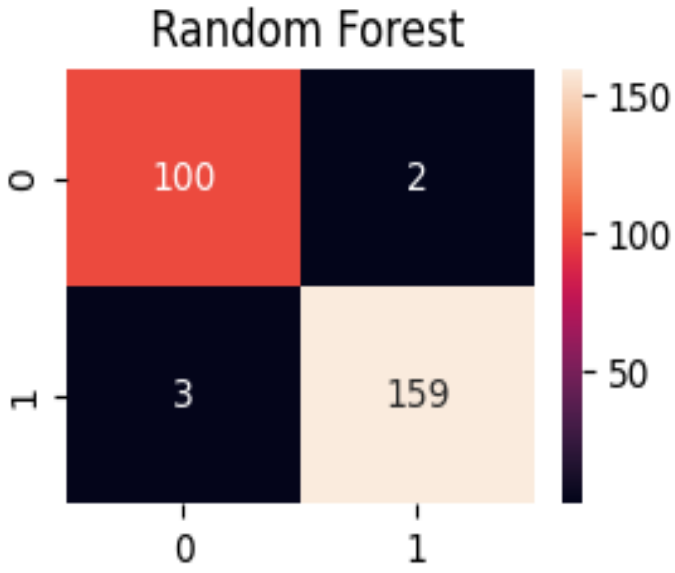

Figure 5.4 Confusion matrix of RF

Figure 5.5 shows the comparative performance of various models used in predicting cardiac disease. The figure presents essential assessment data, including accuracy, precision, recall, and F1-score, for each model assessed. Performance is quantified using a percentage metric and offers a comprehensive evaluation of each model's accuracy in predicting cardiac disease. The values for each metric are as follows: Accuracy 98%, Precision 99%, Recall 98%, F1-score 99%.

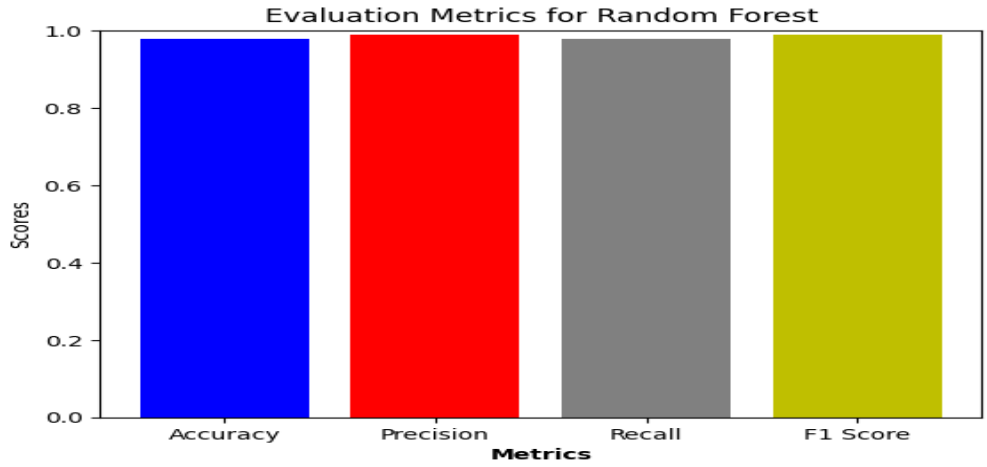

Figure 5.5 Evaluation metrics for RF

The ROC Curve assesses the predictive capability of the RF model for heart disease by plotting the relationship between the false positive rate and the true positive rate at various threshold levels. The model's AUC is 0.99, which indicates high efficacy in predicting heart disease (Figure 5.6).

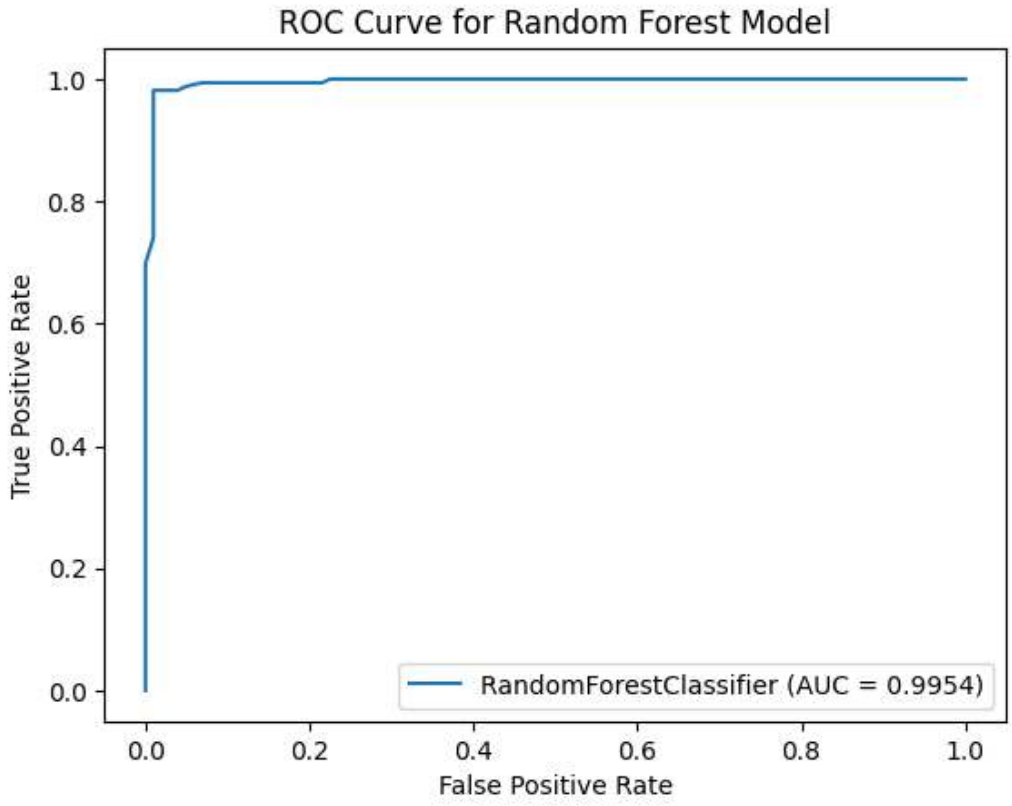

Figure 5.6 ROC curve for RF

5.1.3 K-nearest neighbors

In the test data set where the KNN algorithm was applied, predictions were made for 264 patients. Correct predictions were made for 41 people without heart disease and 119 people with heart disease. Incorrect predictions were made for 61 people without heart disease and 43 people with heart disease. KNN algorithm applied model; the accuracy of the test data set was 60% (Figure 5.7).

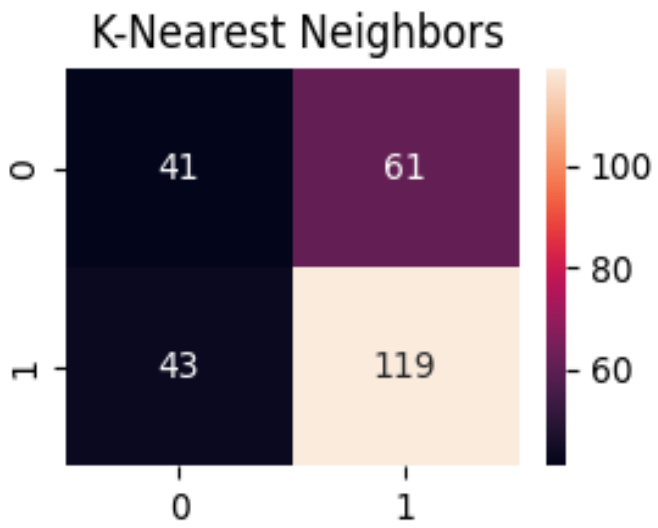

Figure 5.7 KNN confusion matrix

Figure 5.8 shows the comparison of performance metrics for heart disease prediction models. The performance is measured in percentage and offers an overview of the effectiveness of each model in predicting heart disease. The values for each metric are as follows: Accuracy 60%, Precision 66%, Recall 73%, F1-score 70%.

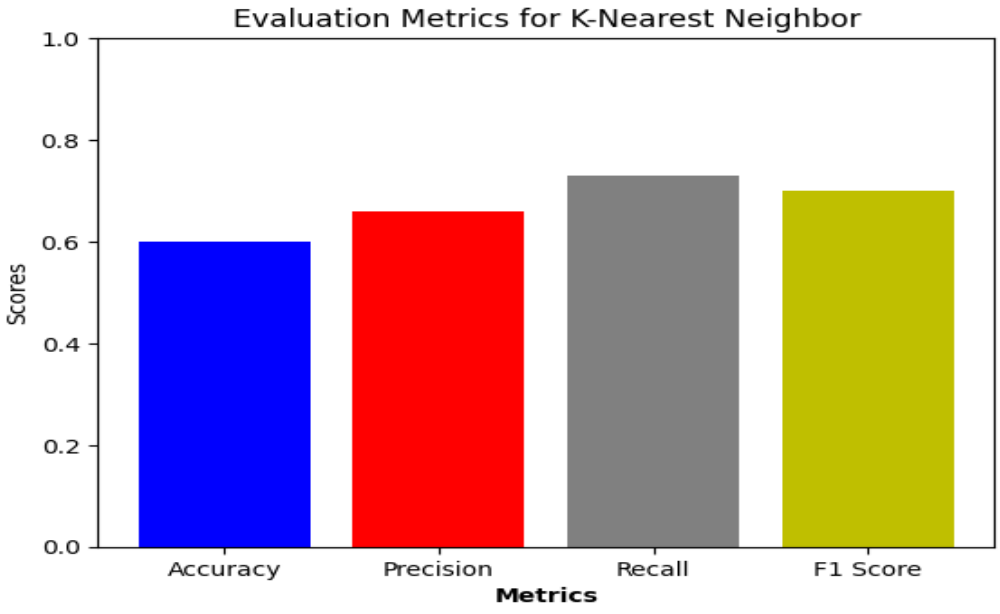

Figure 5.8 Evaluation metrics for KNN

ROC Curve in Figure 5.9 assesses the KNN model's ability to predict heart disease by plotting the false positive rate against the true positive rate at varying classification thresholds. ROC curve for model is 0.59, suggesting its effectiveness in predicting heart disease.

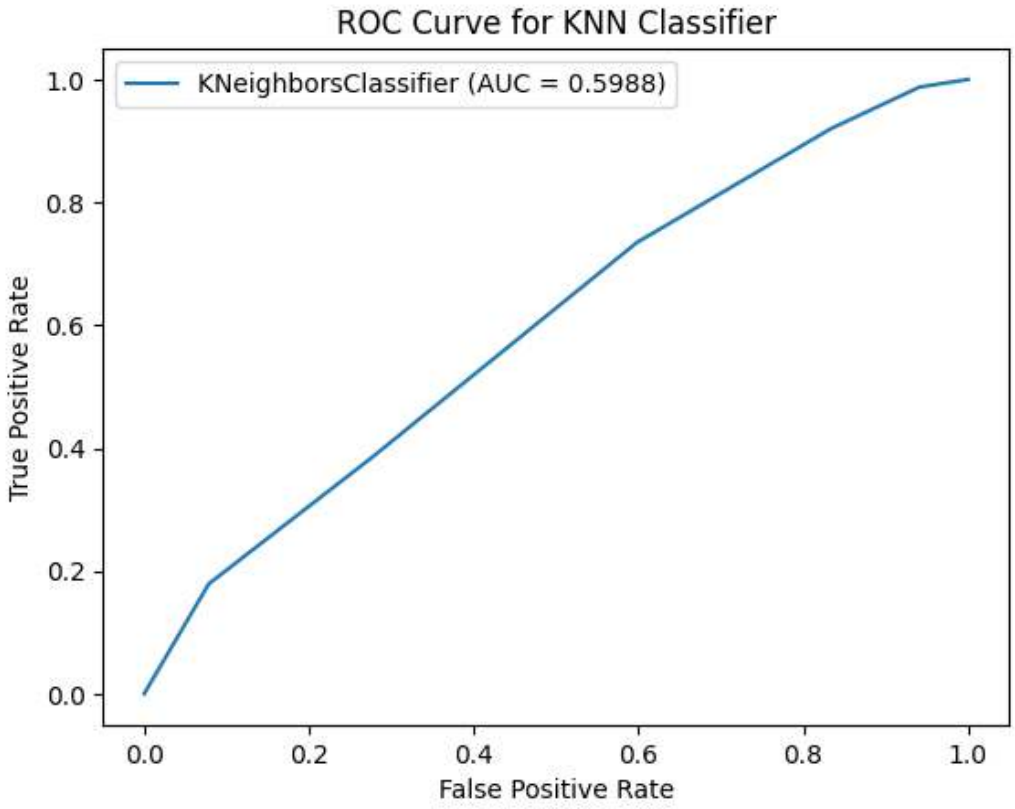

Figure 5.9 ROC curve for KNN

5.1.4 Decision tree

In the test data set where the DT algorithm was applied, predictions were made for 264 patients. Correct predictions were made for 101 people without heart disease and 160 people with heart disease. Incorrect predictions were made for 1 people without heart disease and 2 people with heart disease. DT algorithm applied model; the accuracy of the test data set was 98% (Figure 5.10).

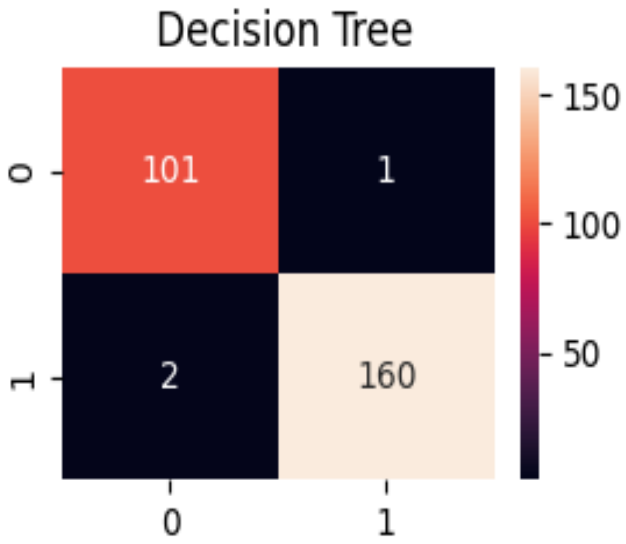

Figure 5.10 Confusion matrix of DT

Figure 5.11 shows the comparison of performance metrics for heart disease prediction models. The performance is measured in percentage and offers an overview of the effectiveness of each model in predicting heart disease. The values for each metric are as follows: Accuracy 98%, Precision 99%, Recall 98%, F1-score 98%.

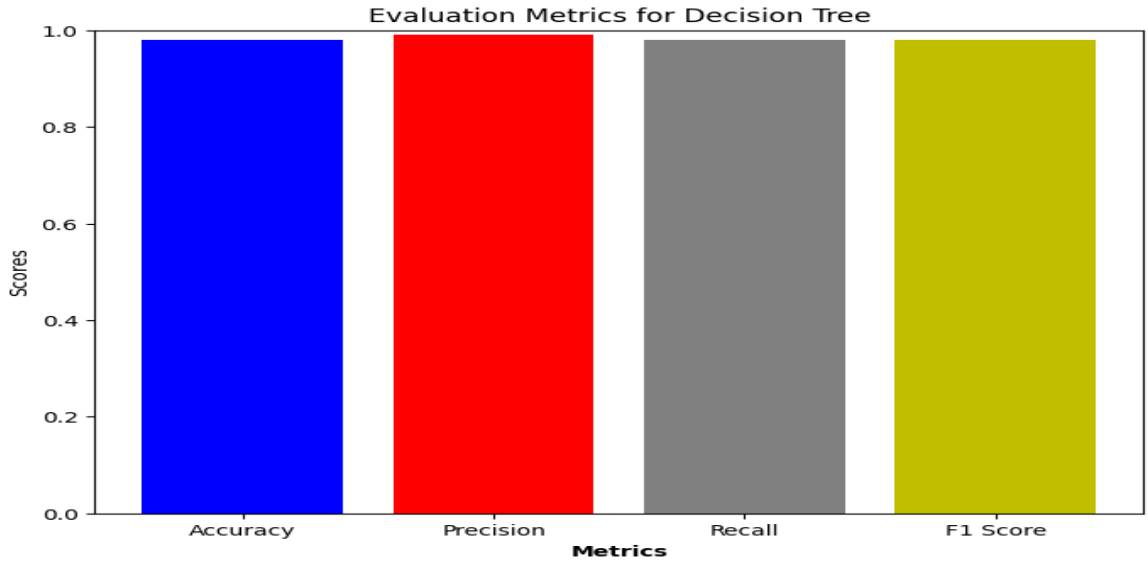

Figure 5.11 Evaluation metrics for DT

The ROC curve evaluates the DT model's ability to predict heart disease by plotting the false positive rate against the true positive rate at various classification thresholds. The ROC curve for this model shows an AUC of 0.98, indicating its high effectiveness in predicting heart disease (Figure 5.12).

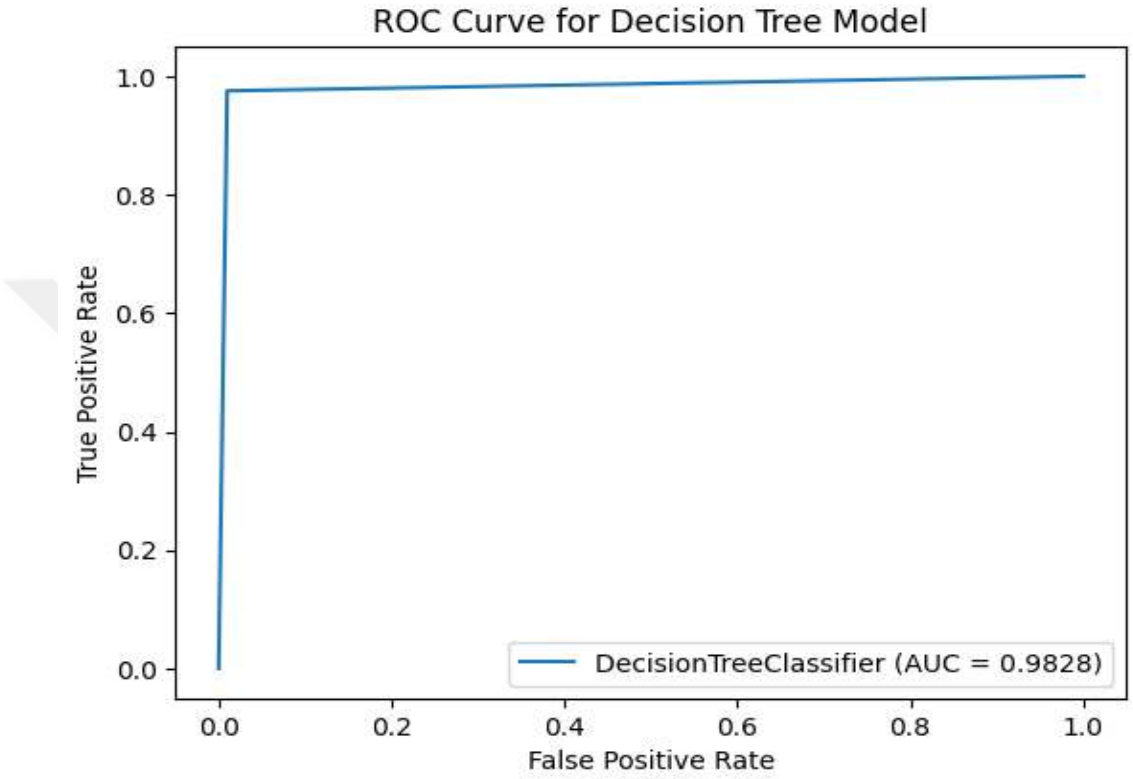

Figure 5.12 ROC curve for DT

5.1.5 Support vector machine

In the test data set where the SVM algorithm was applied, predictions were made for 264 patients. Correct predictions were made for 34 people without heart disease and 134 people with heart disease. Incorrect predictions were made for 68 people without heart disease and 28 people with heart disease. SVM algorithm applied model; the accuracy of the test data set was 63%.

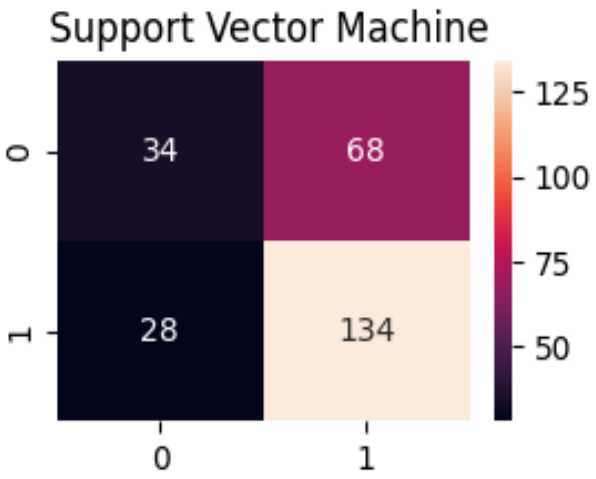

Figure 5.13 SVM confusion matrix

In Figure 5.14, comparison of performance metrics for heart disease prediction models. The performance is measured in percentage and offers an overview of the effectiveness of each model in predicting heart disease. The values for each metric are as follows: Accuracy 63%, Precision 66%, Recall 83%, F1-score 74%.

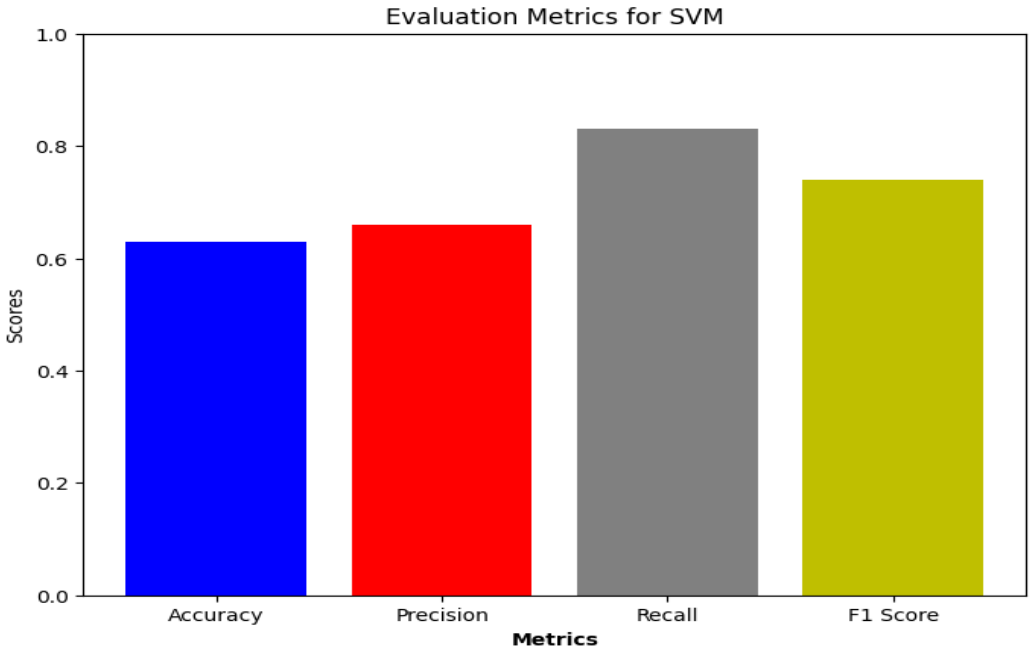

Figure 5.14 Evaluation metrics for SVM

The ROC curve evaluates the SVM model's ability to predict heart disease by plotting the false positive rate against the true positive rate at various classification thresholds. The ROC curve for this model shows an AUC of 0.68, indicating its high effectiveness in predicting heart disease (Figure 5.15).

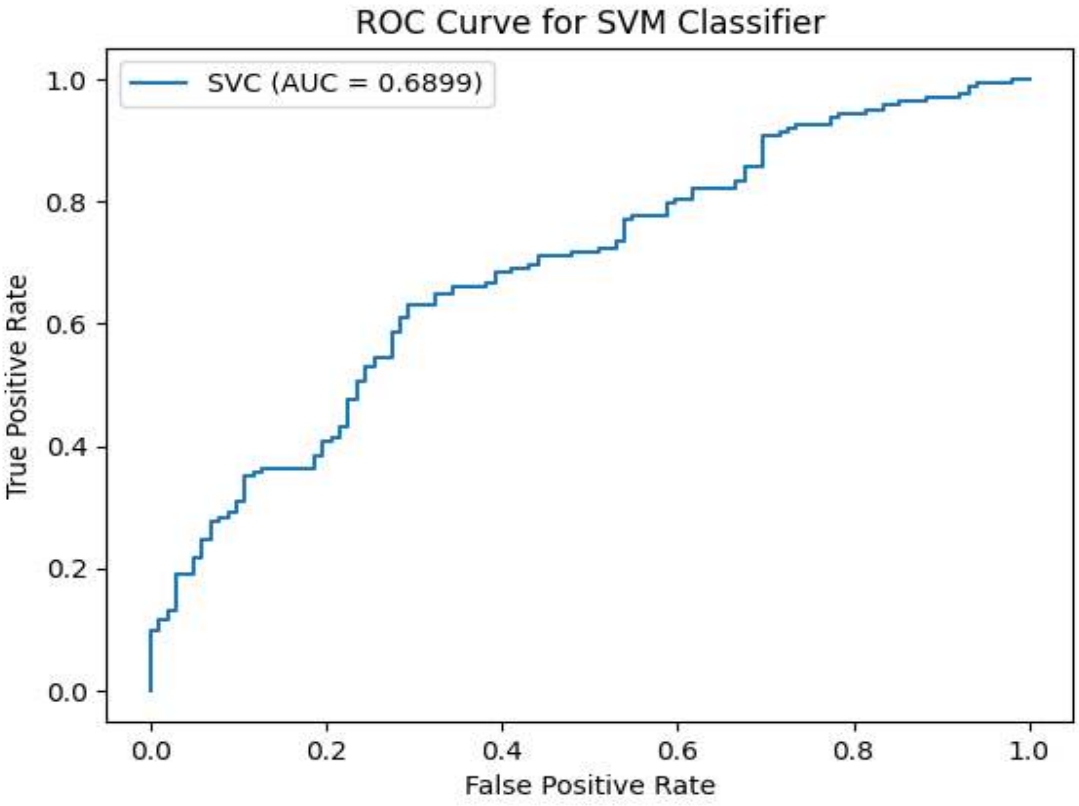

Figure 5.15 ROC curve for SVM

5.1.6 Naive bayes

In the test data set where the NB algorithm was applied, predictions were made for 264 patients. Correct predictions were made for 102 people without heart disease and 71 people with heart disease. Incorrect predictions were made for 0 people without heart disease and 91 people with heart disease. NB algorithm applied model; the accuracy of the test data set was 66% (Figure 5.16).

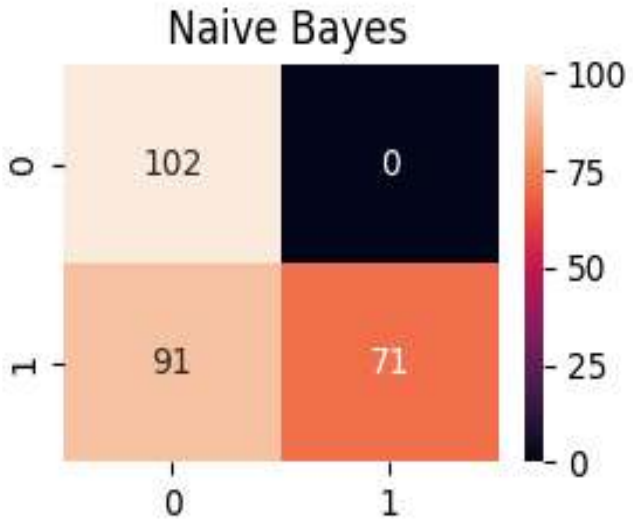

Figure 5.16 Confusion matrix of NB

Figure 5.17 shows the comparison of NB performance metrics for heart disease prediction models. The performance is measured in percentage and offers an overview of the effectiveness of each model in predicting heart disease. The values for each metric are as follows: Accuracy 66%, Precision 99%, Recall 44%, F1-score 61%.

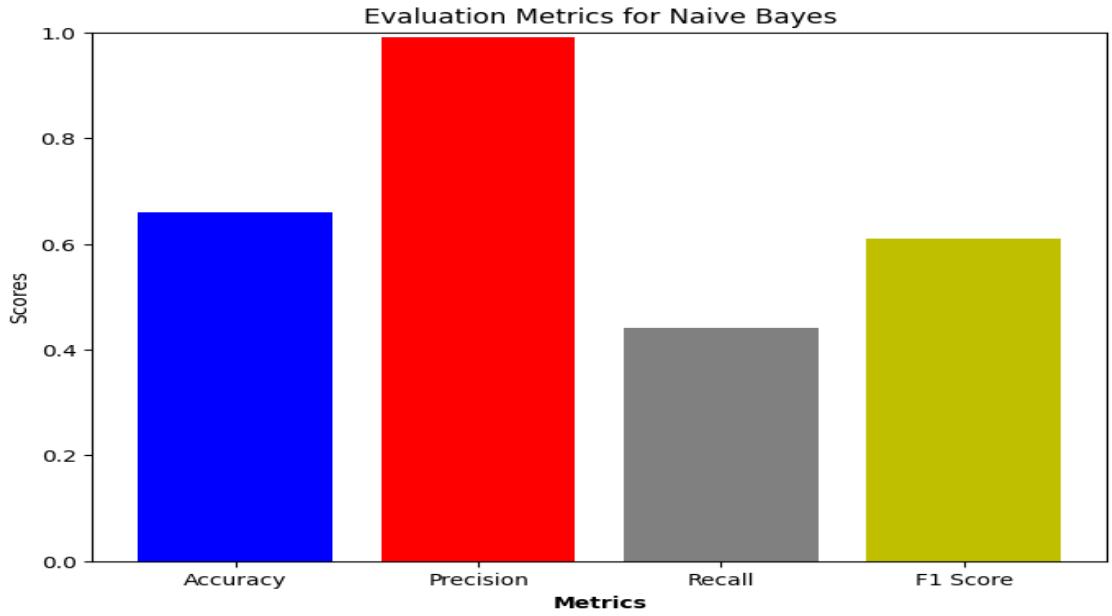

Figure 5.17 Evaluation metrics for NB

The ROC curve evaluates the NB model's ability to predict heart disease by plotting the false positive rate against the true positive rate at various classification thresholds. The ROC curve for this model shows an AUC of 0.82, indicating its high effectiveness in predicting heart disease (Figure 5.18).

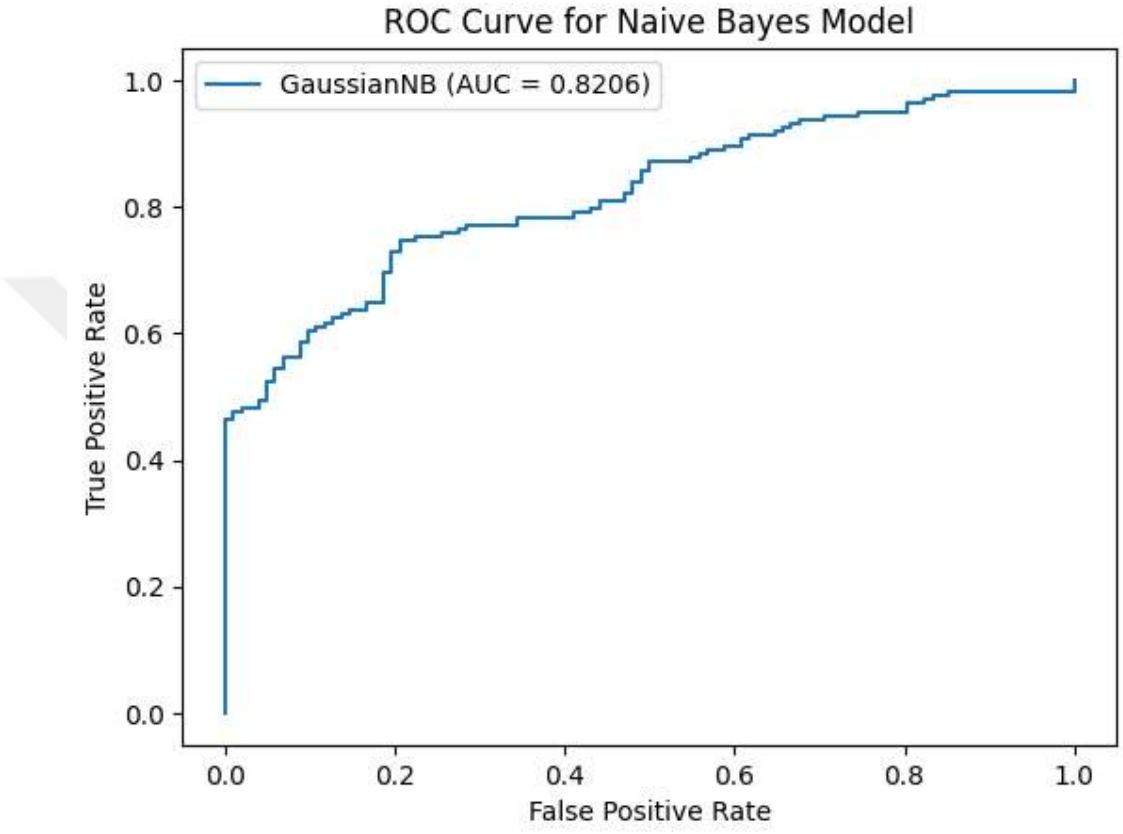

Figure 5.18 ROC curve for NB

5.1.7 Ensemble learning model

In the test data set where the Bagging algorithm was applied, predictions were made for 264 patients. Correct predictions were made for 101 people without heart disease and 159 people with heart disease. Incorrect predictions were made for 1 people without heart disease and 3 people with heart disease. The bagging algorithm applied the model; the accuracy of the test data set was 99% (Figure 5.19).

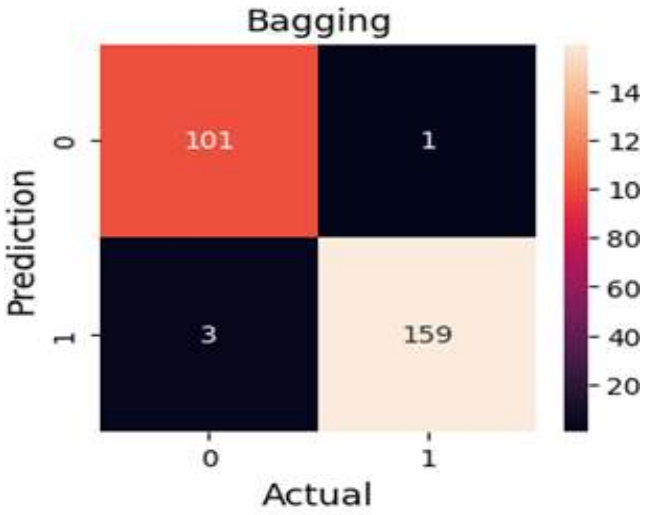

Figure 5.19 Confusion matrix of bagging

In Figure 5.20, comparison of bagging performance metrics for heart disease prediction models. The performance is measured in percentage and offers an overview of the effectiveness of each model in predicting heart disease. The values for each metric are as follows: Accuracy 99%, Precision 99%, Recall 99%, F1-score 99%.

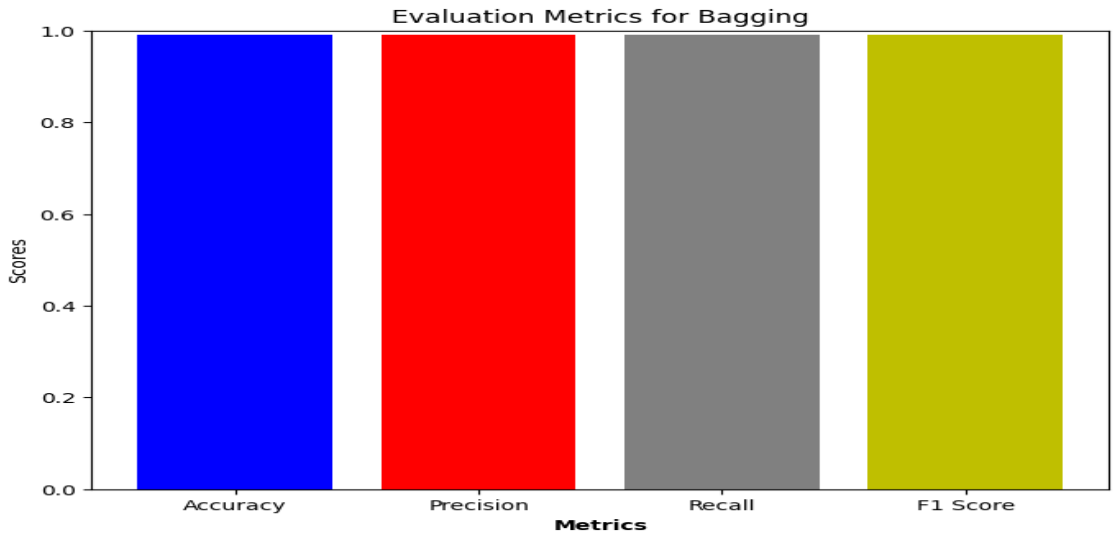

Figure 5.20 Evaluation metrics for bagging

The ROC curve evaluates the bagging model ability to predict heart disease by plotting the false positive rate against the true positive rate at various classification thresholds.

The ROC curve for this model shows an AUC of 0.99, indicating its high effectiveness in predicting heart disease (Figure 5.21).

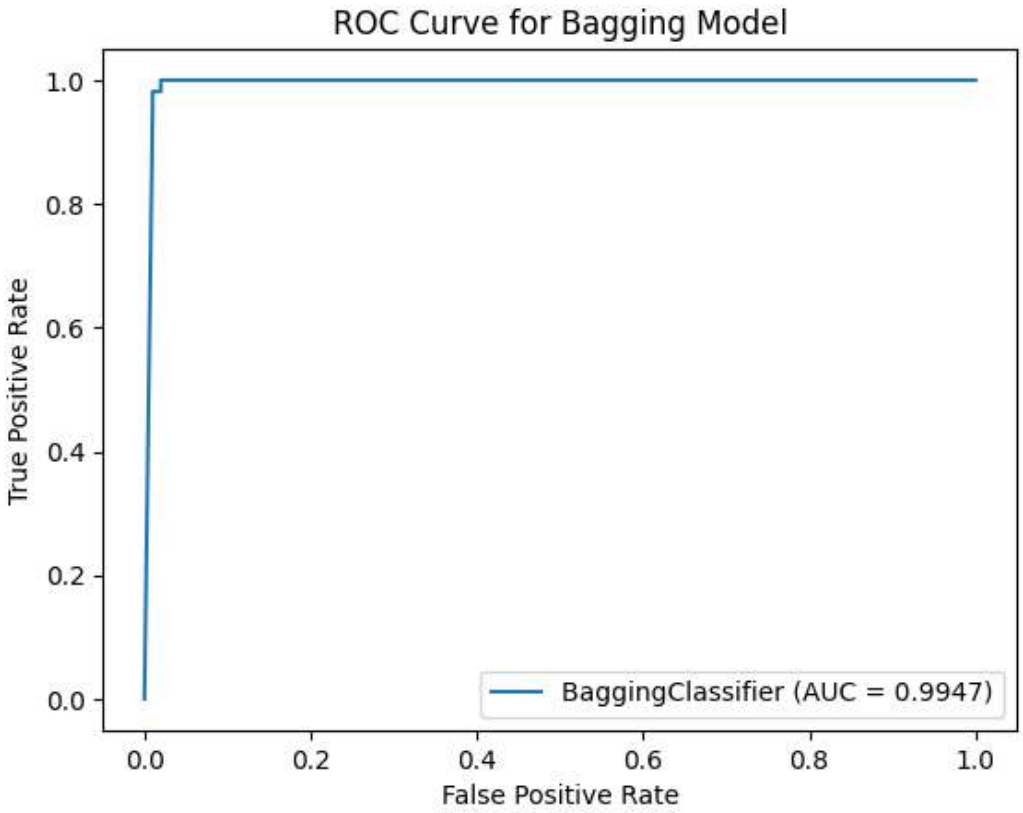

Figure 5.21 ROC curve for bagging

As Figure 5.22, the test data set where the stacking algorithm was applied, predictions were made for 264 patients. Correct predictions were made for 100 people without heart disease and 159 people with heart disease. Incorrect predictions were made for 2 people without heart disease and 3 people with heart disease. The stacking algorithm applied the model; the accuracy of the test data set was 98%.

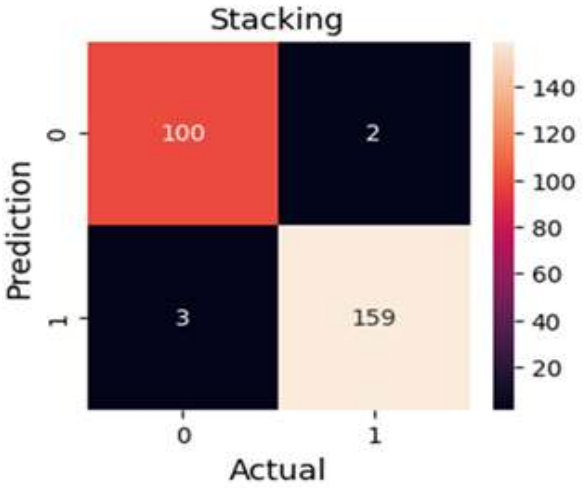

Figure 5.22 Confusion matrix of stacking

In Figure 5.23 comparison of stacking performance metrics for heart disease prediction models. The performance is measured in percentage and offers an overview of the effectiveness of each model in predicting heart disease. The values for each metric are as follows: Accuracy 98%, Precision 99%, Recall 98%, F1-score 99%.

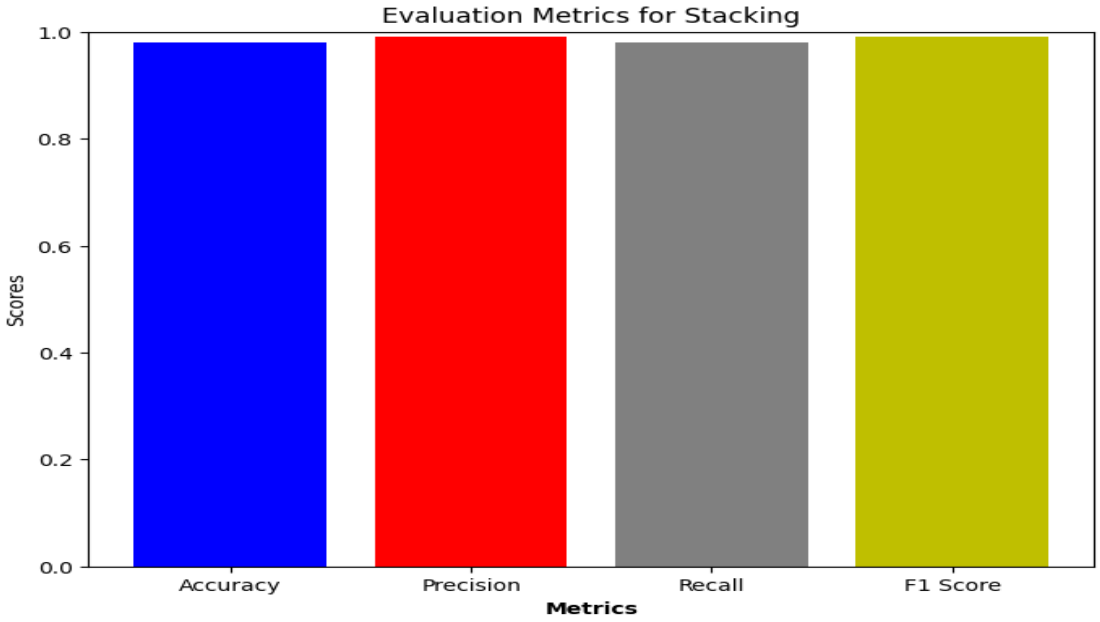

Figure 5.23 Evaluation metrics for stacking

ROC Curve assesses the stacking model's ability to predict heart disease by plotting the false positive rate against the true positive rate at varying classification thresholds. The

Area Under the Curve for this model is 0.99, suggesting its effectiveness in predicting heart disease.

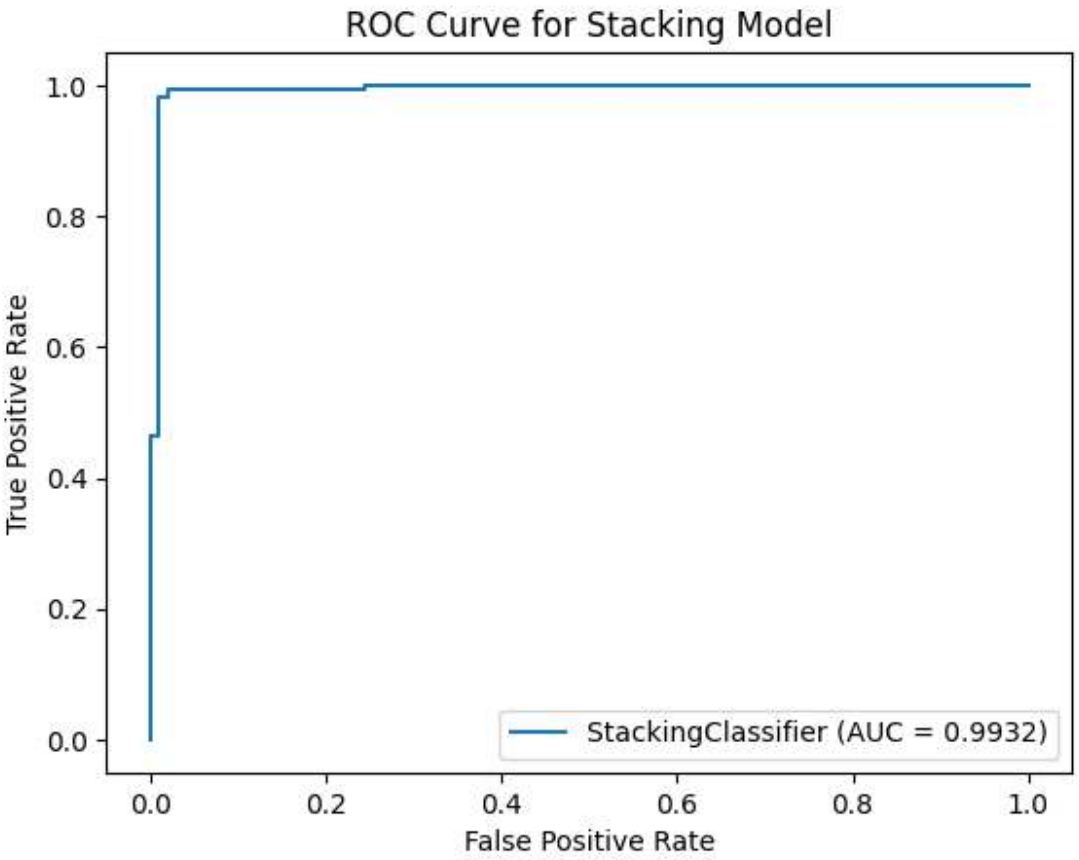

Figure 5.24 ROC curve for stacking

In the test data set where the Bagging algorithm was applied, predictions were made for 264 patients. Correct predictions were made for 98 people without heart disease and 158 people with heart disease. Incorrect predictions were made for 4 people without heart disease and 4 people with heart disease. Boosting algorithm applied model; the accuracy of the test data set was 97% (Figure 5.25).

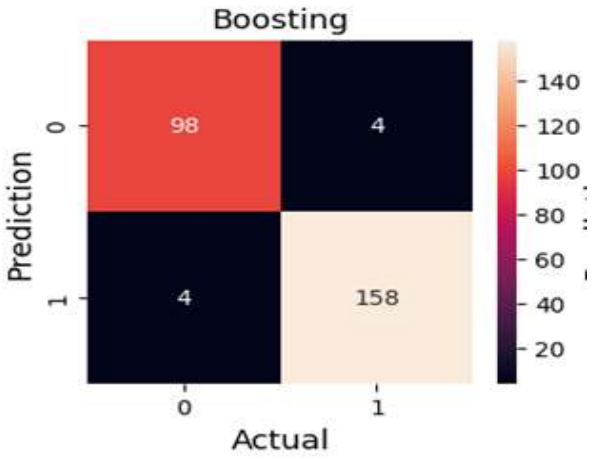

Figure 5.25 Confusion matrix of boosting

Figure 5.26 comparison of boosting performance metrics for heart disease prediction models.The performance is measured in percentage and offers an overview of the effectiveness of each model in predicting heart disease. The values for each metric are as follows: Accuracy 97%, Precision 98%, Recall 98%, F1-score 98%.

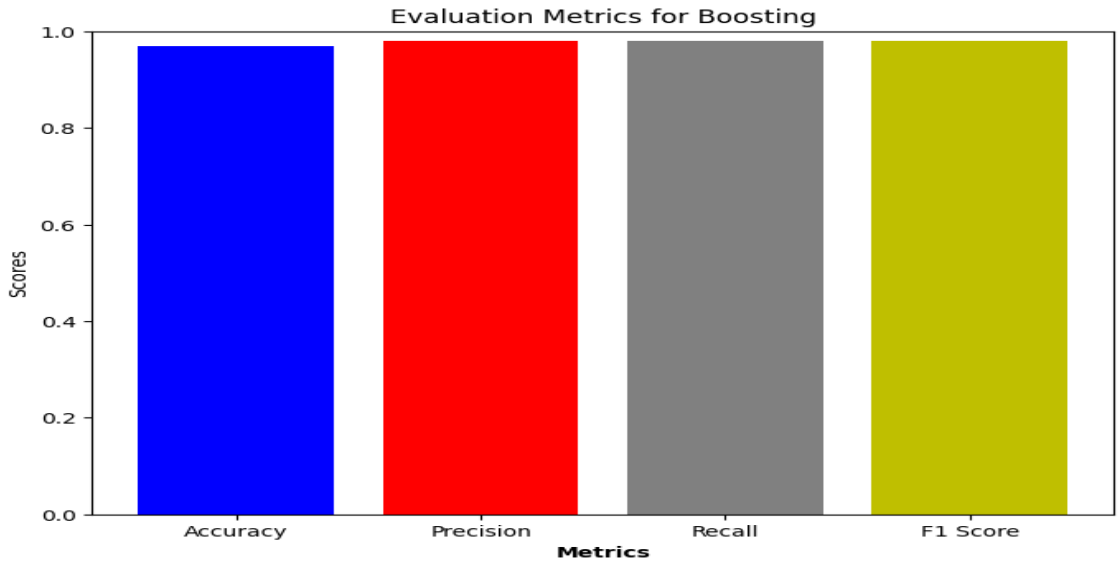

Figure 5.26 Evaluation metrics for boosting

ROC Curve assesses the boosting model's ability to predict heart disease by plotting the false positive rate against the true positive rate at varying classification thresholds. The AUC for boosting model as shown in Figure 5.27 is 0.99, suggesting its effectiveness in predicting heart disease.

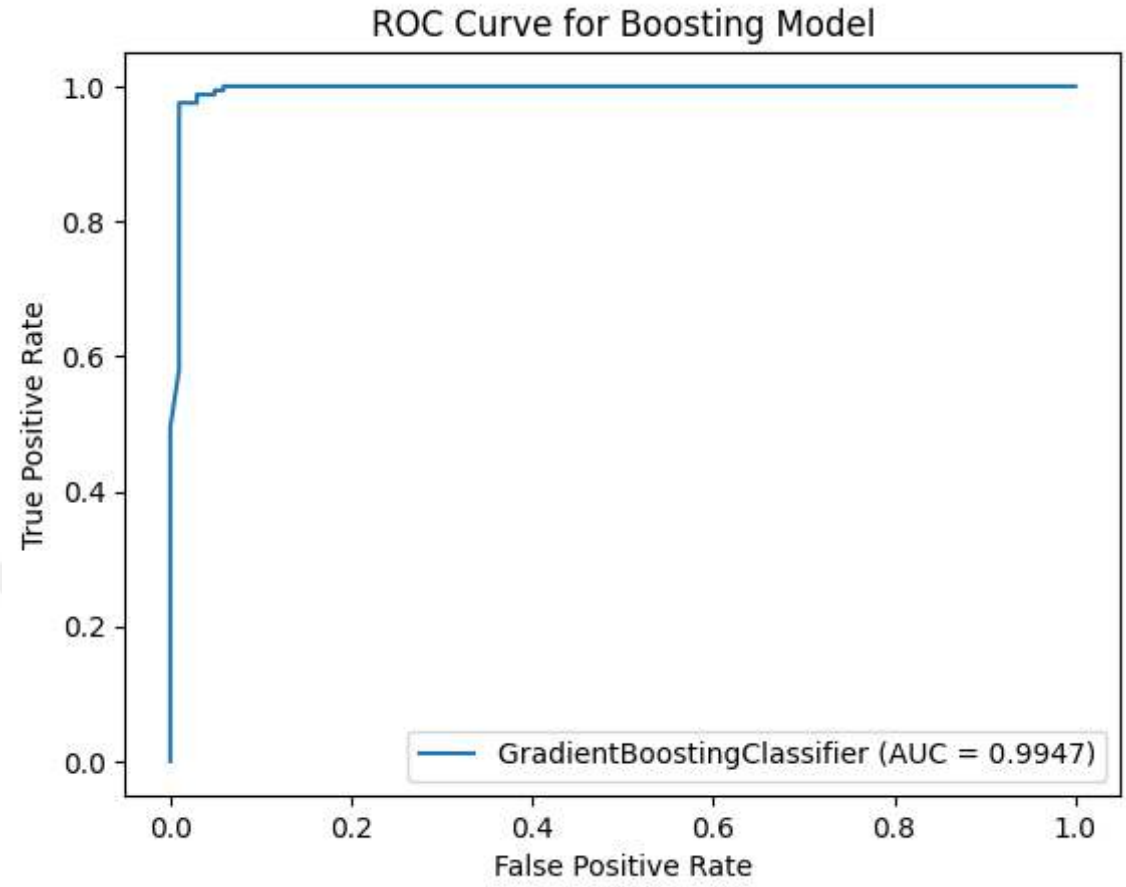

Figure 5.27 ROC curve for boosting

Table 5.1 Performance value of models

MODEL	PRECISION	RECALL	F1-SCORE	AUC-ROC	ACCURACY
Boosting	98%	98%	98%	99%	97%
Bagging	99%	99%	99%	99%	99%
Stacking	99%	98%	99%	99%	98%
LR	82%	82%	82%	86%	78%
RF	99%	98%	99%	99%	98%
DT	99%	98%	98%	98%	98%
KNN	66%	73%	70%	59%	60%
SVM	66%	83%	74%	68%	63%
NB	99%	44%	61%	82%	66%

5.2 Discussion

- The evaluation of ML models, particularly those using Bagging as an Ensemble Learning algorithm, indicates a significant performance difference between the training and test datasets. While the models obtained extraordinary accuracy rates of

99% on the training dataset, they plummeted to 60% when applied to the independent test dataset.

- The significant decrease in accuracy on the test dataset raises concerns about overfitting during model training. Overfitting occurs when the model learns too much about the training dataset, catching noise and outliers rather than actual patterns. As a result, the model may fail to generalize to new and unknown data, resulting in lower performance on the test dataset.

Bagging, as an ensemble learning algorithm, performed admirably during the training dataset evaluation. Bagging enabled the ensemble of models to properly learn the complexity of the training data, attaining near-perfect accuracy. However, this excellent performance on the training dataset did not convert effortlessly to the test dataset, underscoring the significance of evaluating models' generalization capabilities.

6. CONCLUSIONS AND RECOMMENDATIONS

To sum up, this study has explored the use of ML, especially ensemble learning techniques, in the prediction of cardiac disease. Boosting, bagging, and stacking three ensemble learning strategies performed remarkably well in the study, with accuracy rates of 97%, 99%, and 98%, respectively. This accomplishment highlights how sophisticated algorithms may be used to increase the precision of heart disease diagnosis a vital component of modern healthcare. The best performance, bagging, showed remarkable precision as well as consistency, making it a good fit for practical application. These findings have practical consequences for both public health programs and individual patient care that go beyond scholarly investigation. Thinking about the future of medicine, the results force a reevaluation of diagnostic approaches. Combining group learning with bagging, in particular, shows promise as a revolutionary tool that may identify those at risk more quickly and accurately. Better health outcomes and more efficient medical interventions are possible as a result of this. The study's inherent limitations, such as the features of the dataset, are acknowledged, but the overall message is still very clear: ML, with an emphasis on ensemble approaches, has the potential to completely change the way that heart disease is diagnosed. It urges medical personnel to adopt these developments to manage and prevent heart illnesses in the future in a more proactive, data-driven, and ultimately successful manner.

REFERENCES

Adhikari, B., & Shakya, S. 2022. Heart disease prediction using ensemble model. lecture notes in networks and systems, 857–868

Ali, A. A., Hassan, H. S., & Anwar, E. M. 2020. Heart diseases diagnosis based on a novel convolution neural network and gate recurrent unit technique. In: 12th International Conference on Electrical Engineering.

Ali, M. M., Paul, B. K., Ahmed, K., Bui, F. M., Quinn, J. M. W., & Moni, M. A. 2021. Heart disease prediction using supervised machine learning algorithms: Performance analysis and comparison. Computers in Biology and Medicine, 136, 104672.

Almustafa, K. M. 2020. Prediction of heart disease and classifiers' sensitivity analysis. BMC Bioinformatics, 21(1).

Alon, N., Gonen, A., Hazan, E., & Moran, S. (2021). Boosting simple learners. *ArXiv (Cornell University)*.

Atallah, R., & Al-Mousa, A. 2019. Heart disease detection using machine learning majority voting ensemble method. 2019 2nd International Conference on New Trends in Computing Sciences.

Bharti, R., Khamparia, A., Shabaz, M., Dhiman, G., Pande, S., & Singh, P. 2021. Prediction of heart disease using a combination of machine learning and deep learning. Computational Intelligence and Neuroscience, 2021, 1–11.

Çil, E., & Güneş, A. 2021. Makine öğrenmesi algoritmalarıyla kalp hastalıklarının tespit edilmesine yönelik performans analizi Performance analysis of machine learning algorithms used to detect heart diseases. Anadolu Bil Meslek Yüksekokulu Dergisi, 17(65): 57–77.

Deshpande, N. M., Gite, S., & Aluvalu, R. (2021). A review of microscopic analysis of blood cells for disease detection with AI perspective. *PeerJ Computer Science*, *7*, e460.

Diwakar, M., Tripathi, A., Joshi, K., Memoria, M., Singh, P., & kumar, N. 2021. Latest trends on heart disease prediction using machine learning and image fusion. Materials Today: Proceedings, 37, 3213–3218.

Gao, X.-Y., Amin Ali, A., Shaban Hassan, H., & Anwar, E. M. 2021. Improving the accuracy for analyzing heart diseases prediction based on the ensemble method. Complexity, 2021, 1–10.

Ghosh, S., Dasgupta, A., & Swetapadma, A. (2019). A Study on Support Vector Machine based Linear and Non-Linear Pattern Classification. IEEE Xplore.

JAVATPOINT. 2021. Website: https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning, Date of access: 02.01.2021

Jindal, H., Agrawal, S., Khera, R., Jain, R., & Nagrath, P. 2021. Heart disease prediction using machine learning algorithms. IOP Conference Series: Materials Science and Engineering, 1022: 012072.

Katarya, R., & Meena, S. K. 2020. Machine learning techniques for heart disease prediction: a comparative study and analysis. Health and Technology.

Khandelwal, Y. (2021, August 13). *Ensemble Stacking for Machine Learning and Deep Learning*. Analytics Vidhya.

Mallela, R. C., Bhavani, R. L., & Ankayarkanni, B. 2021. Disease prediction using machine learning techniques, In: 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI).

Masoodi, F. S., Abrar, I., & Bamhdi, A. M. (2022). An Effective Intrusion Detection System Using Homogeneous Ensemble Techniques. *International Journal of Information Security and Privacy*, *16*(1), 1–18.

MEDIPOL, T. 2015. Website: https://medipol.com.tr/kurumsal/haberler/medyada-medipol/turkiyede-her-25-dakikada-bir-kisi-kalp-damar-hastaliklarindan-oluyor. Date of access: 03.08.2015.

Mendonça, F., Mostafa, S. S., Morgado-Dias, F., Ravelo-García, A. G., & Figueiredo,. (2022). ProBoost: a Boosting Method for Probabilistic Classifiers. *ArXiv (Cornell University)*.

Møller, H. M., Larsen, K. G., & Ritzert, M. (2023). AdaBoost is not an Optimal Weak to Strong Learner. *ArXiv (Cornell University)*.

Nahar, J., Imam, T., Tickle, K. S., & Chen, Y.-P. P. 2013. Association rule mining to detect factors which contribute to heart disease in males and females. Expert Systems with Applications, 40(4): 1086–1093.

Obasi, T., & Omair Shafiq, M. 2019. Towards comparing and using machine learning techniques for detecting and predicting Heart Attack and Diseases. 2019 IEEE International Conference on Big Data.

Patel, J., Khaked, A. A., Patel, J., & Patel, J. 2021. Heart disease prediction using machine learning. Lecture Notes in Networks and Systems, 653–665.

Pradhan, N., & Singh, A. S. 2020. Machine learning architecture and framework. machine learning and the internet of medical things in healthcare, Scrivener, 1–24.

Rajdhan, A., Agarwal, A., Sai, M., Ravi, D., & Ghuli, Dr. P. 2020. Heart disease prediction using machine learning. International Journal of Engineering Research And, V9(04).

Rashid, Tarik A.; Hassan, Bryar 2022, Heart Attack Dataset, Mendeley Data, V1.

Reddy, N. S. C., Shue Nee, S., Zhi Min, L., & Xin Ying, C. 2019. Classification and feature selection approaches by machine learning techniques: heart disease prediction. International Journal of Innovative Computing, 9(1).

Sagheer, A., Zidan, M., & Abdelsamea, M. M. 2019. A novel autonomous perceptron model for pattern classification applications. Entropy, 21(8): 763.

SAINT-CIRGUE, G. 2019. Website: https://www.yumpu.com/fr/document/view/67482157/apprendre-le-machine-learning-en-une-semaine-guillaume-saint-cirgue-z-liborg. Date of access: 03.09.2019

Sapra, L., Sandhu, J. K., & Goyal, N. 2021. Intelligent method for detection of coronary artery disease with ensemble approach. 1033–1042.

Shobana, V., & Nandhini, K. (2022). Ensemble Techniques to improve the performance of the High Dimensional MultiClass Algorithms. *2022 First International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT)*.

Shouman, M., Turner, T., & Stocker, R. 2012. Applying k-nearest neighbour in diagnosing heart disease patients. International Journal of Information and Education Technology, 220–223.

Srivastava, K., & Choubey, D. K. 2020. Heart disease prediction using machine learning and data mining. International Journal of Recent Technology and Engineering, 9(1): 21–219.

Wang, J., Li, F., Li, J., Hou, C., Qian, Y., & Liang, J. (2024). RSS-Bagging: Improving Generalization Through the Fisher Information of Training Data. *IEEE Transactions on Neural Networks and Learning Systems*, 1–15.

WHO. 2023. Website: https://www.who.int/health-topics/cardiovascular-diseases#tab=tab_1%20.%202022. Date of access: 02.04.2023.

CURRICULUM VITAE

Personal Information

Name and Surname : Mohamed ALI YOUSSOUF

Education

MSc	Çankırı Karatekin University Graduate School of Natural and Applied Sciences Department of Electronics and Computer Engineering	2021-2024
Undergraduate	University of Djibouti Faculty of Science Department of Mathematics	2017-2020

Work Experience

Year	Institution	Position
2023-Present	TechPro Education	Front-End Developer

Academic Activities

1. Mohamed Ali Youssouf & Fuat Türk, 2023. Heart Disease Prediction Using Machine Learning, In:2nd International Karatekin Science and Technology Conference, pp. 624-626, Çankırı, Turkyie. ISBN: 978-605-82910-8-9.