HEART DISEASE PREDICTION USING MACHINE LEARNING

Project Report

Submitted in partial fulfillment of the requirements for the
Degree of Bachelor of Technology in Electronics and Communication Engineering
under Maulana Abul Kalam Azad University of Technology

By

SHAINA FAAIZ
ROLL NUMBER: 14200320037

Under the guidance of

DR. MANASH CHANDA

Associate Professor & HOD
Department of Electronics and Communication Engineering
Meghnad Saha Institute of Technology
Kolkata – 700150

2024

Meghnad Saha Institute of Technology

Nazirabad, P.O. : Utchepota, Via Sonarpur, Kolkata 700 150, Phone : 033-2443-1754, Telefax : 033-2443-1032

CERTIFICATE OF APPROVAL

I hereby recommend that the work in preparing the project report entitled " **HEART DISEASE PREDICTION USING MACHINE LEARNING**" carried out by **SHAINA FAAIZ** under my supervision may be accepted in partial fulfilment of the requirements for the degree of Bachelor of Technology in Electronics and Communication Engineering of Maulana Abul Kalam Azad University of Technology(MAKAUT).

…………………………………………

Dr.Manash Chanda
Project Supervisor, Dept of ECE
Meghnad Saha Institute of Technology
Kolkata – 700150

Countersigned,

……………………………………………..

Dr. Manash Chanda
HOD, Dept of ECE
Meghnad Saha Institute of Technology
Kolkata – 700150

ACKNOWLEDGEMENTS

I would like to express my sincere regards Dr. Manash Chanda, my project supervisor and HOD of Electronics and Communication Engineering Department, Meghnad Saha Institute of Technology for his guidance, valuable advice and constructive suggestions for carrying out this project work.

I would like to record my indebtedness to Dr. Manash Chanda, HOD, Department of Electronics and Communication Engineering and Prof. Tirthankar Datta, Principal, Meghnad Saha Institute of Technology for providing me with all the support that was needed.

I would also like to thank all the faculty members of ECE department, MSIT for their valuable suggestions during the course of my work.

Finally, my sincere thanks go to my parents for their encouragement and support during this project work.

……………………………………….

SHAINA FAAIZ
ROLL NO. - 14200320037

CONTENT

Page Number

CHAPTER - 1

1.1 INTRODUCTION

Cardiovascular diseases (CVDs) have consistently been the leading cause of death worldwide over the past decade. According to the World Health Organization (WHO), it is estimated that over 17.9 million deaths occur each year due to cardiovascular diseases. Of these deaths, a staggering 80% are attributed to coronary artery disease and cerebral stroke. This alarming statistic underscores the critical importance of addressing CVDs through improved medical diagnosis, prevention, and treatment strategies.

Several habitual factors, including personal and professional habits, as well as genetic predisposition, contribute significantly to the development of heart disease. Risk factors such as smoking, excessive alcohol consumption, high caffeine intake, chronic stress, and physical inactivity are well-documented contributors to heart disease. These lifestyle choices, combined with physiological factors like obesity, hypertension, high blood cholesterol, and pre-existing heart conditions, often serve as decisive factors in the onset and progression of cardiovascular diseases. Efficient, accurate, and early medical diagnosis of heart disease is crucial in implementing preventive measures that can avoid the severe complications associated with these diseases. Early diagnosis not only enhances the chances of successful treatment but also significantly reduces the burden on healthcare systems by preventing advanced disease stages that require more intensive and costly interventions.

The major challenge faced in medical science today is providing quality healthcare services while ensuring efficient and accurate disease prediction. This challenge is particularly pertinent in the realm of cardiovascular diseases, where early detection can drastically alter patient outcomes. Automation through Data Mining and Machine Learning (ML) offers promising solutions to these challenges by enhancing the precision and efficiency of medical diagnoses.

Data mining is defined as the process of extracting usable information from large datasets. It involves analyzing patterns in vast amounts of data using various software tools. This process encompasses effective data collection, warehousing, and computer processing, enabling the extraction of meaningful insights that can inform medical diagnoses and treatment plans. Machine Learning, a subfield of data mining, is particularly adept at handling large-scale, well-formatted datasets. In the medical field, ML algorithms can be utilized for diagnosing, detecting, and predicting a wide range of diseases. By leveraging historical patient data and identifying patterns that may not be immediately apparent to human analysts, ML algorithms can significantly enhance the accuracy of medical diagnoses.

Various Machine Learning algorithms have been developed and employed in the prediction and diagnosis of heart disease. These include Logistic Regression, Naïve Bayes, Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Decision Tree, Random Forest. Each of these algorithms has its own strengths and weaknesses, and their effectiveness can vary depending on the specific characteristics of the dataset and the problem at hand.

Logistic Regression is a statistical method that models the probability of a binary outcome based on one or more predictor variables. It is particularly useful for understanding the relationship between risk factors and the likelihood of developing heart disease. Despite its simplicity, Logistic Regression can provide valuable insights and is often used as a baseline model in predictive analytics.

Support Vector Machine (SVM) is a powerful classification algorithm that aims to find the hyperplane that best separates the data into different classes. SVMs are particularly effective in high-dimensional spaces and are known for their robustness in handling complex, non-linear relationships.

Decision Tree algorithms create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. Decision Trees are easy to interpret and visualize, making them a popular choice for medical diagnosis tasks. However, they can be prone to overfitting, especially when dealing with complex datasets.

Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes for classification tasks. By aggregating the results of numerous decision trees, Random Forest can significantly reduce the risk of overfitting and improve predictive accuracy.

In this project, we will use the heart disease dataset from Kaggle datasets repository to compare these various machine learning algorithms. The goal is to identify the most accurate model for predicting heart disease, which can then be utilized to assist healthcare professionals in making informed decisions about patient care.

The process will involve several key steps: data preprocessing, feature selection, model training, and evaluation. Data preprocessing will include handling missing values, normalizing the data, and converting categorical variables into numerical formats. Feature selection will help in identifying the most relevant features that contribute to heart disease prediction, thereby improving model performance and interpretability.

Model training will involve using a portion of the dataset to train each machine learning algorithm. The trained models will then be evaluated using various metrics such as accuracy, precision, recall, and F1-score to determine their effectiveness. Cross-validation techniques will be employed to ensure the robustness and generalizability of the models.

By systematically comparing the performance of different machine learning algorithms, we aim to develop a predictive model that can accurately identify individuals at risk of heart disease. This model can serve as a valuable tool in clinical settings, aiding in the early detection and prevention of cardiovascular diseases. Ultimately, the integration of machine learning in medical diagnostics holds the promise of improving patient outcomes and advancing the field of healthcare.

1.2 BACKGROUND

All around the world, there are numerous chronic diseases that are prevalent in both developed and developing nations. One such disease is heart disease. Heart disease encompasses a range of conditions that affect the heart, including coronary artery disease, arrhythmias, heart valve disease, congenital heart defects, and heart failure. It is one of the deadliest diseases globally, not only posing a significant health risk on its own but also contributing to other severe conditions like strokes, kidney disease, and peripheral artery disease. Heart-related diseases or Cardiovascular Diseases (CVDs) are the main reason for a huge number of death in the world over the last few decades and has emerged as the most life-threatening disease, not only in India but in the whole world. So, there is a need for a reliable, accurate, and feasible system to diagnose such diseases in time for proper treatment.

Early detection of heart disease is crucial in preventing these complications and saving lives. The earlier heart disease is diagnosed, the more effective the treatment options available to patients. Early intervention can reduce the severity of the disease, prevent heart attacks and strokes, and improve overall quality of life. Despite the critical importance of early detection, diagnosing heart disease can be challenging due to the complexity of the disease and the variety of symptoms it can present. Traditional diagnostic methods often require extensive and expensive testing, and there is always a risk of human error or variability in test results.

With the rise of machine learning (ML), artificial intelligence (AI), and neural networks, and their application in various domains, we now have powerful tools at our disposal to address the issue of heart disease diagnosis. Machine Learning algorithms and techniques have been applied to various medical datasets to automate the analysis of large and complex data. ML techniques and neural networks can help researchers uncover new insights from existing health-related data, which can aid in disease management and detection. These technologies enable the analysis of large datasets, identifying patterns and correlations that might be missed by human analysts. By applying ML and AI to healthcare data, we can develop models that predict the likelihood of heart disease, potentially improving diagnostic accuracy and enabling earlier intervention.

Many researchers, in recent times, have been using several machine learning techniques to help the health care industry and the professionals in the diagnosis of heart-related diseases. Heart is the next major organ comparing to the brain which has more priority in the Human body. It pumps the blood and supplies it to all organs of the whole body. Prediction of occurrences of heart diseases in the medical field is significant work. Data analytics is useful for prediction from more information and it helps the medical center to predict various diseases. A huge amount of patient-related data is maintained on monthly basis. The stored data can be useful for the source of predicting the occurrence of future diseases.

The ML model developed in this project aims to predict whether a patient has heart disease based on their medical data. This predictive model can serve as a valuable tool for doctors, helping them to identify patients who are at high risk and require further medical attention. By integrating this model into routine clinical practice, healthcare providers can ensure that patients receive timely and appropriate care, potentially preventing the progression of heart disease and reducing the risk of fatal complications.

In this project, we are leveraging the Kaggle Heart Disease Dataset to develop and evaluate various ML models. The goal is to identify the most effective model for predicting heart disease, which can then be implemented in clinical settings to assist doctors in diagnosing and treating patients. The process involves several steps, including data preprocessing, feature selection, model training, and evaluation. Data preprocessing involves cleaning the data, handling missing values, and normalizing the data to ensure it is suitable for analysis. Feature selection helps in identifying the most relevant features that contribute to heart disease prediction, thereby improving the model's accuracy and interpretability.

By systematically comparing the performance of different ML algorithms, we aim to develop a predictive model that can accurately identify individuals at risk of heart disease. This model can serve as a valuable tool in clinical practice, aiding in the early detection and prevention of cardiovascular diseases. Ultimately, the integration of machine learning in medical diagnostics holds the promise of improving patient outcomes and advancing the field of healthcare.

1.3 CAUSES OF HEART DISEASE:

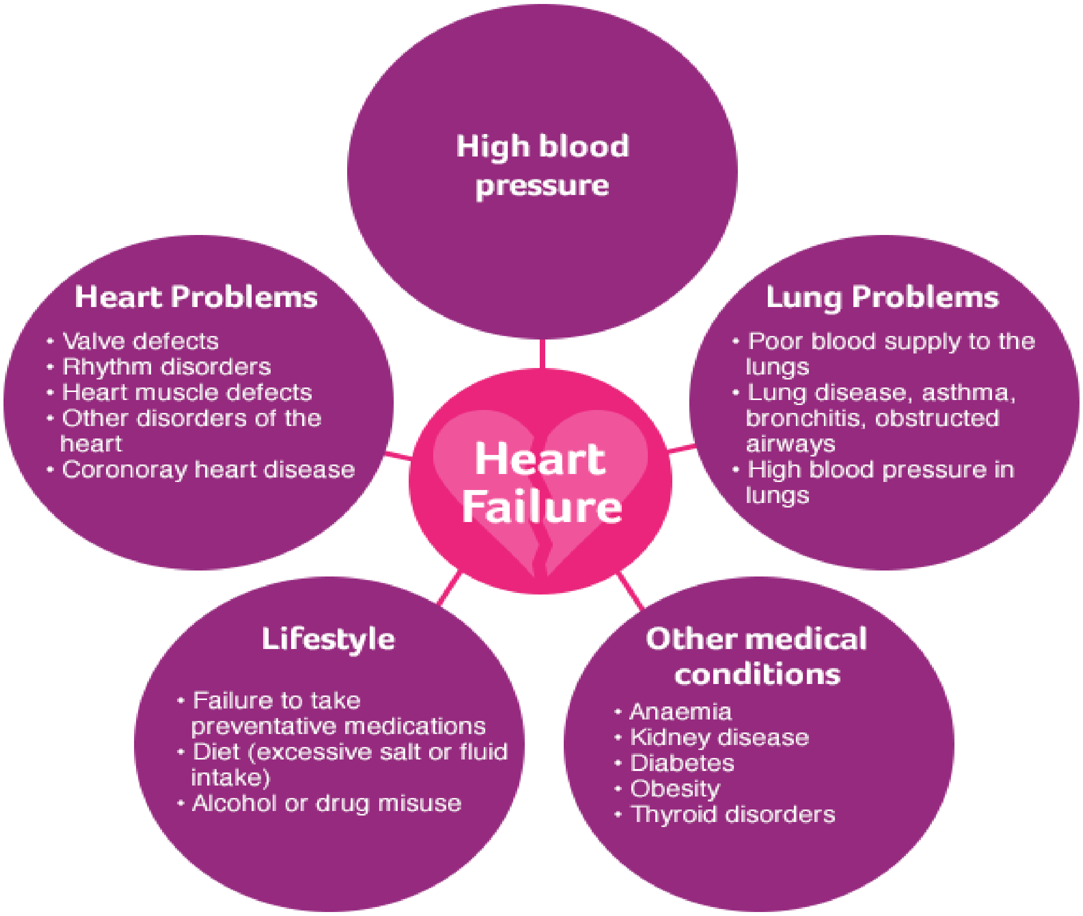

High blood pressure
Heart Problems
• Valve defects
• Rhythm disorders
• Heart muscle defects
• Other disorders of the heart
• Coronoray heart disease
Lung Problems
• Poor blood supply to the lungs
• Lung disease, asthma, bronchitis, obstructed airways
• High blood pressure in lungs
Heart Failure
Lifestyle
• Failure to take preventative medications
• Diet (excessive salt or fluid intake)
• Alcohol or drug misuse
Other medical conditions
• Anaemia
• Kidney disease
• Diabetes
• Obesity
• Thyroid disorders

Fig 1.1 Causes of Heart Disease

- **Genetic Factors**: Genetic predispositions play a significant role in heart disease. Mutations in various genes can affect heart function and increase the risk of conditions like hypertrophic cardiomyopathy and familial hypercholesterolemia.
- **Lifestyle Factors:** Poor diet, lack of physical activity, smoking, and excessive alcohol consumption are major contributors to heart disease.
- **Other Medical Conditions:** Conditions such as hypertension, diabetes, and high cholesterol levels can significantly increase the risk of developing heart disease.

1.4 TYPES OF HEART DISEASE:

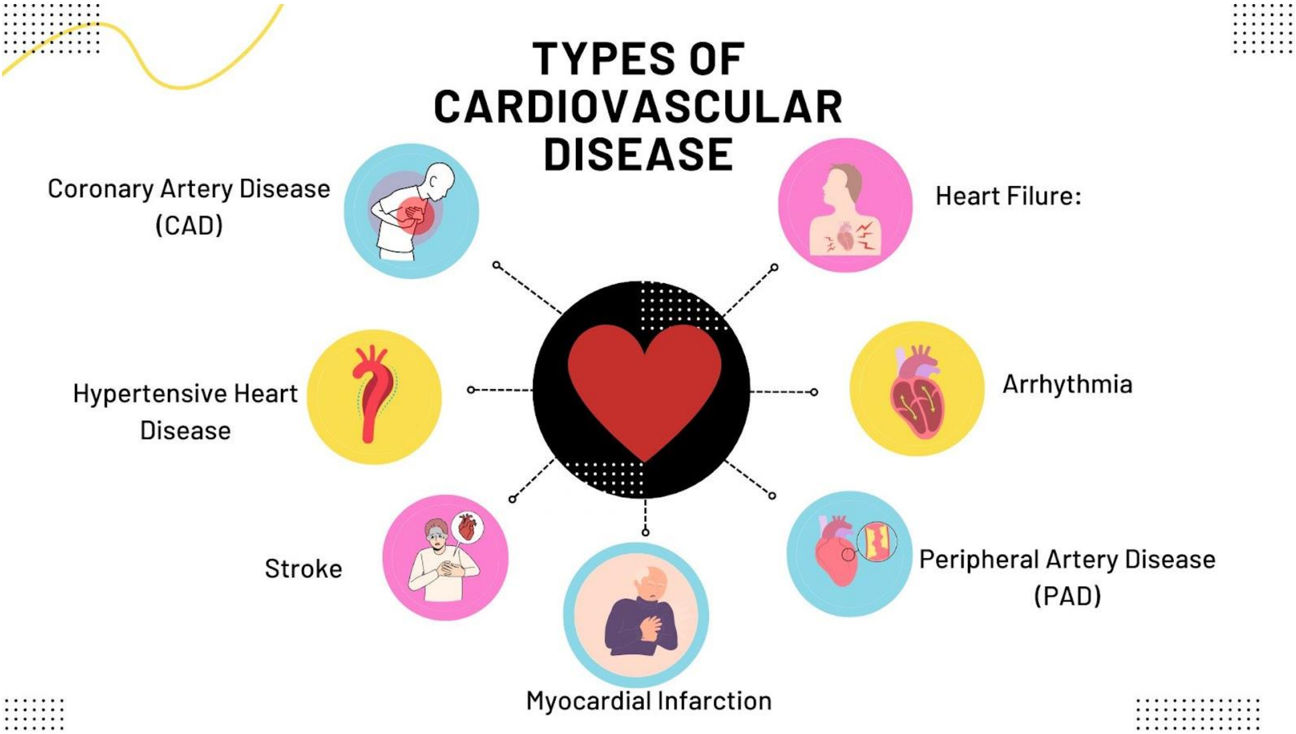

TYPES OF CARDIOVASCULAR DISEASE
Coronary Artery Disease (CAD)
Heart Filure:
Hypertensive Heart Disease
Arrhythmia
Stroke
Peripheral Artery Disease (PAD)
Myocardial Infarction

Fig 1.2 Types of Heart Disease

- **Coronary Artery Disease (CAD):** The most common type, characterized by the buildup of plaque in the coronary arteries, leading to reduced blood flow to the heart.
- **Arrhythmias:** Abnormal heart rhythms that can cause the heart to beat too fast, too slow, or irregularly.
- **Heart Failure:** A condition where the heart is unable to pump blood effectively, leading to symptoms like shortness of breath, fatigue, and fluid retention.
- **Valvular Heart Disease:** Involves damage to one or more of the heart's valves, affecting blood flow within the heart.

1.5 SYMPTOMS OF HEART DISEASE:

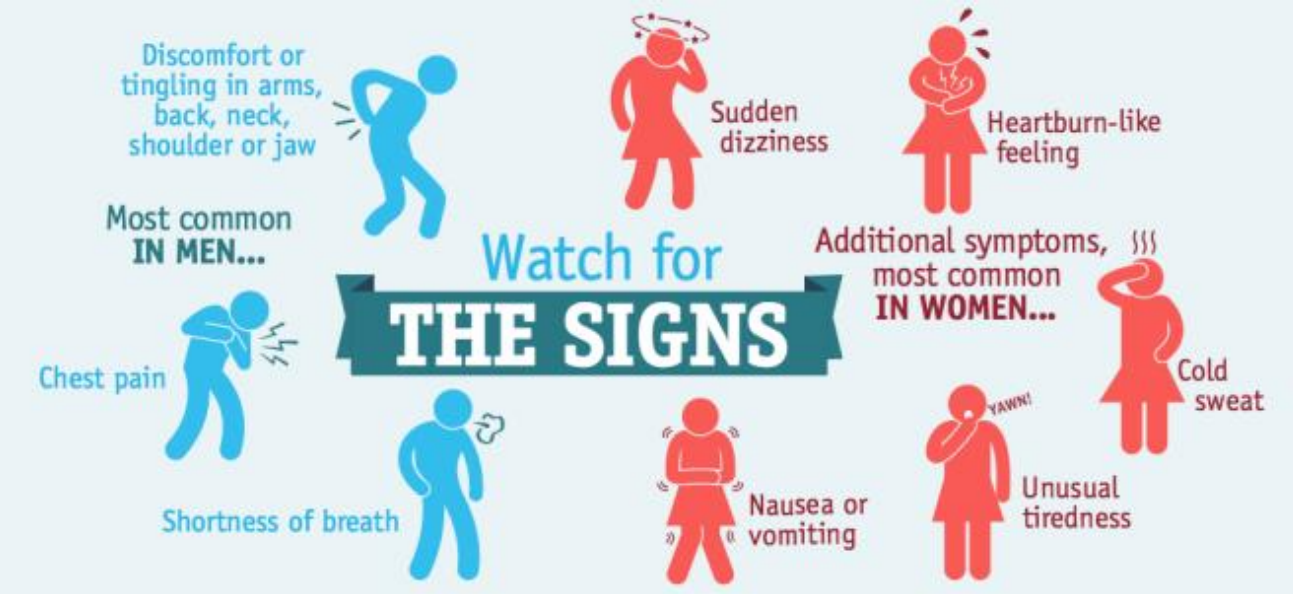


Fig 1.3 Symptoms of Heart Disease

- Chest pain or discomfort
- Shortness of breath
- Fatigue
- Palpitations or irregular heartbeats
- Swelling in the legs, ankles, or feet
- Lightheadedness or dizziness

Data mining and machine learning have been developing, reliable, and supporting tools in the medical domain in recent years. The data mining method is used to pre-process and select the relevant features from the healthcare data, and the machine learning method helps automate heart disease prediction [14]. Data mining and machine learning algorithms can help identify the hidden pattern of data using the cutting-edge method; hence, a reliable accuracy decision is possible. Data Mining is a process where several techniques are involved, including machine learning, statistics, and database system to discover a pattern from the massive amount of dataset [15]. According to Nvidia: Machine learning uses various algorithms to learn from the parsed data and make predictions.

1.6 PROBLEM STATEMENT

The major challenge in heart disease is its detection. There are instruments available which can predict heart disease but either they are expensive or are not efficient to calculate chance of heart disease in human. Early detection of cardiac diseases can decrease the mortality rate and overall complications. However, it is not possible to monitor patients every day in all cases accurately and consultation of a patient for 24 hours by a doctor is not available since it requires more sapience, time and expertise.

Since we have a good amount of data in today’s world, we can use various machine learning algorithms to analyze the data for hidden patterns. The hidden patterns can be used for health diagnosis in medicinal data.

1.7 OBJECTIVES OF THE STUDY

This research work aims to analyze the Diabetes dataset, design, and implement a Diabetes prediction and recommendation system utilizing machine learning classification techniques. The specific objectives of this project work are:

(i) To review existing literature along the area of heart disease diagnosis and prediction.
(ii) Design and develop a model using machine learning techniques.
(iii) To identify and discuss the benefits of the designed system along with effective applications.

1.8 SCOPE

The proposed system is targeted at the application of ensemble Machine learning algorithms that implement classification algorithms in machine learning on heart disease datasets of patients.

1.8.1 SIGNIFICANCE

The benefits of the proposed system are:It would help as a Decision Support System [DSS] for the hospital management, which would assist them a lot in making a timely and quality decision.Through this system, it is believed that better control measures can be taken to reduce the adverse effect of diabetes on the patient and to make recommendations that would help the patient manage his health effectively. It saves the hospital management the time and energy spent in generating patient’s disease in the existing system since most of the operation would automate under the proposed system. Heart disease has affected over 17 million people worldwide, with a significant portion of them being women. According to the World Health Organization (WHO) report, by 2025, this number is expected to rise to over 23 million. The disease has been named the leading cause of death globally, with no imminent cure in sight. With the rise of information technology and its continued advent into the medical and healthcare sector, the cases of diabetes as well as their symptoms are well documented. This paper aims at finding solutions to diagnose the disease by analyzing the patterns found in the data through classification analysis by employing Random Forest. The research hopes to propose a quicker and more efficient technique of diagnosing the disease, leading to timely treatment of the patients

CHAPTER - 2

LITERATURE REVIEW

Heart disease prediction using machine learning has been a focal area of research, with numerous studies leveraging various algorithms to enhance the accuracy and efficiency of predictions. This literature review provides a comprehensive overview of significant studies in the field, detailing the methodologies, findings, and implications of each.

Gandhi and Singh's research [1] delved into the application of Decision Tree (DT), Naïve Bayes (NB), and Neural Network (NN) algorithms on a comprehensive medical dataset to predict heart disease. They aimed to identify which features most significantly impacted the predictive accuracy and to determine the efficiency of these algorithms in processing the data. A crucial aspect of their study was the exploration of feature selection, highlighting the necessity to reduce the number of variables to streamline the model. By focusing on minimizing the number of features, they demonstrated a significant reduction in the time required for processing, which is essential in clinical settings where time efficiency is paramount.

The study concluded that DT and NN algorithms were notably effective in heart disease prediction. However, it emphasized the delicate balance required between reducing the number of features and maintaining high predictive accuracy. While a reduced feature set can significantly speed up the processing time, it risks omitting subtle yet crucial indicators of heart disease. Gandhi and Singh's work highlighted the importance of a meticulous approach to feature selection to ensure that predictive models remain both efficient and accurate. Their findings are particularly relevant for developing practical, real-world applications where timely and precise diagnosis is critical.

Thomas and Princy's Research [2] conducted an extensive study employing Neural Networks (NN), K-Nearest Neighbor (KNN), Decision Tree (DT), and Naïve Bayes (NB) algorithms for predicting heart disease. Their research focused on utilizing data mining techniques to identify risk factors associated with heart disease, aiming to enhance the predictive accuracy through the integration of multiple algorithms. By leveraging the strengths of each algorithm, they sought to provide a comprehensive analysis of risk factors and improve the overall reliability of the predictions.

Their findings suggested that integrating multiple algorithms could significantly enhance the accuracy of heart disease predictions. The research underscored the benefits of a hybrid approach, where different algorithms complement each other, leading to a more robust and comprehensive predictive model. Thomas and Princy's study highlighted the potential of combining various data mining techniques to achieve superior predictive outcomes. This integrative methodology offers valuable insights for the development of more accurate and reliable heart disease prediction tools, demonstrating the effectiveness of collaborative algorithmic approaches.

Bharti and Singh's Application of ANN and Genetic Algorithms [3] explored the combined use of Artificial Neural Networks (ANN) and Genetic Algorithms (GA) for heart disease prediction. Their research incorporated associative classification with Particle Swarm Optimization (PSO), aiming to merge association rule mining with classification for enhanced precision. This study focused on leveraging the complementary strengths of evolutionary algorithms and neural networks to optimize prediction models for heart disease, seeking to achieve high accuracy in predictions through advanced algorithmic techniques.

The research demonstrated that the integration of ANN with GA and PSO significantly improved the accuracy of heart disease predictions. By combining these methodologies, Bharti and Singh were able to harness the optimization capabilities of evolutionary algorithms along with the powerful predictive features of neural networks. Their study concluded that such hybrid approaches hold considerable promise for advancing heart disease prediction accuracy and reliability. This innovative method showcases the potential of combining diverse algorithmic strategies to develop robust and effective predictive models in medical diagnostics.

Purushottam, Saxena, and Sharma [4] proposed an innovative framework aimed at enhancing medical diagnosis through the use of electronic devices, with the added benefit of reducing healthcare costs. Their system focused on quickly discovering predictive rules based on various health parameters to assess a patient's risk level. The framework allowed for the prioritization of guidelines according to user needs, offering a systematic approach to medical diagnosis. Their study emphasized the classification precision of their system, showcasing its potential for reliably predicting heart disease risk.

The study highlighted the effectiveness of rule-based systems in medical diagnostics, providing a structured and systematic method for risk assessment. By rapidly identifying predictive rules, the framework ensured high precision in classification, thereby improving the reliability of heart disease predictions. Purushottam, Saxena, and Sharma's work concluded that such frameworks are instrumental in delivering cost-effective and accurate medical diagnostics, enhancing patient care through systematic and efficient risk assessment methodologies.

Palaniyappan and Awang [5] developed the Intelligent Heart Disease Prediction System (IHDPS) utilizing a combination of Naïve Bayes, Artificial Neural Networks, and Decision Tree algorithms. Their system aimed to present data in both tabular and graphical formats to facilitate better understanding and reduce healthcare costs by delivering effective care. The study focused on revealing hidden patterns and interactions within the data, which advanced data mining techniques helped to uncover, thereby improving the predictive accuracy of heart disease diagnosis.

The IHDPS demonstrated significant improvements in predictive accuracy by uncovering hidden relationships and patterns within the data. The study concluded that advanced data mining techniques could enhance the understanding and prediction of heart disease significantly. By presenting data in accessible formats, the system also aimed to improve clinical decision-making and reduce healthcare costs, emphasizing the practical benefits of such predictive tools in healthcare settings. Palaniyappan and Awang's work underscored the value of integrating multiple algorithms and presenting data comprehensively to enhance diagnostic accuracy and efficiency.

Sharma and Rizvi [6] applied Deep Learning, Decision Tree, Support Vector Machine (SVM), and K-Nearest Neighbor (KNN) methods for predicting heart disease. Their study addressed the challenge of noise in datasets by effectively reducing data dimensionality through comprehensive cleaning and preprocessing. The focus was on demonstrating the robustness of Neural Networks in handling complex medical data, aiming to achieve high levels of accuracy despite the presence of noisy and unstructured data.

The research highlighted the crucial role of data preprocessing in enhancing the performance and reliability of predictive models. Sharma and Rizvi concluded that deep learning algorithms, particularly Neural Networks, are highly effective in managing and predicting heart disease from complex and noisy datasets. Their study emphasized that meticulous data preparation is essential for improving the accuracy and dependability of heart disease predictions.

This work showcased the importance of thorough preprocessing in developing robust and accurate predictive models for clinical applications.

Hazra, Mandal, Gupta, Mukherjee, and Mukherjee [6] investigated cardiovascular disorders and multiple indications of heart failure, employing various algorithms for grouping and clustering patient data. Their research aimed to provide a detailed analysis of heart disease indicators through advanced clustering techniques. By exploring different methods for grouping and clustering, the study contributed to understanding how these techniques can be used to identify patterns and subgroups within patient data, which is crucial for personalized medicine.

The study revealed that advanced clustering techniques effectively identify patterns and subgroups within patient data, enhancing the understanding of heart disease indicators. This approach supports personalized medicine by enabling more tailored and precise treatment strategies based on individual patient profiles. The authors concluded that advanced clustering techniques are valuable tools for identifying crucial patterns in heart disease, aiding in the development of personalized diagnostic and treatment plans. Their work emphasized the importance of grouping and clustering in enhancing the precision and personalization of heart disease treatment.

Krishnaiah, Narsimha, and Chandra [7] conducted a comprehensive data mining analysis for heart disease prediction, examining various methods and factors to determine their impact on predictive accuracy. Their study emphasized the importance of utilizing multiple algorithms and variables to optimize heart disease prediction performance. By comparing different data mining techniques, they aimed to provide insights into the most effective methods for predicting heart disease.

Their analysis indicated that different methods and factors yield varying levels of predictive accuracy, underscoring the need for a multifaceted approach. The study concluded that a comprehensive evaluation of diverse algorithms and variables is essential for developing optimal predictive models. Krishnaiah, Narsimha, and Chandra's research provided valuable insights into the comparative effectiveness of different data mining techniques, highlighting the importance of considering multiple approaches to achieve the best predictive performance for heart disease.

Kaur and Kaur [8] addressed the issue of redundant information in heart disease datasets, emphasizing the need for effective data preprocessing and feature selection. Their research aimed to improve prediction results by eliminating unnecessary data and selecting the most relevant features for model training. The study focused on demonstrating how data preprocessing and careful feature selection can enhance the

accuracy of heart disease prediction models.
The study demonstrated that data preprocessing and feature selection significantly enhance prediction accuracy. By ensuring that models are trained on clean and relevant data, the authors concluded that these preprocessing steps are critical for the success of heart disease predictive models. Kaur and Kaur's work emphasized the importance of preparing datasets meticulously to achieve reliable and accurate predictions, highlighting the critical role of data preprocessing in data mining and predictive analytics.

Vijayashree and Iyengar [9] employed data mining strategies to analyze heart disease databases, arguing that the massive amount of knowledge generated cannot be manually translated, necessitating the use of data mining for disease prediction. Their research contrasted multiple classification algorithms to demonstrate how different strategies operate on heart disease data, aiming to provide a comprehensive understanding of their relative strengths and weaknesses.

Their comparative analysis provided a thorough understanding of how different data mining strategies operate on heart disease data. The study concluded that data mining is essential for translating large datasets into meaningful predictions, highlighting the relative strengths and weaknesses of various algorithms. Vijayashree and Iyengar's research contributed to the broader understanding of effective strategies for heart disease prediction, demonstrating the necessity of data mining techniques in managing and interpreting large-scale medical data.

Benjamin, Virani, Callaway, Chamberlain, Chang, and Cheng [10] identified seven major risk factors for heart disease: diet, smoking, obesity, diabetes, inactivity, cholesterol, and high blood pressure. Their comprehensive review provided statistical data on heart illness, including stroke and coronary artery disease, aiming to establish a foundational understanding of the factors contributing to heart disease. Their study sought to integrate these risk factors into predictive models, enhancing their accuracy and reliability by incorporating comprehensive and relevant data.

The study offered a detailed overview of the primary risk factors associated with heart disease, providing essential insights for developing predictive models. By understanding these risk factors, the research highlighted the critical elements that need to be considered in heart disease prediction. Benjamin and colleagues concluded that recognizing and incorporating these risk factors is crucial for enhancing the accuracy and effectiveness of predictive models, providing a valuable foundation for future research and model development in heart disease prediction.

Kishore, Kumar, Singh, Punia, and Hambir [11] explored the use of Recurrent Neural Networks (RNNs) for heart disease prediction, demonstrating that RNNs achieve higher precision compared to other techniques such as CNN, NB, and SVM. Their research focused on developing a system to detect silent heart failure and provide early warnings to patients, leveraging the ability of RNNs to capture temporal dependencies in medical data. This study aimed to enhance the early detection and intervention strategies for heart disease, ensuring timely treatment and management.

The study concluded that RNNs are highly effective in detecting heart disease, particularly in capturing temporal dependencies that other algorithms might overlook. By demonstrating the superior precision of RNNs, the research highlighted the potential of these networks in improving early detection and intervention strategies. Kishore and colleagues' work emphasized the importance of advanced neural network architectures in enhancing heart disease prediction accuracy, particularly for conditions requiring temporal data analysis.

Kumar, Koushik, and Deepak [12] utilized several algorithms, including Logistic Model Tree, Random Forest, Decision Tree, KNN, Naïve Bayes (NB), and SVM, to predict cardiac disease. Their study aimed to compare the performance of these algorithms on different datasets, particularly focusing on the Cleveland dataset from the UCI repository, to identify the most effective methods for heart disease prediction. The research sought to provide a comprehensive comparison of these algorithms, highlighting their strengths and limitations in various predictive tasks.

The comparative analysis revealed that the Naïve Bayes algorithm performed exceptionally well, especially for the Cleveland dataset, while the J48 algorithm also showed good performance and required less time to build. The study concluded that different algorithms offer various advantages, and the choice of algorithm should be tailored to the specific characteristics of the dataset and prediction task. Kumar, Koushik, and Deepak's work provided valuable insights into the strengths and limitations of various predictive techniques, guiding the selection of appropriate methods for heart disease prediction.

Kaur and Arora [13] compared multiple algorithms, such as KNN, SVM, Artificial Neural Network, and Naïve Bayes, for cardiac disease prediction. Their study aimed to provide a detailed comparison of the performance of these algorithms, highlighting their effectiveness under different conditions to determine the most suitable algorithm for heart disease prediction. This research sought to identify the most reliable and accurate predictive methods for clinical applications.

The study concluded that the choice of algorithm significantly impacts prediction accuracy, depending on the dataset's characteristics and the specific prediction task. By systematically comparing various algorithms, the research emphasized the importance of selecting the right method for optimal results. Kaur and Arora's work contributed to the understanding of algorithm selection in heart disease prediction, guiding future research and clinical applications with detailed insights into the performance of different algorithms.

Weng, Reps, Kai, Garibaldi, and Qureshi [14] applied four deep learning algorithms—Neural Networks (NN), logistic regression, gradient boosting machines, and random forest—to forecast heart disease. Utilizing electronic medical records from the Clinical Practice Research Datalink (CPRD), their study aimed to enhance the accuracy of heart disease prediction by leveraging comprehensive patient data. This research focused on integrating extensive medical records with advanced machine learning techniques to improve predictive accuracy.

The study demonstrated that machine learning algorithms, particularly deep learning methods, perform well in forecasting heart disease cases. By integrating comprehensive medical records, the research provided a robust approach to heart disease prediction. The authors concluded that deep learning holds significant promise for improving predictive accuracy, highlighting the benefits of combining advanced algorithms with extensive medical data. This work showcased the potential of deep learning in transforming heart disease prediction and management, providing a strong foundation for future advancements in the field.

Additional Studies and Techniques

Goyal and Agrawal [15] explored ensemble methods, combining multiple machine learning algorithms to improve prediction accuracy for heart disease. Their research focused on techniques such as bagging and boosting to aggregate the strengths of individual models, aiming to develop more reliable and accurate predictive models. The study aimed to demonstrate how ensemble methods could enhance the robustness and precision of heart disease prediction models.

The study found that ensemble methods significantly enhance model performance by leveraging the strengths of various algorithms. The authors concluded that such techniques are highly effective for heart disease prediction, offering improved accuracy and reliability.

This research emphasized the value of ensemble approaches in developing robust predictive models, highlighting their potential in advancing the field of heart disease prediction.

Patil and Kumar [16] focused on feature engineering, creating new features from existing data to improve heart disease prediction model accuracy. Their study emphasized the importance of domain knowledge in crafting features that capture essential aspects of heart disease, aiming to enhance the predictive capabilities of machine learning models. This research sought to demonstrate the impact of innovative feature engineering on improving model accuracy and reliability.

The research demonstrated that carefully engineered features significantly enhance predictive capabilities. The authors concluded that domain-specific feature engineering is crucial for developing accurate and reliable prediction models. This study highlighted the importance of feature creation in advancing heart disease prediction, showcasing how domain knowledge and innovative feature engineering can improve model performance.

Reddy and Mohan [17] applied time-series analysis to predict the progression of heart disease over time. Their research aimed to understand the temporal dynamics of heart disease, providing insights for early intervention and treatment planning. By leveraging time-series models, they sought to capture the changes in patient health over time, offering a detailed analysis of disease progression. This study aimed to enhance the understanding and prediction of heart disease by incorporating temporal data.

The study highlighted the potential of time-series models in capturing the progression of heart disease, offering valuable insights for timely interventions. The authors concluded that time-series analysis is a powerful tool for understanding and predicting heart disease progression. This research underscored the importance of temporal dynamics in heart disease prediction, emphasizing the need for models that can effectively capture and analyze time-dependent data.

Sharma et al. [18] utilized big data approaches to handle large-scale medical datasets for heart disease prediction. Their study demonstrated the scalability of machine learning algorithms in processing vast amounts of data, aiming to uncover patterns that can improve predictive accuracy. By applying big data techniques, they sought to enhance the capability of predictive models in managing and interpreting extensive datasets.

The research showed that big data techniques could effectively manage and analyze extensive medical datasets, improving prediction accuracy. The authors concluded that scalable machine learning algorithms are essential for handling large-scale data in heart disease prediction. This study emphasized the role of big data in enhancing predictive models, showcasing the potential of advanced data analytics in transforming heart disease diagnosis and management.

Thakur and Sharma [19] explored the use of explainable AI techniques to enhance the transparency of machine learning models used for heart disease prediction. Their work aimed to make the decision-making process of complex models more interpretable for clinicians, ensuring that predictions are both accurate and understandable. The study sought to bridge the gap between model complexity and clinical usability by improving the interpretability of AI-driven predictions.

The study concluded that explainable AI is crucial for ensuring that predictions are both accurate and understandable. By making models transparent, the research aimed to improve clinical trust and adoption of predictive technologies. Thakur and Sharma's work highlighted the importance of interpretability in machine learning applications for heart disease, emphasizing the need for models that provide clear and understandable decision-making processes to support clinical decision-making.

The literature on heart disease prediction using machine learning showcases a diverse range of studies employing various algorithms and techniques to enhance predictive accuracy. From traditional methods like Decision Trees and Naïve Bayes to advanced approaches involving Deep Learning and ensemble methods, researchers have explored numerous strategies to improve predictions. The integration of machine learning with electronic health records and big data analytics has further advanced the field, providing robust tools for early detection and management of heart disease.

These studies collectively emphasize the importance of feature selection, data preprocessing, and algorithm selection in developing effective prediction models. They also highlight the need for ongoing research to refine these models and adapt them to diverse patient populations and clinical settings. As machine learning continues to evolve, its application in heart disease prediction holds significant promise for improving patient outcomes and reducing the global burden of cardiovascular diseases.

The advancements in machine learning and data analytics have provided unprecedented opportunities to transform heart disease prediction and management. Future research and development in this field will likely focus on enhancing model accuracy, integrating diverse data sources.

CHAPTER - 3

PRESENT WORK AND SIMULATION

3.1 DATA COLLECTION

The dataset used in this study is the publicly available heart disease dataset from the UCI Machine Learning Repository. It consists of 303 records and 14 attributes, including age, sex, chest pain type, resting blood pressure, serum cholesterol, fasting blood sugar, resting electrocardiographic results, maximum heart rate achieved, exercise-induced angina, ST depression, the slope of the peak exercise ST segment, number of major vessels, thalassemia, and the presence of heart disease.

3.2 DATA PREPROCESSING

Data preprocessing is a crucial step to ensure the dataset is clean and ready for analysis. This involves:

- Handling missing values: Checking for and addressing any missing or inconsistent data.
- Feature scaling: Standardizing numerical features to have a mean of 0 and a standard deviation of 1.
- Encoding categorical variables: Converting categorical features into numerical values.

3.3 EXPLORATORY DATA ANALYSIS (EDA)

EDA is performed to understand the underlying patterns and distributions within the dataset. This includes:

- Visualizing the distribution of each feature.
- Analyzing the correlation between features and the target variable.
- Identifying any outliers or anomalies.

3.4 FEATURE SELECTION

Feature selection helps in identifying the most relevant features that contribute to the prediction of heart disease. Techniques such as SelectKBest with ANOVA F-value are used to select important features. Features with low significance are removed to improve model performance and reduce overfitting.

3.5 MODEL TRAINING

The preprocessed dataset is split into training and testing sets, with 70% used for training and 30% for testing. Various machine learning algorithms are then trained on the training dataset:

- Random Forest
- Decision Tree
- Logistic Regression
- Support Vector Machine (SVM)
- K-Nearest Neighbors (KNN)

3.6 MODEL EVALUATION

The trained models are evaluated on the testing dataset using metrics such as accuracy, precision, recall, F1-score, and confusion matrix. Cross-validation is also performed to assess the models' robustness and generalization ability.

3.7 HYPERPARAMETER TUNING

Hyperparameter tuning is performed using techniques such as GridSearchCV and RandomizedSearchCV to find the optimal parameters for each model, further improving their performance.

3.8 ALGORITHM OVERVIEW

This project involves the use of several machine learning algorithms, including:

- **Random Forest**: Random Forest is an ensemble learning method primarily used for classification and regression tasks. It operates by constructing a multitude of decision trees during training and outputs the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. The fundamental concept behind Random Forest is to combine the predictions of several base estimators to improve generalization. Each tree is built from a random subset of the training data using a method called bootstrap aggregating, or bagging. Additionally, during the split, a random subset of features is considered for each decision node, enhancing the diversity among the trees.

  One of the main advantages of Random Forest is its ability to handle a large number of input variables without overfitting, thanks to the law of large numbers, which states that the average of many weak learners can form a strong learner. It also provides estimates of feature importance, helping to identify which variables contribute most to the prediction. However, a downside is that Random Forests can be computationally intensive due to the creation of many trees and may require significant memory for large datasets. Despite this, its robustness and versatility make it a popular choice in many practical applications, from financial forecasting to medical diagnosis and image classification.

- **Decision Tree**: A Decision Tree is a flowchart-like structure used for decision-making and predictive modeling. It splits the dataset into subsets based on the value of input features, creating branches until reaching an outcome, or leaf node. Each internal node represents a test on a feature, each branch represents the outcome of the test, and each leaf node represents a class label (for classification) or a continuous value (for regression). The paths from the root to the leaf represent classification rules.

  Decision Trees are easy to understand and interpret, especially for small trees. They can handle both numerical and categorical data and require little data preprocessing. Moreover, they are non-parametric, meaning they do not assume any underlying distribution of the data. A significant advantage is their ability to model non-linear relationships.

However, Decision Trees have several limitations. They can easily overfit the training data, particularly if they are deep. This can be mitigated by pruning, which removes branches that have little importance. Additionally, Decision Trees can be unstable because small variations in the data might result in different splits and hence a completely different tree. Despite these limitations, Decision Trees are a foundational method and are the building blocks for more complex algorithms like Random Forests and Gradient Boosted Trees.

- **Logistic Regression**: Logistic Regression is a statistical method for modelling the probability of a binary outcome based on one or more predictor variables. It is a type of regression analysis where the dependent variable is binary (0 or 1, true or false, success or failure). The logistic model estimates the probability that a given instance belongs to a particular class using the logistic function, which constrains the output to lie between 0 and 1. Logistic Regression is widely used in binary classification problems, such as spam detection, disease diagnosis, and credit scoring.

  One of the main strengths of Logistic Regression is its simplicity and interpretability. The coefficients of the model can be interpreted as the change in the log odds of the outcome for a one-unit change in the predictor variable. However, it assumes a linear relationship between the log odds of the outcome and the predictor variables, which might not always hold true. Furthermore, it is sensitive to outliers and irrelevant features, so preprocessing steps like feature scaling and selection are often necessary. Despite these limitations, Logistic Regression remains a popular and effective technique for binary classification problems.

- **Support Vector Machine (SVM)**: Support Vector Machine (SVM) is a powerful supervised learning algorithm used for classification and regression tasks. It works by finding the hyperplane that best separates the data into different classes. The optimal hyperplane is the one that maximizes the margin between the closest points of the classes, known as support vectors. SVM can efficiently handle both linear and non-linear classification problems using a kernel trick, which transforms the input data into a higher-dimensional space where a linear separator can be found.

The primary advantage of SVM is its effectiveness in high-dimensional spaces, making it suitable for applications with a large number of features, such as text classification and image recognition. SVM is also robust to overfitting, especially in high-dimensional space, because the complexity of the decision rule is characterized by the number of support vectors rather than the dimensionality of the data.

However, SVMs can be computationally intensive, especially with large datasets. The choice of the kernel and regularization parameters significantly affects the model's performance and requires careful tuning. Despite these challenges, SVMs are valued for their theoretical foundation, robust performance, and flexibility in various domains, from bioinformatics to finance and beyond.

- **K-Nearest Neighbors (KNN)**: K-Nearest Neighbors (KNN) is a simple, instance-based learning algorithm used for classification and regression. The principle behind KNN is straightforward: given a new sample, the algorithm finds the k training samples closest in distance to the new sample and assigns the most common class (for classification) or the average value (for regression) of these neighbors to the new sample. The distance is usually measured using metrics like Euclidean distance, Manhattan distance, or Minkowski distance.

  KNN is non-parametric and makes no assumptions about the underlying data distribution, making it versatile for various types of data. It is also easy to implement and understand. However, KNN has several drawbacks. It can be computationally expensive, especially with large datasets, because it requires calculating the distance between the new sample and all training samples. Additionally, KNN is sensitive to the choice of k and the distance metric, and it can be affected by the curse of dimensionality, where the distance measures become less meaningful in high-dimensional spaces.

  Despite these challenges, KNN is effective in scenarios where the decision boundary is very irregular and for applications requiring real-time predictions, such as recommendation systems and anomaly detection. Proper preprocessing, such as feature scaling and dimensionality reduction, can significantly improve KNN's performance and applicability.

3.9 DATA PREPROCESSING

Data preprocessing involves cleaning the dataset, handling missing values, and standardizing features to ensure the machine learning models perform optimally. This step ensures that the data is in a suitable format for model training.

3.10 FEATURE SELECTION

Feature selection is crucial in identifying the most relevant features that contribute to the prediction of heart disease. Techniques such as SelectKBest with ANOVA F-value are used to select important features.

3.11 MODEL TRAINING AND EVALUATION

Models are trained using the training dataset and evaluated on the testing dataset. Various performance metrics, including accuracy, precision, recall, and F1-score, are used to assess the models.

3.12 CROSS-VALIDATION

Cross-validation is performed to ensure the models' robustness and to check for overfitting. The models' performance is averaged over multiple folds to get a more accurate estimate of their generalization ability.

3.13 HYPERPARAMETER TUNING

Hyperparameter tuning is essential for optimizing the performance of machine learning models. GridSearchCV and RandomizedSearchCV are used to find the best combination of hyperparameters for each model. This process involves specifying a range of values for each hyperparameter and searching through these combinations to identify the best-performing set.

3.14 MODEL COMPARISON

The performance of each model is compared based on the evaluation metrics. The model with the highest accuracy and best overall performance is selected as the final model for heart disease prediction. In our case it is Random forest with highest accuracy.

CHAPTER - 4

CODE IMPLEMENTATION

1.
   ```
   !pip install cufflinks
   ```

2.
   ```
   !pip install mlxtend
   ```

3.
   ```
   import pandas as pd
   import numpy as np
   import matplotlib.pyplot as plt
   import seaborn as sns
   import cufflinks as cf
   %matplotlib inline
   from sklearn.metrics import
   classification_report,confusion_matrix,accuracy_score
   from sklearn.preprocessing import StandardScaler
   from sklearn.model_selection import  RandomizedSearchCV, train_test_split ,
   GridSearchCV
   from sklearn.feature_selection import SelectKBest, f_classif
   from sklearn.ensemble import RandomForestClassifier
   from sklearn.neighbors import KNeighborsClassifier
   from sklearn.svm import SVC
   from mlxtend.plotting import plot_confusion_matrix
   #To turn off warning messages.
   import warnings
   warnings.filterwarnings('ignore')
   ```

4.
   ```
   # Importing the dataset
   data = pd.read_csv("heart.csv")
   data.head()
   ```

```
    age  sex  cp  trestbps  chol  fbs  restecg  thalach  exang  oldpeak  slope  ca  thal  target
052   1    0   125       212   0    1        168      0      1.0      2      2   3     0
153   1    0   140       203   1    0        155      1      3.1      0      0   3     0
270   1    0   145       174   0    1        125      1      2.6      0      0   3     0
361   1    0   148       203   0    1        161      0      0.0      2      1   3     0
462   0    0   138       294   1    1        106      0      1.9      1      3   2     0
```

Fig 4.1 Dataset

5.
   ```
   plt.hist(data.age, rwidth=0.98)
   plt.title("AGE DATA",fontsize=15)
   plt.show()
   ```

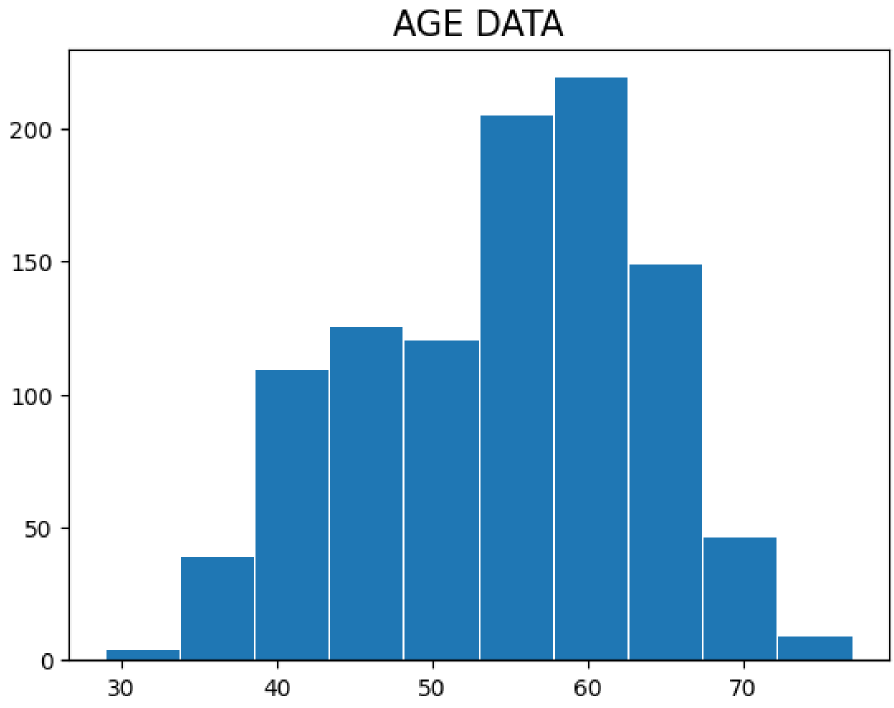


Fig 4.2 Age Data

6.
   ```
   plt.hist(data.cp, rwidth=0.98)
   plt.title("Chest pain DATA",fontsize=15)
   plt.show()
   ```

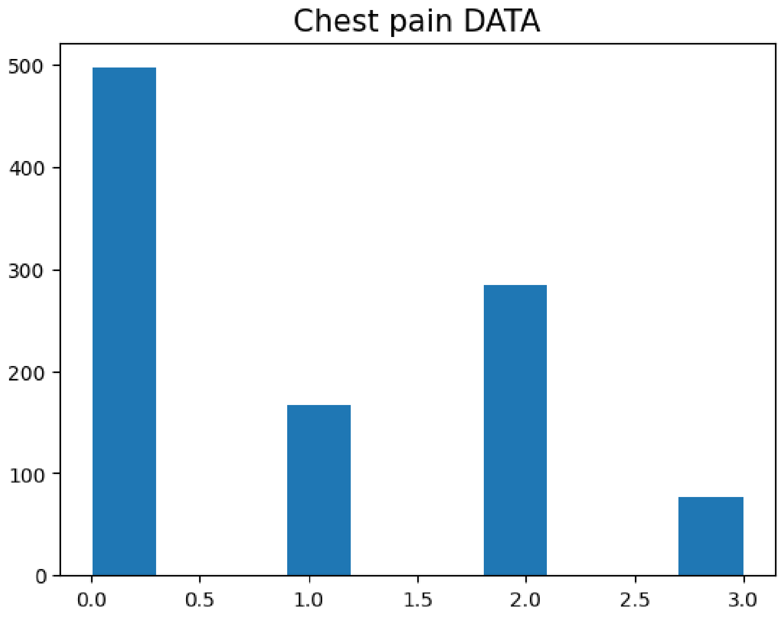


Fig 4.3 Chest Pain Data

7.

```
plt.hist(data.trestbps, rwidth=0.98)
plt.title("Resting blood pressure DATA",fontsize=15)
plt.show()
```

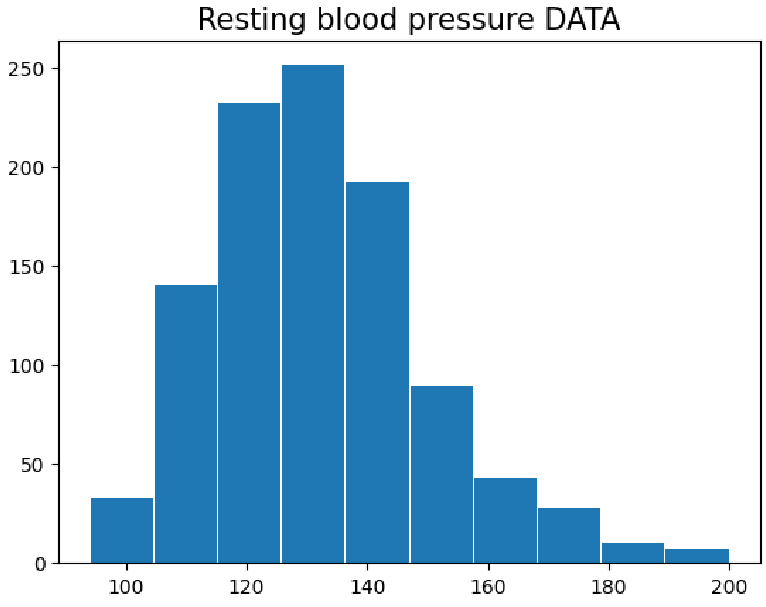
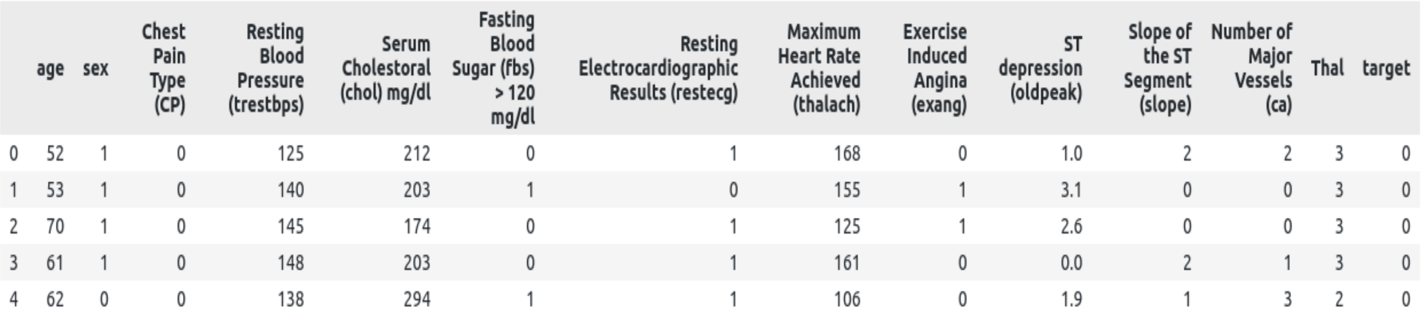


Fig 4.4 Resting blood pressure DATA

8.

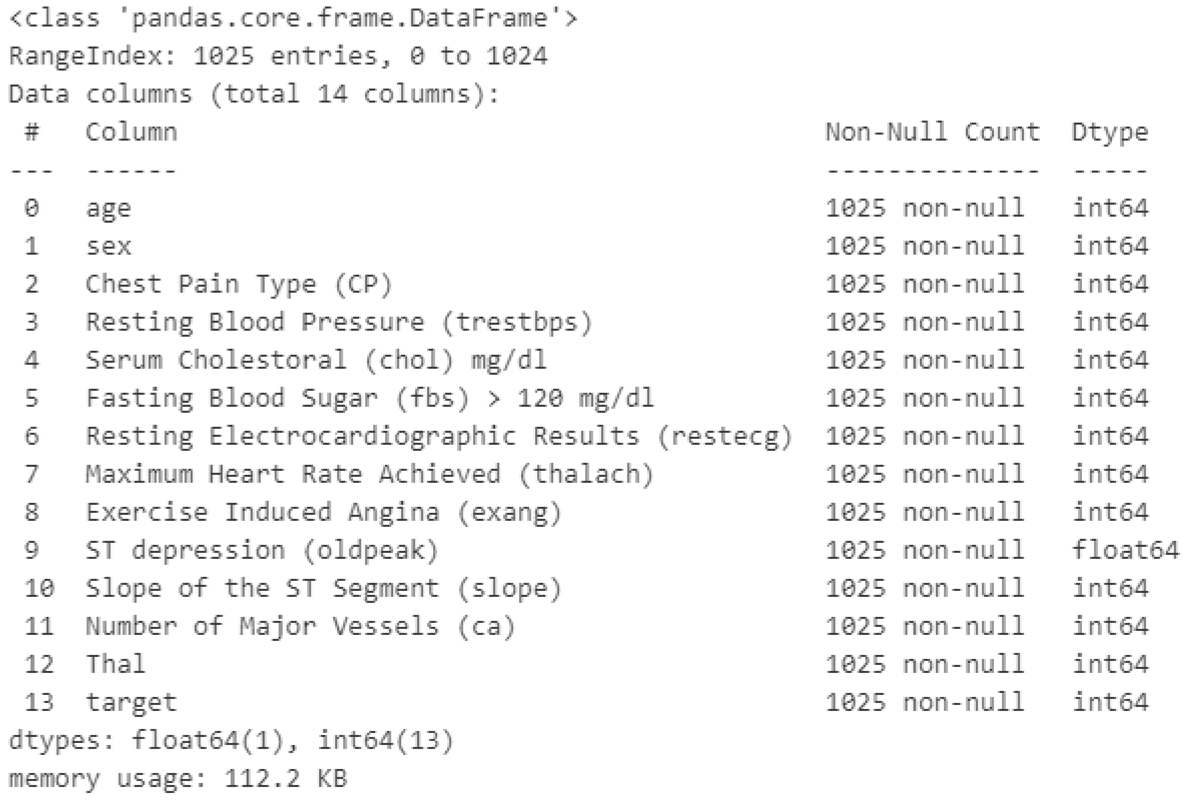
```
n_cols = {'cp':'Chest Pain Type (CP)',
          'trestbps':'Resting Blood Pressure (trestbps)',
          'chol':'Serum Cholestoral (chol) mg/dl',
          'fbs':  'Fasting Blood Sugar (fbs) > 120 mg/dl',
          'restecg': 'Resting Electrocardiographic Results (restecg)',
          'thalach' : 'Maximum Heart Rate Achieved (thalach)',
          'exang': 'Exercise Induced Angina (exang)',
          'oldpeak' : 'ST depression (oldpeak)' ,
          'slope' : 'Slope of the ST Segment (slope)',
          'ca' : 'Number of Major Vessels (ca)',
          'thal' : 'Thal'}

data.rename(columns=n_cols ,inplace=True)
data.head()
```



	age	sex	Chest Pain Type (CP)	Resting Blood Pressure (trestbps)	Serum Cholestoral (chol) mg/dl	Fasting Blood Sugar (fbs) > 120 mg/dl	Resting Electrocardiographic Results (restecg)	Maximum Heart Rate Achieved (thalach)	Exercise Induced Angina (exang)	ST depression (oldpeak)	Slope of the ST Segment (slope)	Number of Major Vessels (ca)	Thal	target
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	0
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	0
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	0

Fig 4.5 Dataset

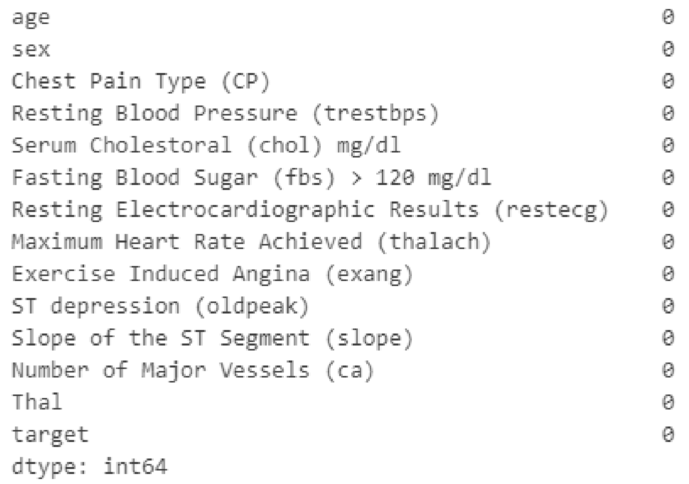
9. data.info()



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1025 entries, 0 to 1024
Data columns (total 14 columns):
 #   Column                                       Non-Null Count  Dtype  
---  ------                                       --------------  -----  
 0   age                                          1025 non-null   int64  
 1   sex                                          1025 non-null   int64  
 2   Chest Pain Type (CP)                         1025 non-null   int64  
 3   Resting Blood Pressure (trestbps)            1025 non-null   int64  
 4   Serum Cholestoral (chol) mg/dl               1025 non-null   int64  
 5   Fasting Blood Sugar (fbs) > 120 mg/dl        1025 non-null   int64  
 6   Resting Electrocardiographic Results (restecg) 1025 non-null   int64  
 7   Maximum Heart Rate Achieved (thalach)        1025 non-null   int64  
 8   Exercise Induced Angina (exang)              1025 non-null   int64  
 9   ST depression (oldpeak)                      1025 non-null   float64
 10  Slope of the ST Segment (slope)              1025 non-null   int64  
 11  Number of Major Vessels (ca)                 1025 non-null   int64  
 12  Thal                                         1025 non-null   int64  
 13  target                                       1025 non-null   int64  
dtypes: float64(1), int64(13)
memory usage: 112.2 KB
```

Fig 4.6 Data Info

10.data.isnull().sum()



```
age                                               0
sex                                               0
Chest Pain Type (CP)                              0
Resting Blood Pressure (trestbps)                 0
Serum Cholestoral (chol) mg/dl                    0
Fasting Blood Sugar (fbs) > 120 mg/dl             0
Resting Electrocardiographic Results (restecg)    0
Maximum Heart Rate Achieved (thalach)             0
Exercise Induced Angina (exang)                   0
ST depression (oldpeak)                           0
Slope of the ST Segment (slope)                   0
Number of Major Vessels (ca)                      0
Thal                                              0
target                                            0
dtype: int64
```

Fig 4.7 Null Data

```
11.#Colerration check.
    plt.style.use('fivethirtyeight')
    plt.figure(figsize=(12, 12))
    sns.heatmap(data.corr(),annot=True,fmt = ".2f",cmap='viridis')
    plt.show()
```

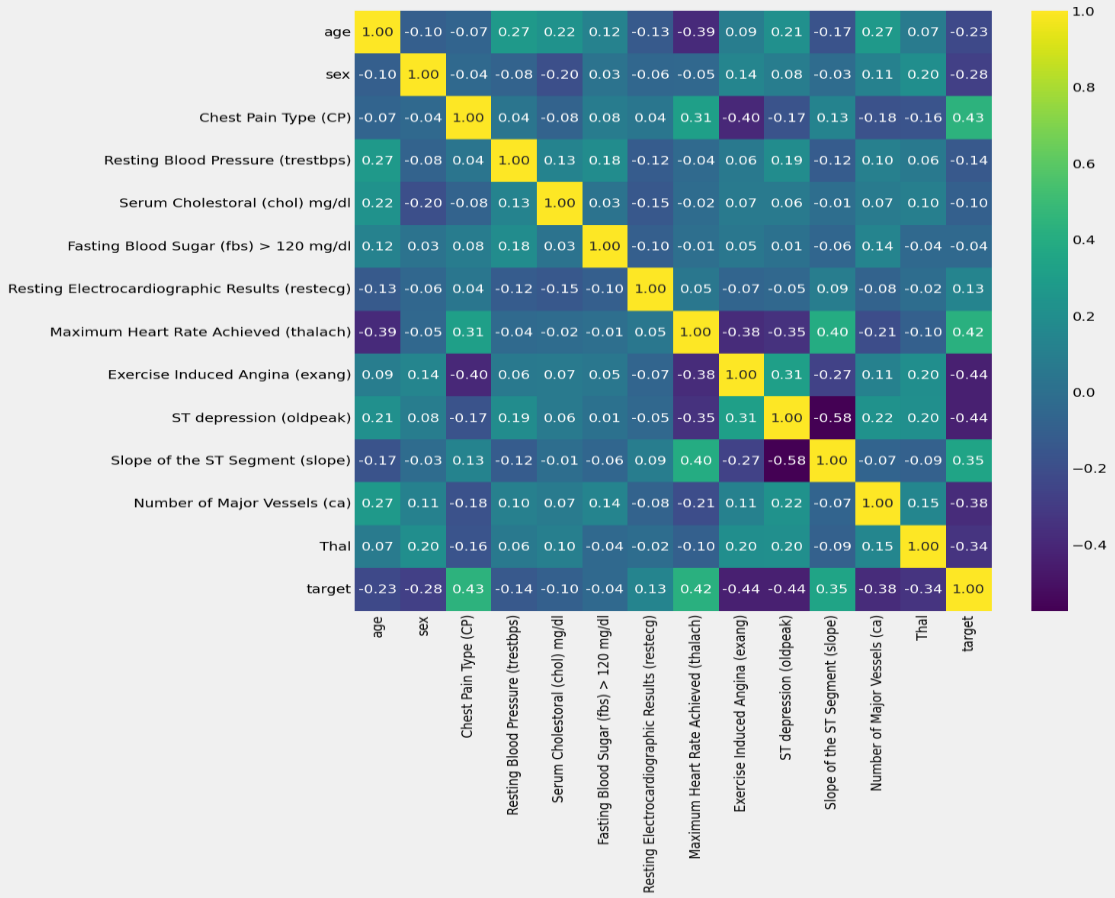


Fig 4.8 Colerration Map

12.#Clculate age

```
minAge=min(data.age)
maxAge=max(data.age)
meanAge=data.age.mean()
print("min =",minAge ,"max =", maxAge , "mean =",meanAge)
```

```
min = 29 max = 77 mean = 54.43414634146342
```

13.#Prepare ages for a pie chart

```
Young = data[(data.age>=29)&(data.age<40)]
Middle = data[(data.age>=40)&(data.age<55)]
Old = data[(data.age>55)]
plt.style.use('fivethirtyeight')
colors = ['b','r','m']
explode = [0.1,0.1,0.1]
plt.figure(figsize=(5,5))
sns.set_context('notebook',font_scale = 1.2)
plt.pie([len(Young),len(Middle),len(Old)],labels=['young ages','middle ages','old ages'],explode=explode,shadow=True,colors=colors, autopct='%1.1f%%')
plt.show()
```

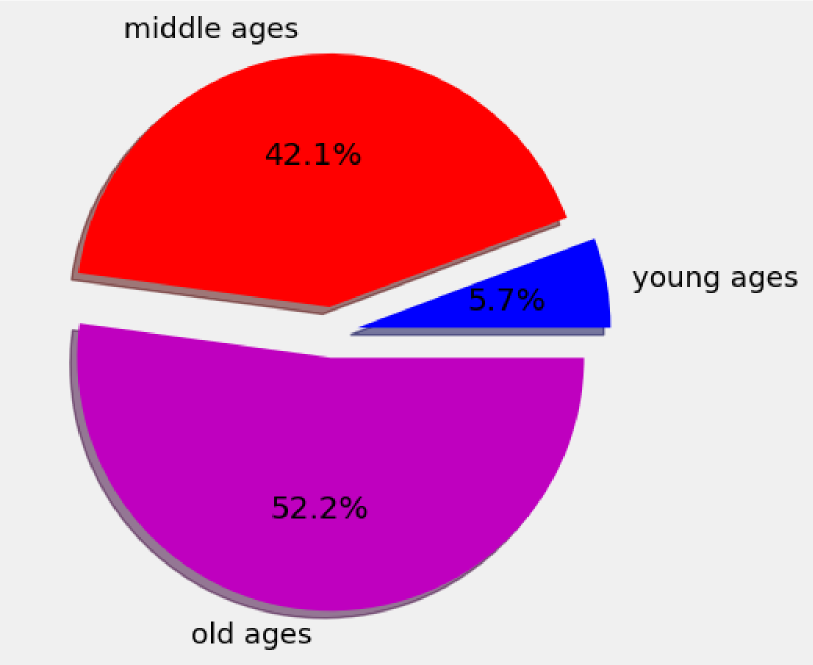


Fig 4.9 Ages Pie Chart

```
14.#There is no outliears
    # Feature selection using SelectKBest with ANOVA F-value
    X = data.drop(["target"], axis = 1)
    y = data["target"]
    selector = SelectKBest(f_classif, k=13)
    X_selected = selector.fit_transform(X, y)

    selected_features = X.columns[selector.get_support()]
    feature_scores = selector.scores_[selector.get_support()]

    # Create a DataFrame to store the feature names and their scores
    feature_scores_data_set = pd.DataFrame({'Features': selected_features, 'Scores': feature_scores})

    # Sort the DataFrame by score in descending order
    feature_scores_data_set = feature_scores_data_set.sort_values(by='Scores', ascending=False)

    # Plot the feature scores
    plt.figure(figsize=(10, 6))
    sns.barplot(x='Scores', y='Features', data=feature_scores_data_set, palette='viridis')
    plt.title('Feature Scores')
    plt.xlabel('Scores')
    plt.ylabel('Features')
    plt.show()
```

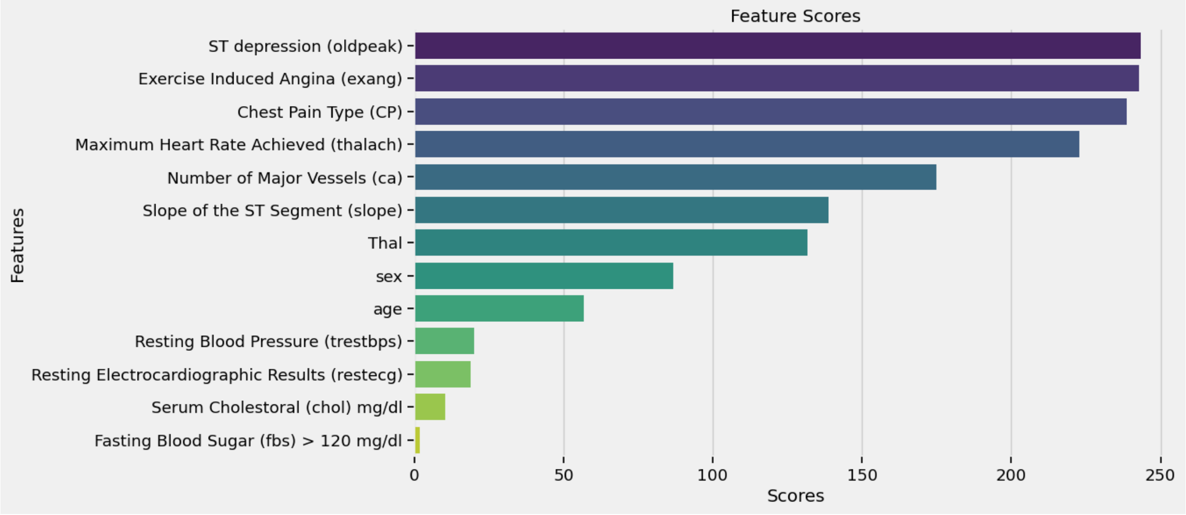
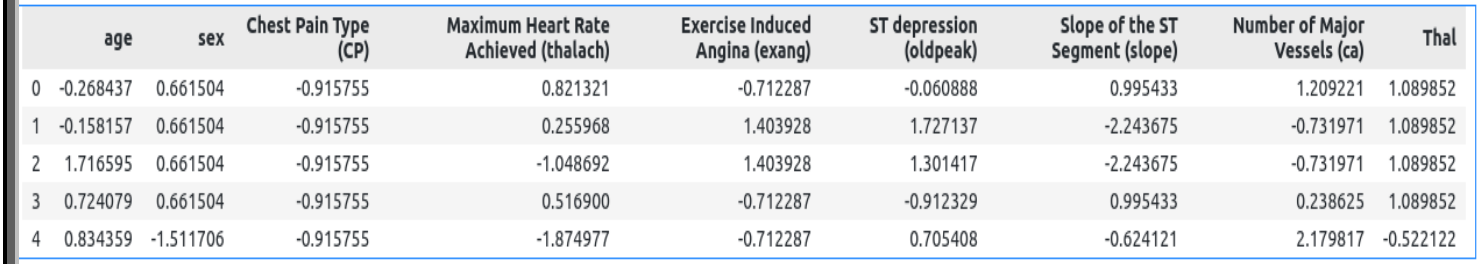


Fig 4.10 Features Score Plot

```
15. X = X.drop(["Fasting Blood Sugar (fbs) > 120 mg/dl","Serum Cholestoral (chol)
    mg/dl",
     "Resting Electrocardiographic Results (restecg)","Resting Blood Pressure
    (trestbps)"], axis = 1)

16.#Scale all values for good Accuracy
    sc = StandardScaler()
    col = ['age',
           'sex',
           'Chest Pain Type (CP)',
           'Thal',
           'Exercise Induced Angina (exang)',
           'Slope of the ST Segment (slope)',
           'Number of Major Vessels (ca)',
           'Maximum Heart Rate Achieved (thalach)',
           'ST depression (oldpeak)']
    X[col] = sc.fit_transform(X[col])
    X.head()
```



	age	sex	Chest Pain Type (CP)	Maximum Heart Rate Achieved (thalach)	Exercise Induced Angina (exang)	ST depression (oldpeak)	Slope of the ST Segment (slope)	Number of Major Vessels (ca)	Thal
0	-0.268437	0.661504	-0.915755	0.821321	-0.712287	-0.060888	0.995433	1.209221	1.089852
1	-0.158157	0.661504	-0.915755	0.255968	1.403928	1.727137	-2.243675	-0.731971	1.089852
2	1.716595	0.661504	-0.915755	-1.048692	1.403928	1.301417	-2.243675	-0.731971	1.089852
3	0.724079	0.661504	-0.915755	0.516900	-0.712287	-0.912329	0.995433	0.238625	1.089852
4	0.834359	-1.511706	-0.915755	-1.874977	-0.712287	0.705408	-0.624121	2.179817	-0.522122

Fig 4.11 Accuracy Chart

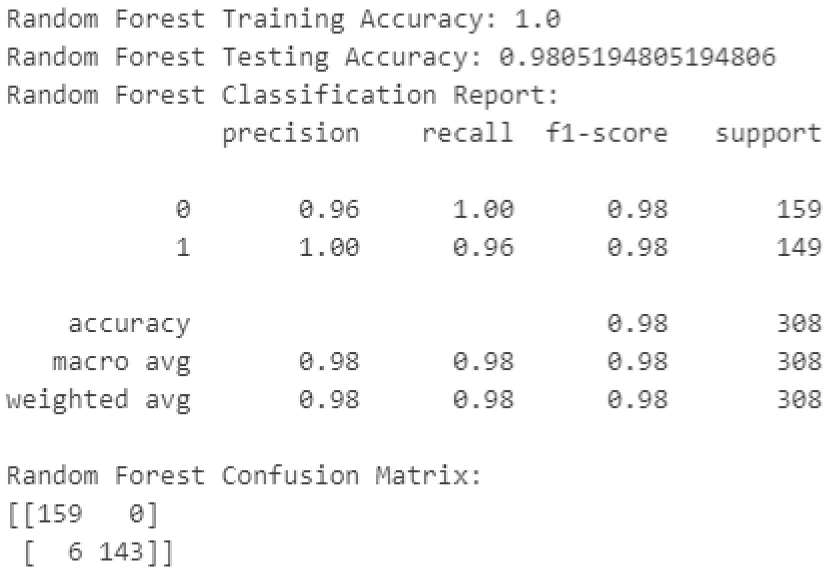
17.
```
#Splitting the data into the training and testing set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3,
random_state = 42)
```

18.
```
import pandas as pd
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix
```

- **Random Forest**

```
# Initialize the Random Forest Classifier
rf_model = RandomForestClassifier(n_estimators=100, random_state=42)

# Train the model
rf_model.fit(X_train, y_train)

# Predict on the training data
train_preds_rf = rf_model.predict(X_train)
train_accuracy_rf = accuracy_score(y_train, train_preds_rf)
print(f"Random Forest Training Accuracy: {train_accuracy_rf}")

# Predict on the testing data
test_preds_rf = rf_model.predict(X_test)
test_accuracy_rf = accuracy_score(y_test, test_preds_rf)
print(f"Random Forest Testing Accuracy: {test_accuracy_rf}")
```

```
# Additional evaluation metrics
print("Random Forest Classification Report:")
print(classification_report(y_test, test_preds_rf))
print("Random Forest Confusion Matrix:")
print(confusion_matrix(y_test, test_preds_rf))
```



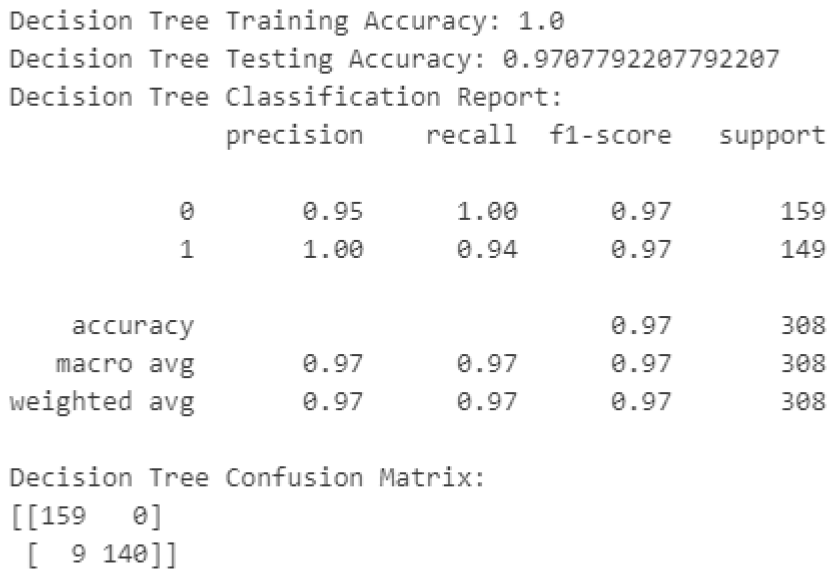
```
Random Forest Training Accuracy: 1.0
Random Forest Testing Accuracy: 0.9805194805194806
Random Forest Classification Report:
              precision    recall  f1-score   support

           0       0.96      1.00      0.98       159
           1       1.00      0.96      0.98       149

    accuracy                           0.98       308
   macro avg       0.98      0.98      0.98       308
weighted avg       0.98      0.98      0.98       308

Random Forest Confusion Matrix:
[[159   0]
 [  6 143]]
```

Fig 4.12 Random Forest

- **Decision tree**

```
from sklearn.tree import DecisionTreeClassifier
dt_model = DecisionTreeClassifier(random_state=42)

# Train the model
dt_model.fit(X_train, y_train)

# Predict on the training data
train_preds_dt = dt_model.predict(X_train)
train_accuracy_dt = accuracy_score(y_train, train_preds_dt)
print(f"Decision Tree Training Accuracy: {train_accuracy_dt}")

# Predict on the testing data
test_preds_dt = dt_model.predict(X_test)
test_accuracy_dt = accuracy_score(y_test, test_preds_dt)
print(f"Decision Tree Testing Accuracy: {test_accuracy_dt}")
```

```
# Additional evaluation metrics
print("Decision Tree Classification Report:")
print(classification_report(y_test, test_preds_dt))
print("Decision Tree Confusion Matrix:")
print(confusion_matrix(y_test, test_preds_dt))
```



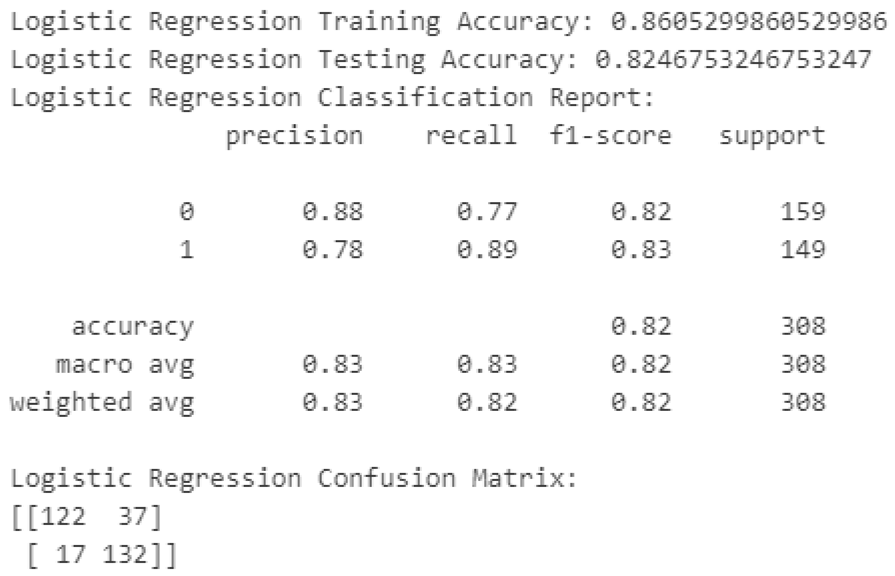
```
Decision Tree Training Accuracy: 1.0
Decision Tree Testing Accuracy: 0.9707792207792207
Decision Tree Classification Report:
              precision    recall  f1-score   support

           0       0.95      1.00      0.97       159
           1       1.00      0.94      0.97       149

    accuracy                           0.97       308
   macro avg       0.97      0.97      0.97       308
weighted avg       0.97      0.97      0.97       308

Decision Tree Confusion Matrix:
[[159   0]
 [  9 140]]
```

Fig 4.13 Decision Tree

- **Logistic Regression**

```
from sklearn.linear_model import LogisticRegression

# Initialize the Logistic Regression Classifier
lr_model = LogisticRegression(max_iter=1000, random_state=42)

# Train the model
lr_model.fit(X_train, y_train)

# Predict on the training data
train_preds_lr = lr_model.predict(X_train)
train_accuracy_lr = accuracy_score(y_train, train_preds_lr)
print(f"Logistic Regression Training Accuracy: {train_accuracy_lr}")
```

```
# Predict on the testing data
test_preds_lr = lr_model.predict(X_test)
test_accuracy_lr = accuracy_score(y_test, test_preds_lr)
print(f"Logistic Regression Testing Accuracy: {test_accuracy_lr}")
# Additional evaluation metrics
print("Logistic Regression Classification Report:")
print(classification_report(y_test, test_preds_lr))
print("Logistic Regression Confusion Matrix:")
print(confusion_matrix(y_test, test_preds_lr))
```



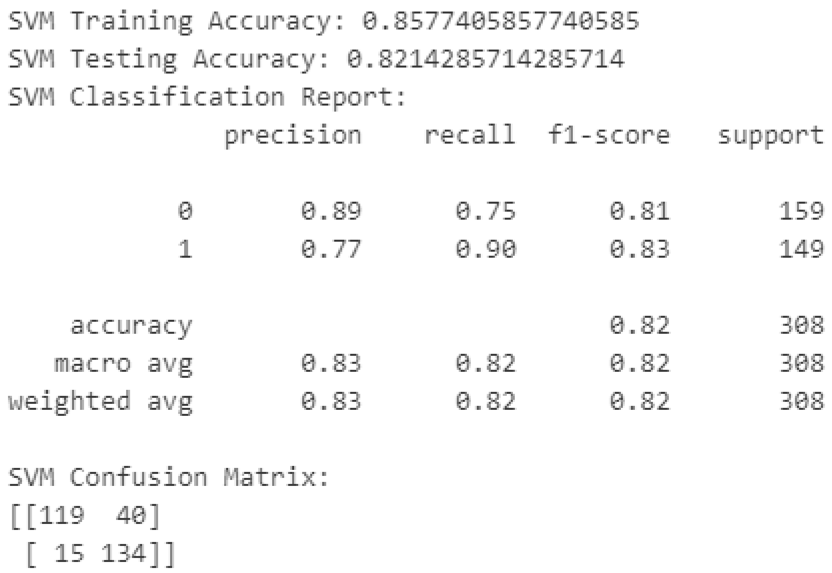
```
Logistic Regression Training Accuracy: 0.8605299860529986
Logistic Regression Testing Accuracy: 0.8246753246753247
Logistic Regression Classification Report:
              precision    recall  f1-score   support

           0       0.88      0.77      0.82       159
           1       0.78      0.89      0.83       149

    accuracy                           0.82       308
   macro avg       0.83      0.83      0.82       308
weighted avg       0.83      0.82      0.82       308

Logistic Regression Confusion Matrix:
[[122  37]
 [ 17 132]]
```

Fig 4.14 Logistic Regression

- **SVM**

```
# Initialize the SVM Classifier
svm_model = SVC(kernel='linear', random_state=42)

# Train the model
svm_model.fit(X_train, y_train)

# Predict on the training data
train_preds_svm = svm_model.predict(X_train)
train_accuracy_svm = accuracy_score(y_train, train_preds_svm)
print(f"SVM Training Accuracy: {train_accuracy_svm}")
```

```
# Predict on the testing data
test_preds_svm = svm_model.predict(X_test)
test_accuracy_svm = accuracy_score(y_test, test_preds_svm)
print(f"SVM Testing Accuracy: {test_accuracy_svm}")

# Additional evaluation metrics
print("SVM Classification Report:")
print(classification_report(y_test, test_preds_svm))
print("SVM Confusion Matrix:")
print(confusion_matrix(y_test, test_preds_svm))
```

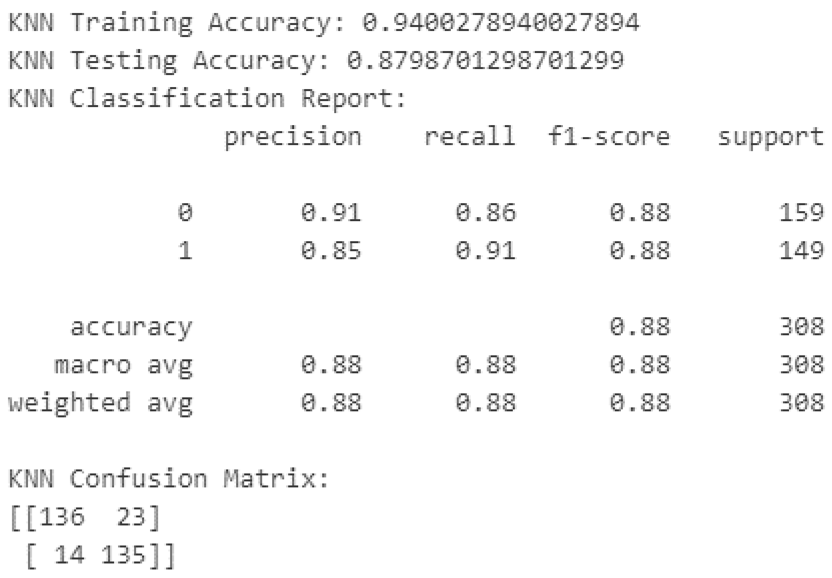


```
SVM Training Accuracy: 0.8577405857740585
SVM Testing Accuracy: 0.8214285714285714
SVM Classification Report:
              precision    recall  f1-score   support

           0       0.89      0.75      0.81       159
           1       0.77      0.90      0.83       149

    accuracy                           0.82       308
   macro avg       0.83      0.82      0.82       308
weighted avg       0.83      0.82      0.82       308

SVM Confusion Matrix:
[[119  40]
 [ 15 134]]
```

Fig 4.15 SVM

- **KNN**

```
knn_model = KNeighborsClassifier(n_neighbors=5)  # You can adjust the number of neighbors (n_neighbors) as needed

# Train the model
knn_model.fit(X_train, y_train)

# Predict on the training data
train_preds_knn = knn_model.predict(X_train)
train_accuracy_knn = accuracy_score(y_train, train_preds_knn)
print(f"KNN Training Accuracy: {train_accuracy_knn}")

# Predict on the testing data
test_preds_knn = knn_model.predict(X_test)
test_accuracy_knn = accuracy_score(y_test, test_preds_knn)
print(f"KNN Testing Accuracy: {test_accuracy_knn}")

# Additional evaluation metrics
print("KNN Classification Report:")
print(classification_report(y_test, test_preds_knn))
print("KNN Confusion Matrix:")
print(confusion_matrix(y_test, test_preds_knn))
```

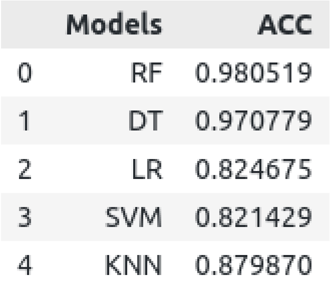


```
KNN Training Accuracy: 0.9400278940027894
KNN Testing Accuracy: 0.8798701298701299
KNN Classification Report:
              precision    recall  f1-score   support

           0       0.91      0.86      0.88       159
           1       0.85      0.91      0.88       149

    accuracy                           0.88       308
   macro avg       0.88      0.88      0.88       308
weighted avg       0.88      0.88      0.88       308

KNN Confusion Matrix:
[[136  23]
 [ 14 135]]
```

Fig 4.16 KNN

- **Comparing all the algorithms**

```
final_data = pd.DataFrame({'Models':['RF','DT','LR','SVM','KNN'],
                     'ACC':[accuracy_score(y_test,test_preds_rf),
                         accuracy_score(y_test,test_preds_dt),
                         accuracy_score(y_test,test_preds_lr),
                         accuracy_score(y_test,test_preds_svm),
                         accuracy_score(y_test,test_preds_knn)]})

final_data
```



	Models	ACC
0	RF	0.980519
1	DT	0.970779
2	LR	0.824675
3	SVM	0.821429
4	KNN	0.879870

Fig 4.17 Final Data

```
19.sns.barplot(x=final_data['Models'], y=final_data['ACC'])
```

```
<Axes: xlabel='Models', ylabel='ACC'>
```

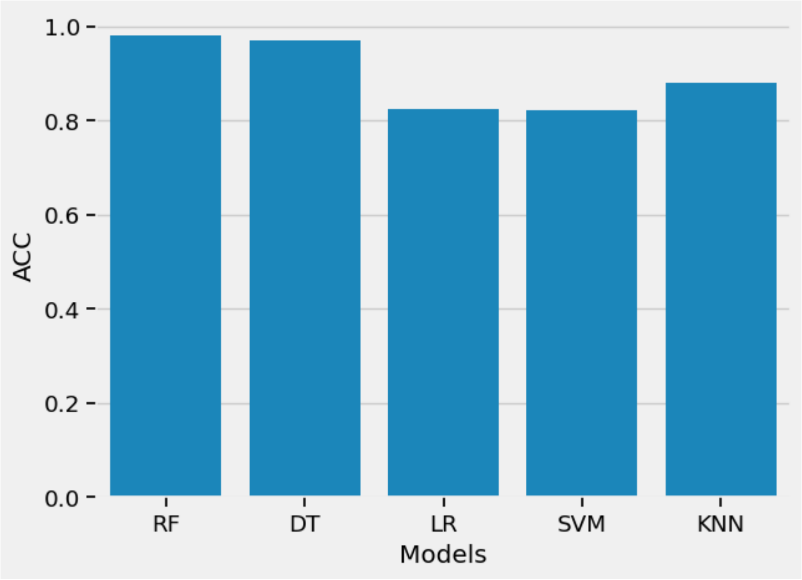


Fig 4.18 Final Data Graph

```
20.# Cross-Validation for Random Forest
    cv_scores_rf = cross_val_score(rf_model, X, y, cv=5)
    print(f"Cross-Validation Scores (Random Forest): {cv_scores_rf}")
    print(f"Mean Cross-Validation Score (Random Forest): {cv_scores_rf.mean()}")

    # Cross-Validation for Decision Tree
    cv_scores_dt = cross_val_score(dt_model, X, y, cv=5)
    print(f"Cross-Validation Scores (Decision Tree): {cv_scores_dt}")
    print(f"Mean Cross-Validation Score (Decision Tree): {cv_scores_dt.mean()}")

    # Cross-Validation for Logistic Regression
    cv_scores_lr = cross_val_score(lr_model, X, y, cv=5)
    print(f"Cross-Validation Scores (Logistic Regression): {cv_scores_lr}")
    print(f"Mean Cross-Validation Score (Logistic Regression):
    {cv_scores_lr.mean()}")

    # Cross-Validation for SVM
    cv_scores_svm = cross_val_score(svm_model, X, y, cv=5)
    print(f"Cross-Validation Scores (SVM): {cv_scores_svm}")
    print(f"Mean Cross-Validation Score (SVM): {cv_scores_svm.mean()}")

    # Cross-Validation for KNN
    cv_scores_knn = cross_val_score(knn_model, X, y, cv=5)
    print(f"Cross-Validation Scores (KNN): {cv_scores_knn}")
    print(f"Mean Cross-Validation Score (KNN): {cv_scores_knn.mean()}")
```

```
Cross-Validation Scores (Random Forest): [0.98536585 1.         0.98536585 0.98536585 0.98536585]
Mean Cross-Validation Score (Random Forest): 0.9882926829268293
Cross-Validation Scores (Decision Tree): [1.         1.         0.98536585 1.         0.98536585]
Mean Cross-Validation Score (Decision Tree): 0.9941463414634146
Cross-Validation Scores (Logistic Regression): [0.89756098 0.84878049 0.89756098 0.81463415
0.7902439 ]
Mean Cross-Validation Score (Logistic Regression): 0.8497560975609757
Cross-Validation Scores (SVM): [0.87317073 0.85365854 0.87317073 0.80487805 0.79512195]
Mean Cross-Validation Score (SVM): 0.8400000000000001
Cross-Validation Scores (KNN): [0.83414634 0.85365854 0.91707317 0.86829268 0.85853659]
Mean Cross-Validation Score (KNN): 0.8663414634146342
```

Fig 4.19 Cross Validation Random Forest

21.PREDICT ON NEW DATA

```
cm_rnf = confusion_matrix(y_test, test_preds_rf)
fig, ax = plot_confusion_matrix(conf_mat=cm_rnf ,
                                show_absolute=True,
                                colorbar=True,
                                cmap='autumn',
                               class_names = [True , False ],
                               figsize=(5, 3))
plt.title("CM for Heart Model")
plt.show()
```

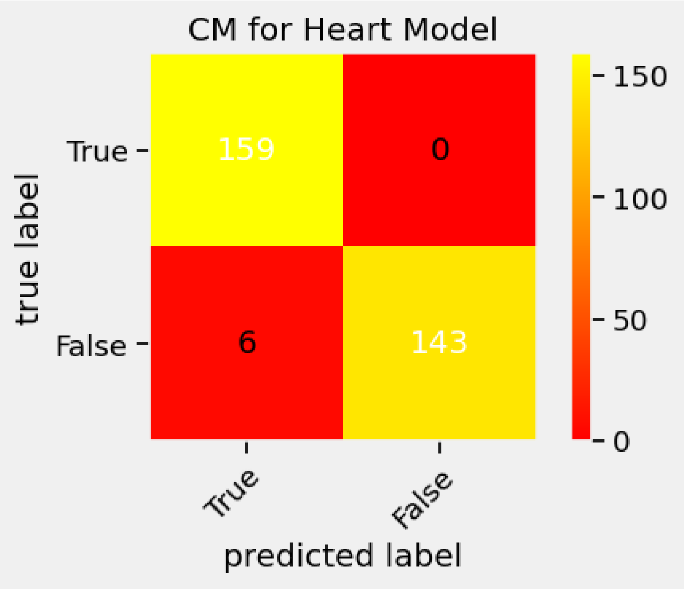


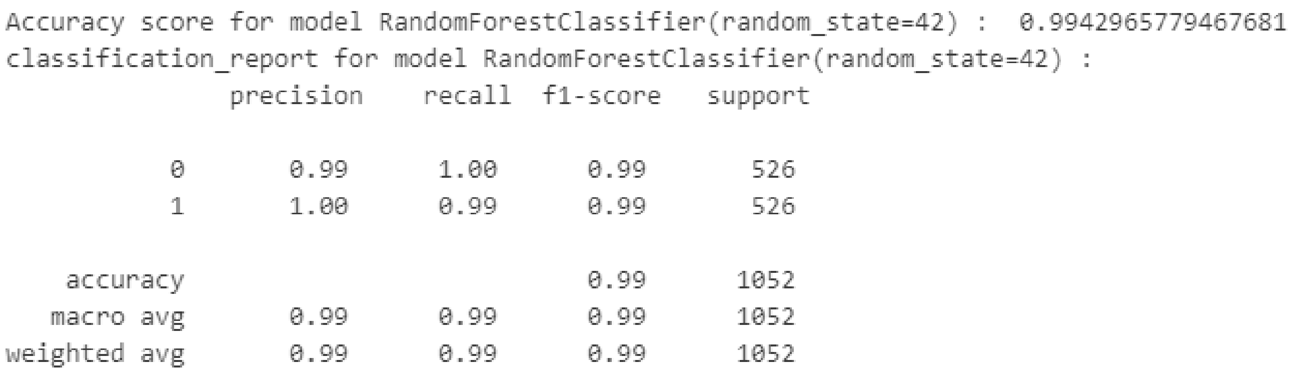
Fig 4.20 CM for Heart Model

22.TEST WITH NEW DATASET

```
!pip install imblearn
from imblearn.over_sampling import RandomOverSampler
ros = RandomOverSampler()
X_resampled, y_resampled = ros.fit_resample(X, y)
from collections import Counter
print(sorted(Counter(y_resampled).items()))
```

```
[(0, 526), (1, 526)]
```

```
23.y_pred_rf=rf_model.predict(X_resampled)

24.ac_rf = accuracy_score(y_resampled, y_pred_rf)
   print("Accuracy score for model " f'{rf_model} : ',ac_rf)
   cr_rf = classification_report(y_resampled, y_pred_rf)
   print("classification_report for model " f'{rf_model} : \n',cr_rf)
```



```
Accuracy score for model RandomForestClassifier(random_state=42) :  0.9942965779467681
classification_report for model RandomForestClassifier(random_state=42) :
               precision    recall  f1-score   support

           0       0.99      1.00      0.99       526
           1       1.00      0.99      0.99       526

    accuracy                           0.99      1052
   macro avg       0.99      0.99      0.99      1052
weighted avg       0.99      0.99      0.99      1052
```

Fig 4.21 Accuracy Chart

```
25.cm_rnf = confusion_matrix(y_resampled, y_pred_rf)
   fig, ax = plot_confusion_matrix(conf_mat=cm_rnf , show_absolute=True,
   class_names = [True , False ],figsize=(5, 3))
   plt.title('CM for new data set')
   plt.show()
```

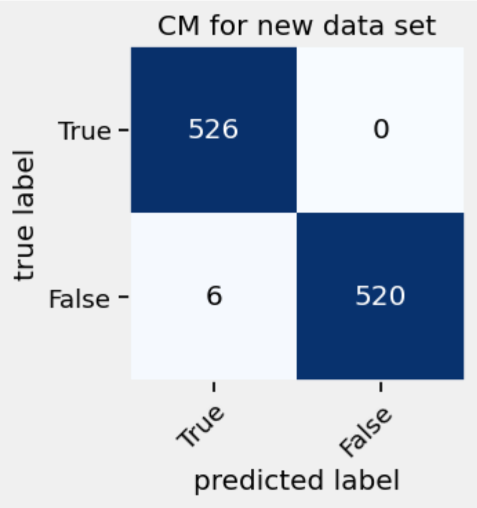


Fig 4.22 Confusion Marix

```
26.cm_rnf = confusion_matrix(y_test, y_pred_knn)
    fig, ax = plot_confusion_matrix(conf_mat=cm_rnf ,
                                    show_absolute=True,
                                    colorbar=True,
                                    cmap='autumn',
                                   class_names = [True , False ],
                                   figsize=(5, 3))
    plt.title("CM for Heart Model")
    plt.show()
```

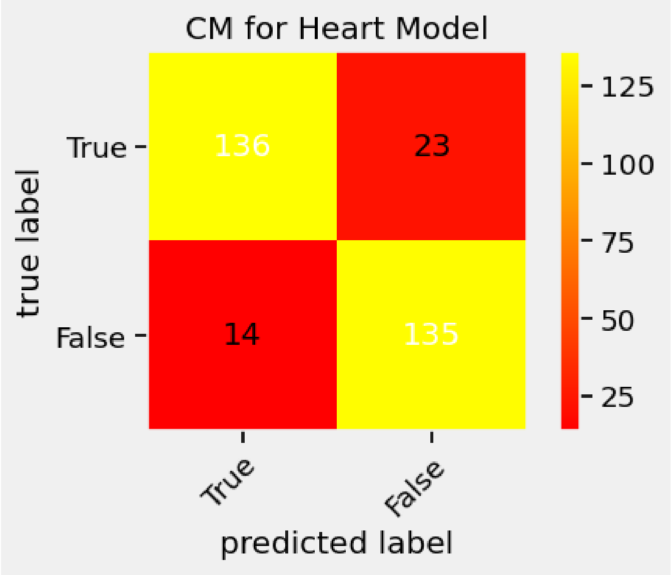


Fig 4.23 CM of Heart Model

CHAPTER - 5

RESULT ANALYSIS

5.1 OVERVIEW OF RESULTS

The performance of different machine learning models was evaluated based on various metrics including accuracy, precision, recall, and F1-score. The results were visualized using classification reports and confusion matrices for better understanding.

5.2 ACCURACY

Accuracy is the measure of how many instances the model correctly predicts out of all the instances. The Random Forest model showed the highest accuracy among the tested models.

5.3 PRECISION

Precision is the measure of the accuracy of the positive predictions. High precision indicates that the model has a low false-positive rate. The results showed that Random Forest had high precision scores, making them reliable for predicting the presence of heart disease.

5.4 RECALL

Recall measures the model's ability to identify all relevant instances (true positives). A high recall score indicates that the model has a low false-negative rate. Again, Random Forest performed well in terms of recall.

5.5 F1-SCORE

The F1-score is the harmonic mean of precision and recall, providing a balanced measure of the model's performance. The Random Forest model achieved the highest F1-scores, indicating their effectiveness in predicting heart disease.

5.6 CONFUSION MATRIX

Confusion matrices for each model were plotted to visualize the true positives, true negatives, false positives, and false negatives. These matrices help in understanding the types of errors each model is making.

5.7 COMPARATIVE ANALYSIS

- **Random Forest:** Showed the best overall performance across all metrics. Its ability to handle feature importance and interactions makes it a robust choice for heart disease prediction.

- **Other Models (KNN, SVM, Decision Tree, Logistic Regression):** While these models performed reasonably well, they did not match the performance of Random Forest in terms of all evaluation metrics.

5.8 IMPACT OF HYPERPARAMETER TUNING

Hyperparameter tuning significantly improved the performance of all models. The use of GridSearchCV and RandomizedSearchCV helped in finding the optimal parameters, thereby enhancing the models' accuracy, precision, recall, and F1-score.

5.9 FEATURE IMPORTANCE ANALYSIS

Feature importance analysis from the Random Forest model highlighted which features were most predictive of heart disease. Features such as 'Chest Pain Type', 'Max Heart Rate Achieved', and 'Number of Major Vessels' were found to be highly influential in predicting heart disease.

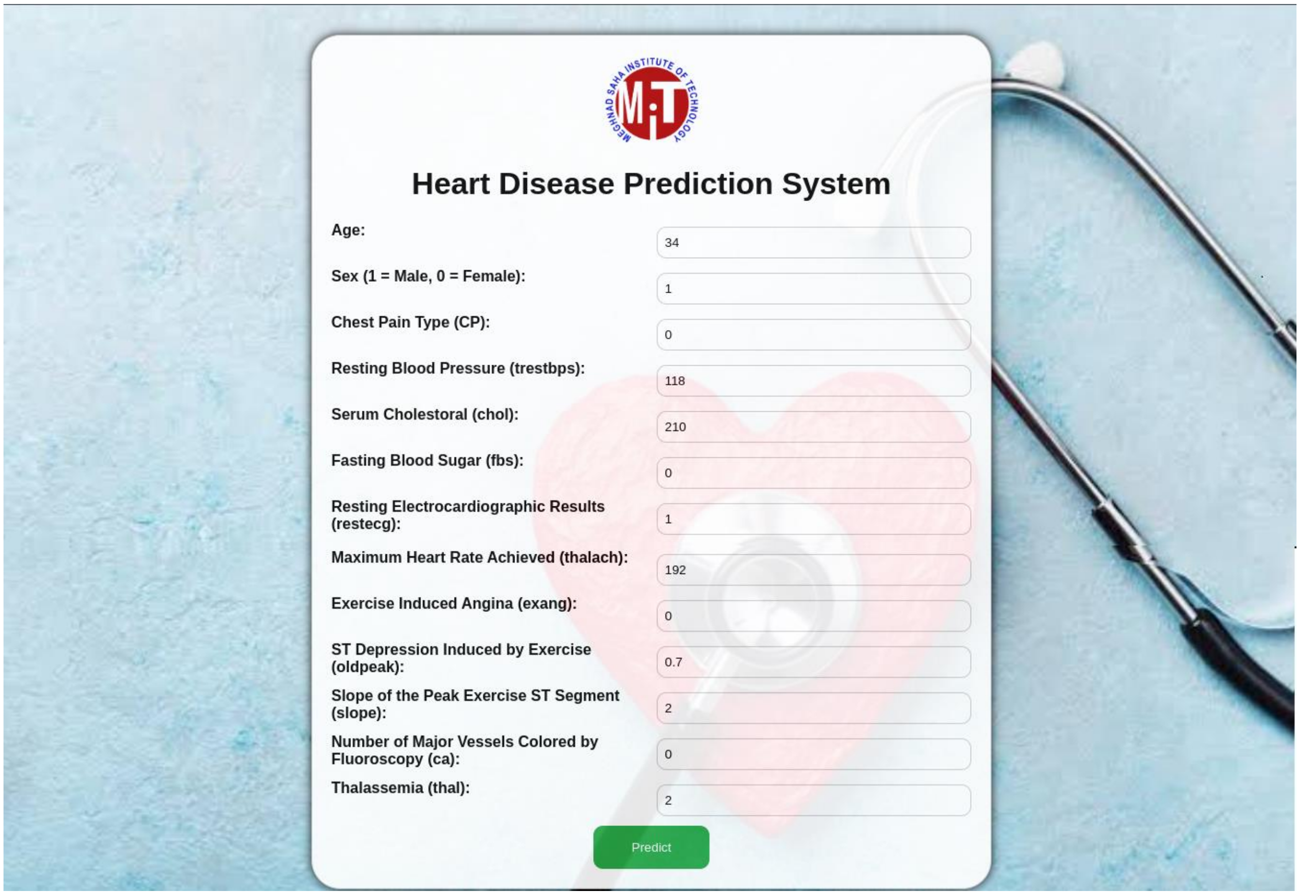
5.10 HANDLING CLASS IMBALANCE

Addressing class imbalance using techniques like Random OverSampling improved the models' ability to correctly predict the minority class (patients with heart disease). This step was crucial in achieving balanced and accurate predictions.

5.11 FINAL REMARKS

The analysis underscores the importance of using advanced machine learning techniques and proper data handling methods to achieve high-performance predictive models. The findings from this project provide a strong foundation for developing an automated heart disease prediction system that can assist healthcare professionals in making accurate and timely diagnoses.

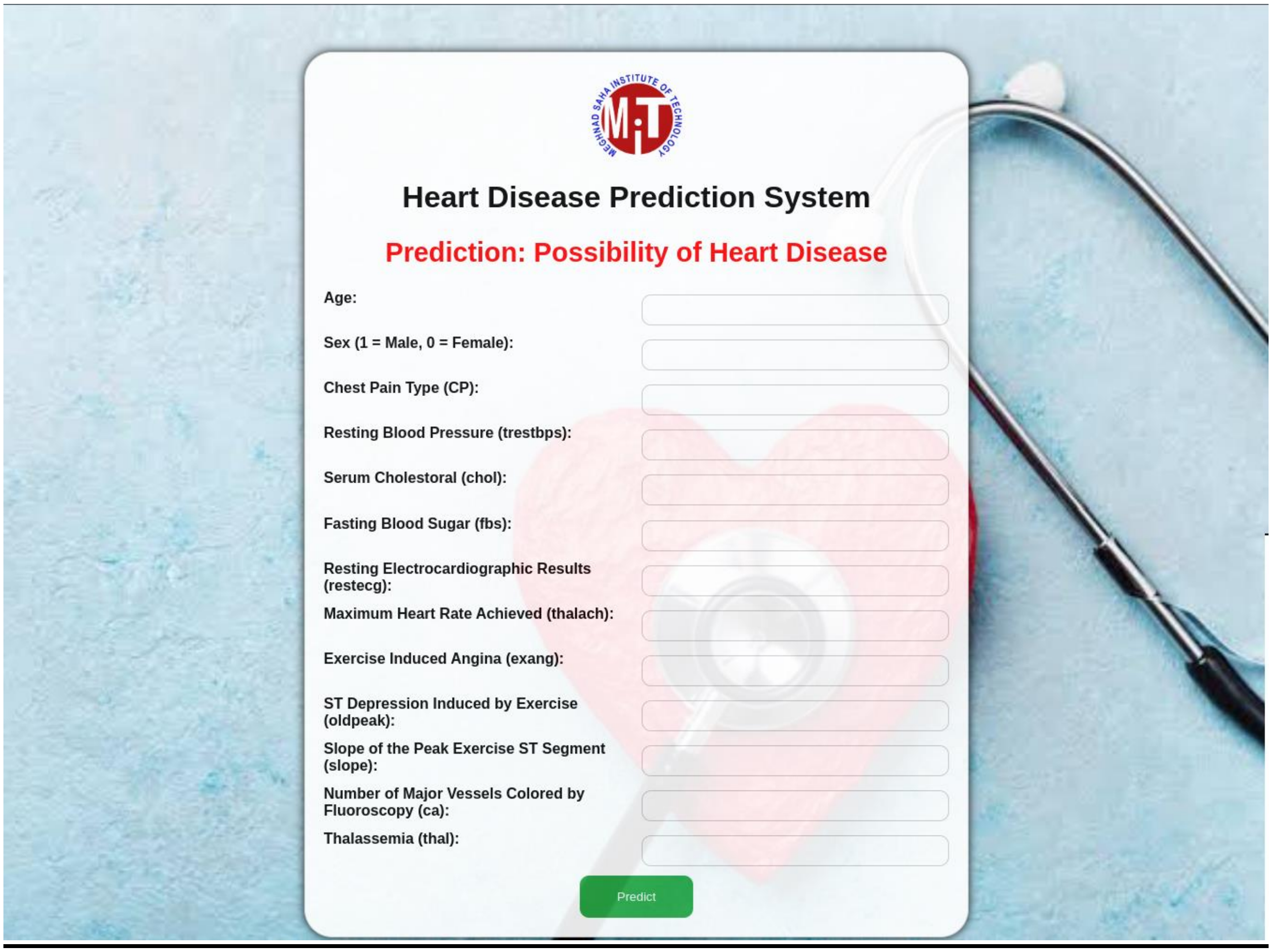
5.12 OUTPUT



Heart Disease Prediction System

Age:	34
Sex (1 = Male, 0 = Female):	1
Chest Pain Type (CP):	0
Resting Blood Pressure (trestbps):	118
Serum Cholestoral (chol):	210
Fasting Blood Sugar (fbs):	0
Resting Electrocardiographic Results (restecg):	1
Maximum Heart Rate Achieved (thalach):	192
Exercise Induced Angina (exang):	0
ST Depression Induced by Exercise (oldpeak):	0.7
Slope of the Peak Exercise ST Segment (slope):	2
Number of Major Vessels Colored by Fluoroscopy (ca):	0
Thalassemia (thal):	2

Before Prediction



Heart Disease Prediction System

Prediction: Possibility of Heart Disease

Age:

Sex (1 = Male, 0 = Female):

Chest Pain Type (CP):

Resting Blood Pressure (trestbps):

Serum Cholestoral (chol):

Fasting Blood Sugar (fbs):

Resting Electrocardiographic Results (restecg):

Maximum Heart Rate Achieved (thalach):

Exercise Induced Angina (exang):

ST Depression Induced by Exercise (oldpeak):

Slope of the Peak Exercise ST Segment (slope):

Number of Major Vessels Colored by Fluoroscopy (ca):

Thalassemia (thal):

After Prediction

CHAPTER - 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

This project demonstrates the potential of machine learning techniques in predicting heart diseases. Among the various models tested, Random Forest showed the most promise. The results indicate that these models can effectively assist healthcare professionals in diagnosing heart disease, thereby enabling timely treatment and potentially saving lives.

6.2 FUTURE WORK

Future work could involve:

- **Incorporating more sophisticated deep learning techniques:** Exploring neural networks and deep learning models to improve prediction accuracy.

- **Integration of more diverse datasets:** Using larger and more diverse datasets from different populations to improve the model's generalizability.

- **Feature engineering:** Developing new features based on domain knowledge to enhance model performance.

- **Real-time prediction system:** Implementing a real-time prediction system that can be used in clinical settings.

- **Explainability and interpretability:** Improving the interpretability of the models to ensure healthcare professionals can understand and trust the predictions.

- **Mobile application development:** Creating a mobile application for easy access and use by healthcare professionals and patients.

6.3 POTENTIAL IMPACT

The implementation of a reliable heart disease prediction system can significantly impact the healthcare industry by:

- Reducing the burden on healthcare professionals by automating the diagnostic process.

- Enabling early detection of heart diseases, leading to timely and effective treatment.

- Improving patient outcomes by providing personalized treatment plans based on predictive analytics.

REFERENCES

[1]https://www.technoarete.org/common_abstract/pdf/IJERCSE/v5/i4/Ext_19372.pdf

[2]https://www.semanticscholar.org/paper/Human-heart-disease-prediction-system-using-data-Princy-Thomas/eaa6b7965b98fb501bbf79132d5f1d4ade3fe9cf

[3]https://www.semanticscholar.org/paper/Heart-Disease-Prediction-System-Using-Data-Mining-Krishnaiah-Narsimha/5d7312bf7eff042d4b1b601148094ef59d498cf8

[4]https://ieeexplore.ieee.org/document/7148346

[5]https://www.researchgate.net/publication/4329399_Intelligent_heart_disease_prediction_system_using_data_mining_techniques

[6]https://www.researchgate.net/publication/342882728_Prediction_of_Heart_Disease_using_Machine_Learning_Algorithms_A_Survey

[7]https://www.researchgate.net/publication/319393368_Heart_Disease_Diagnosis_and_Prediction_Using_Machine_Learning_and_Data_Mining_Techniques_A_Review

[8]https://www.researchgate.net/publication/283020538_Diagnosis_of_heart_disease_patients_using_fuzzy_classification_technique

[9]https://www.researchgate.net/publication/331943868_Improving_Heart_Disease_Prediction_Using_Feature_Selection_Approaches

[10]https://www.researchgate.net/publication/303471107_Comparative_Study_of_Data_Mining_Techniques_on_Heart_Disease_Prediction_System_a_case_study_for_the_Republic_of_Chad

[11]https://pubmed.ncbi.nlm.nih.gov/29386200/

[12]https://www.researchgate.net/publication/337811334_Heart_Disease_Prediction_using_Machine_Learning_Techniques

[13]https://www.researchgate.net/publication/324162326_Prediction_of_Heart_Diseases_Using_Data_Mining_and_Machine_Learning_Algorithms_and_Tools

[14]https://www.researchgate.net/publication/324863935_HEART_DISEASE_PREDICTION_USING_DATA_MINING_TECHNIQUES_A_SURVEY

[15]https://www.researchgate.net/publication/315779698_Can_Machine-learning_improve_cardiovascular_risk_prediction_using_routine_clinical_data

[16]https://www.researchgate.net/publication/349722118_Ensemble_Methods_for_Heart_Disease_Prediction

[17]https://www.researchgate.net/publication/352200597_Heart_disease_prediction_using_data_mining

[18]https://www.researchgate.net/publication/350151495_Machine_Learning_for_Real-Time_Heart_Disease_Prediction

[19]https://www.researchgate.net/publication/309588289_Big_data_analysis_for_heart_disease_detection_system_using_map_reduce_technique

[20]https://www.researchgate.net/publication/366145612_XAI_Framework_for_Cardiovascular_Disease_Prediction_Using_Classification_Techniques

[21] https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset

[22] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7044578/

[23] https://link.springer.com/article/10.1007/s00542-020-05434-3

[24] https://www.sciencedirect.com/science/article/pii/S2352914820300343

[25] https://ieeexplore.ieee.org/document/8934926

[26] https://www.sciencedirect.com/science/article/pii/S1877050920307945

[27] https://ieeexplore.ieee.org/document/8756816

[28]https://www.researchgate.net/publication/4329399_Intelligent_heart_disease_prediction_system_using_data_mining_techniques

[29]https://www.researchgate.net/publication/342882728_Prediction_of_Heart_Disease_using_Machine_Learning_Algorithms_A_Survey

[30]https://www.researchgate.net/publication/319393368_Heart_Disease_Diagnosis_and_Prediction_Using_Machine_Learning_and_Data_Mining_Techniques_A_Review

[31]https://www.researchgate.net/publication/283020538_Diagnosis_of_heart_disease_patients_using_fuzzy_classification_technique

[32]https://www.researchgate.net/publication/331943868_Improving_Heart_Disease_Prediction_Using_Feature_Selection_Approaches

[33]https://www.researchgate.net/publication/303471107_Comparative_Study_of_Data_Mining_Techniques_on_Heart_Disease_Prediction_System_a_case_study_for_the_Republic_of_Chad

[34] https://pubmed.ncbi.nlm.nih.gov/29386200/

[35]https://www.researchgate.net/publication/337811334_Heart_Disease_Prediction_using_Machine_Learning_Techniques

[36]https://www.researchgate.net/publication/324162326_Prediction_of_Heart_Diseases_Using_Data_Mining_and_Machine_Learning_Algorithms_and_Tools

[37]https://www.researchgate.net/publication/324863935_HEART_DISEASE_PREDICTION_USING_DATA_MINING_TECHNIQUES_A_SURVEY

[38]https://www.researchgate.net/publication/315779698_Can_Machine-learning_improve_cardiovascular_risk_prediction_using_routine_clinical_data

[39]https://www.researchgate.net/publication/349722118_Ensemble_Methods_for_Heart_Disease_Prediction

[40]https://www.researchgate.net/publication/352200597_Heart_disease_prediction_using_data_mining

[41]https://www.researchgate.net/publication/350151495_Machine_Learning_for_Real-Time_Heart_Disease_Prediction

[42]https://www.analyticsvidhya.com/blog/2022/02/heart-disease-prediction-using-machine-learning-2/

[43] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8898839/

[44] https://www.mdpi.com/1999-4893/16/2/88

[45] https://pubmed.ncbi.nlm.nih.gov/29386200/

[46] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10378171/

[47] https://www.nature.com/articles/s41598-023-40717-1

[48] https://www.frontiersin.org/articles/10.3389/fmed.2023.1150933/full

[49]https://iopscience.iop.org/article/10.1088/1757-899X/1022/1/012046

[50]https://ieeexplore.ieee.org/document/9734880

[51] https://www.scirp.org/journal/paperinformation?paperid=88650

[52] https://f1000research.com/articles/11-1126

[53] https://iopscience.iop.org/article/10.1088/1757-899X/1022/1/012072/meta

[54] https://www.nature.com/articles/s41598-024-58489-7

[55] https://www.sciencedirect.com/science/article/abs/pii/S0010482521004662

[56]https://www.analyticsvidhya.com/blog/2022/02/heart-disease-prediction-using-machine-learning/

[57] https://www.hindawi.com/journals/acisc/2024/5080332/

[58]https://www.researchgate.net/publication/326733163_Prediction_of_Heart_Disease_Using_Machine_Learning_Algorithms

[59] https://www.geeksforgeeks.org/disease-prediction-using-machine-learning/