DISEASE DETECTION

A major project report submitted in partial fulfilment of the requirement for the award of degree of

Bachelor of Technology

in

Computer Science & Engineering / Information Technology

Submitted by

Vedant Tiwari (201130)

Aditya Sahni (201442)

Under the guidance & supervision of

Dr. Maneet Singh Assistant Professor

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology, Waknaghat, Solan-173234 (India)

TABLE OF CONTENTS

Candidate’s Declaration

We hereby declare that the work presented in this report entitled **‘Disease Detection’** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science & Engineering / Information Technology** submitted in the Department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology, Waknaghat is an authentic record of my own work carried out over a period from August 2023 to May 2024 under the supervision of **Dr. Maneet Singh** (Assistant Professor, Department of Computer Science & Engineering and Information Technology).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

(Student Signature with Date)	(Student Signature with Date)
Student Name: Aditya Sahni	Student Name: Vedant Tiwari
Roll No.: 201442	Roll No.: 201130

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

(Supervisor Signature with Date)
Supervisor Name: Dr. Maneet Singh
Designation: Assistant Professor
Department: Computer Science & Information Technology
Dated:

CERTIFICATION

This is to certify that the work which is being presented in the project report titled “Disease Detection” in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science And Engineering and submitted to the Department of Computer Science And Engineering, Jaypee University of Information Technology, Waknaghat is an authentic record of work carried out by “Vedant Tiwari(201130) and Aditya Sahni(201442)” during the period from August 2023 to May 2024 under the supervision of Mr. Maneet Singh, Department of Computer Science and Engineering, Jaypee University of Information Technology, Waknaghat.

Vedant Tiwari (201130)
Aditya Sahni (201442)

The above statement made is correct to the best of my knowledge.

Mr. Maneet Singh
Assistant Professor
Computer Science & Engineering and Information Technology
Jaypee University of Information Technology, Waknaghat

Acknowledgement

Firstly, I express my heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the project work successfully.

I am really grateful and wish my profound my indebtedness to Supervisor Mr. Maneet Singh, Assistant Professor, Department of CSE Jaypee University of Information Technology, Waknaghat. Deep Knowledge & keen interest of my supervisor in the field of Deep Learning to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

I would also generously welcome each one of those individuals who have helped me straight forwardly or in a roundabout way in making this project a win. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, I must acknowledge with due respect the constant support and patients of our parents.

Vedant Tiwari (201130),
Aditya Sahni (201442)

ABSTRACT

Data mining technology refers to the process of extracting valuable information from extensive datasets, and its applications are widespread in various aspects of human life. One prominent domain benefiting from data mining is healthcare, where it plays a significant role. Among health applications, the focus often turns to heart disease, a globally prevalent and perilous chronic condition. This study aims to predict the occurrence of heart disease in patients using the random forest algorithm.

The dataset, comprising 1025 samples and 14 attributes as features, was sourced from Kaggle. Processing was executed through the Python open-access software in a Jupyter notebook. Machine learning algorithms, specifically the random forest algorithm, were employed to classify and process the datasets. The results are presented in terms of accuracy, sensitivity, and specificity, all expressed as percentages.

The application of the random forest algorithm yielded an accuracy of 87.3% in predicting heart disease, with a precision value of 82.9% and recall value of 94.1%. Additionally, the F1 score demonstrated a diagnosis rate of 88.1% for predicting heart disease using the random forest algorithm. The efficiency exhibited by the random forest algorithm in the classification of heart disease underscores its suitability for the proposed system.

LIST OF FIGURES

LIST OF TABLES

Chapter 01: INTRODUCTION

1.1 Introduction

"The process of manipulating and extracting implicit, known or previously unknown, and possibly relevant information from data is known as machine learning [1]. The use and scope of machine learning are always growing, making it a broad and varied area. It includes a range of ensemble learning, supervised learning, and unsupervised learning classifiers that are used to forecast and assess the correctness of given datasets. This information can be used to assist a huge population in programmes like HDPS. Cardiovascular diseases are a broad category of heart-related ailments that are common in modern culture. Cardiovascular Diseases (CVDs) account for 17.9 million deaths worldwide, according to the World Health Organisation [2], making it the top cause of adult mortality. Our goal is to use a patient's medical history to forecast who is most likely to receive a heart disease diagnosis [6]. It helps diagnose diseases with fewer medical tests and more effective treatments by recognising symptoms like high blood pressure or chest pain, which contributes to prompt and focused care.

Three main data mining techniques are the subject of this project: Random Forest Classifier, KNN, and Logistic Regression. The project outperforms earlier systems that only use one data mining technique, with an accuracy of 87.5%. This project includes the supervised learning technique of logistic regression, which works with discrete values. The goal is to determine a patient's likelihood of receiving a diagnosis of cardiovascular heart disease based on characteristics like age, gender, chest discomfort, fasting blood sugar, etc. The project determines if a patient may have cardiac illness by using a dataset from the UCI repository that includes patient medical history and features.. Three algorithms—Random Forest Classifier, KNN, ANN, and Logistic Regression are used to train the 14 medical characteristics. Random Forest is the most effective of these, obtaining recall of 94%. Lastly, a cost-effective technique is demonstrated for classifying people at risk of heart disease."

1.2 Objectives

1. Create a state-of-the-art technique for identifying heart disease: The main goal of this project is to build a reliable system that can correctly identify heart disease based on input data.
2. Model Selection and Implementation: Using various machine learning algorithms, such as random forest, decision tree, KNN, and others, investigate and use suitable machine learning and deep learning models.
3. Assessment and Performance Measures: Thoroughly verify the accuracy and dependability of the system. Work together for clinical validation with professionals in medicine. Install the programme and integrate it with current systems in actual healthcare settings.

By fulfilling these goals, the project hopes to improve the healthcare system and enable patients and medical professionals to identify and treat cardiac disease early on. It also hopes to promote disease detection and prevention and its practical use in a variety of fields.

1.3 Motivation

Depending on the project's objectives and circumstances, the motivation behind it might take many different forms. The following are some typical reasons to start a project on disease detection:

1. **Early Detection and Intervention:**

   Fast intervention and therapy are made possible by early disease identification, which frequently improves patient outcomes.

   Early disease detection lowers morbidity and death rates by halting the spread of the illness to more advanced stages.

2. **Improved Patient Outcomes:**

   By helping medical professionals to identify and treat illnesses at an earlier and more treatable stage, disease detection initiatives enhance patient outcomes.

   Improved illness prediction accuracy can result in individualised and focused treatment regimens for patients. 3. Assistive technology and accessibility: People with impairments, particularly those with speech difficulties, can benefit from SER assistance. It enhances these people's quality of life by facilitating improved communication and emotional expression.

3. **Reduced Healthcare Costs:**

   Preventive measures and early interventions are frequently more cost-effective than treating advanced diseases. Early disease diagnosis can result in cost savings by minimising the need for extensive and expensive medical interventions associated with advanced-stage diseases.

4. **Public Health Impact:**

   By using data to diagnose diseases, medical practises can become more innovative and efficient by adopting a more data-driven approach to healthcare.

   Patterns and insights that may not be seen through conventional approaches can be found in massive datasets thanks to advanced analytics and machine learning techniques.

5. **Technological Advancements:**

   Wearable technology, artificial intelligence, and machine learning are examples of technological advancements that open up new possibilities for more precise and effective illness detection.

   Disease detection initiatives become more capable as a result of ongoing innovation in healthcare technology.

6. **Empowering Patients:**

Disease detection initiatives have the potential to enable people to take charge of their health by educating them about their risk factors and promoting proactive health care.

7. **Global Health Security:**

Rapid and precise detection is critical for limiting the transmission of diseases across borders.

Disease detection is essential for global health security, helping to identify and respond to new infectious diseases that could pose hazards on a global scale.

8. **Research and Development Opportunities:**

Participating in illness detection initiatives supports continued medical research and development, encouraging creativity and the creation of novel diagnostic instruments and techniques.

1.4 Tools & Techniques

Because of its adaptability, large library, and developing machine learning and app development community, Python and Android development are a sensible choice for a project involving disease detection. For a disease detection project, the following methods for using Python and Android development can be applied:

Tools used are:

1. VS code
2. Python
3. Android Studio
4. Java
5. Kotlin

Procedures:

1. **Data Collection and Preprocessing:** Compile a wide range of patient records, such as imaging, diagnostic tests, and medical history. Employ Python libraries to accomplish this quickly and effectively.

2. **Machine Learning and Deep Learning:** To construct ML models, a variety of machine learning and deep learning libraries are available in Python. `scikit- learn` is a well-liked library for conventional machine learning methods, and
`TensorFlow` and `PyTorch` are prominent deep learning frameworks for creating neural networks.

3. **Feature Extraction:** Relevant features can be extracted from the data using Python. For feature extraction, libraries like "scikit-learn," "Pandas," and "Numpy" are frequently utilised.

4. **Model Training:** Python lets you use ".csv" files as input data to train both machine learning and deep learning models. K-Nearest Neighbours (KNNs), Convolutional Neural Networks (CNN), Random Forest, or Logistic Regression can all be used to implement and train models.

5. **Model evaluation and prediction:** After the model has been trained, it must be tested using various data sets. And assess its performance using assessment measures such as recall, accuracy, precision, F1-score, AUC, RUC, etc.

6. **Visualization:** Python provides a number of modules for data visualisation that are useful for displaying data, the status of model training, and the outcomes. The popular libraries `matplotlib` and ‘seaborn’ are used for visualisation.

7. **Putting an Android application into action:** Lastly, connect the model that has been trained in Python to be used in an Android application by using Android

programming approaches. This will improve the user experience among many other benefits.

1.5 Technical Requirements

1. Hardware Requirements:
    a. **Processing Power:** An efficient computer with a CPU or GPU that can handle the training and inference operations will depend on how sophisticated the machine learning models are.
    b. **Memory:** Sufficient RAM is essential for loading and processing large datasets, especially for using deep learning models.
    c. **Storage:** Storage space for audio datasets, model checkpoints, and other project-related files.
2. Software Requirements:
    a. **Python:** Python is used for the majority of disease detection applications because of its large libraries for data processing, analysis, and machine learning.
    b. **Development Environment:** For writing and debugging code, use an IDE for Python such as Google Colab, Kaggle Notebook, or Visual Studio Code. and Android Studio for creating, troubleshooting, and illustrating the Android application development process.
    c. **Machine Learning Frameworks:** Install the required libraries for machine learning and deep learning, including Keras, TensorFlow, PyTorch, and scikit-learn.
    d. **Data Processing Libraries:** For data preprocessing, libraries like sklearn or Keras are frequently utilised.
    e. **Backend for android application:** Utilise Java frameworks and API for the Android application's backend development.

f. **Frontend:** Utilise Android Studio to create an intuitive user interface.

1.6 Deliverables/ Outcomes

Depending on the precise objectives and project scope, a disease detection project may have different deliverables and results. Nonetheless, the following typical deliverables and possible results from a disease detection project are what you can anticipate:

1. ***Trained Disease Detection Model:*** A trained machine learning or deep learning model that can identify disease based just on the data entered is the main deliverable. This model should be able to forecast the disease using a variety of input data sources, including photographs, findings from diagnostic tests, and medical records. Moreover, additional dos and don'ts for the condition.

2. ***Accuracy Metrics:*** A record or report including information on the trained model's performance, including confusion matrices, accuracy, precision, recall, and F1- score. This offers an assessment of the model's predictive accuracy for the illness.

3. **Codebase:** The codebase of the project, which includes, if relevant, real-time prediction implementations, model training code, and scripts for prepping data. To aid in comprehension and future development, appropriate documentation and code comments are included.

4. **Demo/Prototype:** A functioning prototype or demo demonstrating how the illness detector system works, particularly if it uses real-time patient data as input.

Chapter 02: Literature Survey

Table 1: Literature Survey

S.no.	Paper Title	Journal/ Conference (Year)	Tools/ Techniques/ Dataset	Results	Limitations
1.	Machine learning based marker for coronary artery disease: derivation and validation in two longitudinal cohorts	IEEE Access 2023	Tools: Python, scikit-learn, TensorFlow Techniques: Support Vector Machine (SVM), Convolutional Neural Networks (CNN), feature extraction, data preprocessing Dataset: 1,000 patient records	Achieved an accuracy of 89% in detecting coronary artery disease based on multimodal medical data. - Sensitivity and specificity were 87% and 91%, respectively	Small dataset size may limit generalizability to larger populations.
2.	Skin Disease Detection using Convolutional Neural Network	MDPI - Sensors 2022	Skin Cancer MNSIT: HAM10000; the info was divided into a training set (80% of spectral dataset) and a test set (20% of spectral dataset).	The system was seen to predict the said diseases with an accuracy of around 74% - 75%. The model loss can be seen decreasing abruptly at first and then gradually towards the end.	Accuracy was comparatively low.

3.	The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions	MDPI - Sensors 2022	Using modern machine learning techniques based on random forests, the difference function is first trained using benchmark stream data and then used to predict the Reynolds stress difference in the new stream.	Image recognition technology based on machine learning can well identify white blood cells that are difficult to distinguish with the naked eye, with a recognition rate of up to 90%.	Requires a big dataset of images which is expensive in terms of storage and run time.
4.	A Classification Method of Heart Disease Based on Heart Sound Signal	IEEE Access 2019	Using modern machine learning techniques based on signal/ wavelength analysis, the difference function is first trained using benchmark stream data and then used to predict the polarity by comparing results.	Achieved an accuracy of 88%. Sensitivity and specificity were 87% and 92%, respectively.	Less hidden layers.

5.	Prediction of Heart Diseases using Random Forest	Springer 2012	Using random forest and data mining, the model is first trained using benchmark stream dataset of 303 samples and 14 attributes and then used for classification.	Achieved an accuracy of 86.9%. Sensitivity and specificity were 90.6% and 82.7%, respectively.	No in-depth analysis
6.	Nonlinear analysis of heart murmurs using wavelet-based higher-order spectral parameters	IEEE 2006	Signal processing	Achieved an accuracy of 89.9%. Sensitivity and specificity were 91.6% and 81.7%, respectively.	Complex process.
7.	Analysis of mice heart rate variability obtained through plethysmograph power spectrum	Computers in Cardiology 2001	Signal processing	The system was seen to predict the said diseases with an accuracy of around 85% - 90%.	Comparatively low accuracy.

2.1 Overview of Literature Survey

Based on the available data, it seems that the papers are mainly focused on deep learning approaches for disease identification. This is a summary of the literature with an assessment of any major gaps that may exist:

1. Datasets: Scholars have employed diverse datasets, the majority of which comprise either picture data or data pertaining to various attributes including age, gender, blood pressure, cholesterol, blood sugar level, max and min heart rates, etc.

2. Deep Learning Architectures: Various combinations of CNN, RNN, and other deep learning architectures with many hidden layers have been used. In the course of training the models, additional machine learning techniques including random forest, KNN, and decision tree are employed.

3. Performance Metrics: In several studies, performance indicators like accuracy are underlined. There are claims that using specific deep learning architectures increases accuracy.

4. Challenges: Since the dataset isn't easily accessible and contains a lot of patient personal data, most researchers have trouble gathering data. This could be because of security concerns. Thus, in order to gather data, several of them conducted surveys.

5. Specific Techniques: The majority of the investigators employed popular machine learning models and algorithms, such as Random Forest, ANN, RNN, KNN, Decision Tree, Logistic Regression, and so forth.

6. One of the researchers used sound signals for prediction by using signal processing in machine learning.

2.2 Key Gaps of Literature Survey

1. **Collecting Dataset:** While each paper shown the methodology of implementing and training the model but most of them didn’t talk about collection of data by keeping security and privacy in mind. As these data especially good quality data aren’t easy to collect because no one wish to share their private data and also hospitals doesn’t wish to share their patient’s data without keeping security and privacy in priority.

2. **Size of Dataset:** As these types of datasets aren’t available in bulk So, most of the dataset used seems small. Human body is very much unpredictive as it includes a lot of complex systems works in combination to result anything. One’s genetic, adaptability, biological, physical and other senses are whole different from others. So, predicting disease based on small datasets creates problem in terms of accuracy.

3. **Limited diversity in Datasets:** Except few, most of the researchers used same type of dataset that includes same attributes. And none of the researchers used patient’s past health records in way to predict the disease which is a key gap as the past health records play an important role in prediction.

4. None of the researchers used the patient’s daily schedule and lifestyle in a major way. Diseases like heart are very much dependent on one’s lifestyle. So this seems to be a key gap in all papers.

5. **Scalability Challenges:** While challenges related to scalability are mentioned, the specific nature of these challenges and potential solutions are not deeply discussed. Further investigation into how to scale up disease detection system for

real-world, large-scale applications is a potential gap. None of the papers have talked about scaling the project in a way that everyone could use with ease. None of the papers talked about use of the project with a user-friendly environment.

6. **Lack of Standardization:** Standardised benchmarks or assessment measures for disease detection systems are not mentioned. Creating standardised measures and benchmarks could make it easier to compare how well various models perform across investigations.

7. **Interdisciplinary Perspectives:** There appears to be a deficit in the investigation of multidisciplinary viewpoints, even if the publications concentrate on the technical elements of illness detection.

8. **Real-world Deployment Challenges:** The difficulty of implementing these systems in practical settings is touched upon. Additional research could look into the factors of user acceptance, ethical considerations, and practical problems.

By filling in these gaps, we may be able to gain a deeper and more thorough understanding of illness detection systems, which will improve their efficacy and practicality in real-world situations. To further the field, researchers can think about filling in these gaps in their future work.

Chapter 03: Feasibility Study, Requirement Analysis and Design

3.1 Feasibility Study

3.1.1 Problem Definition

Using a CSV file as the dataset, our goal is to create a machine learning model that can predict an individual's risk of developing heart disease based on their health- related characteristics.

Background:

One of the main causes of illness and death worldwide is cardiovascular disease, which includes heart disease. It is essential to identify those who are at risk of heart disease early in order to put preventive measures and individualised interventions into place. By using machine learning techniques, this study seeks to use pertinent health indicators to forecast an individual's likelihood of developing heart disease.

Dataset:

A CSV file with a number of health-related characteristics for a sample of people is used in the project. The CSV file is organised into rows that correspond to individual patients. The columns include attributes like age, gender, blood pressure, cholesterol, and other relevant health indicators. The dataset's quality and integrity are guaranteed since it comes from a reputable and pertinent source.

Problem Statement:

Create a prediction model that can determine a person's risk of heart disease based on the health-related characteristics that are presented. The CSV file's historical data should be used to train the model, allowing it to identify trends and connections between various characteristics and the existence or absence of heart disease.

Key Tasks:

1. **Data Preprocessing:**

   If any data are missing, handle them by removing or imputation.

   Scale or normalise numerical features to provide reliable model output.

   For model compatibility, adequately encode categorical variables.

2. **Exploratory Data Analysis (EDA):**

   Analyse exploratory data to learn more about the distribution of the variables.

   Determine the relationships between various characteristics and how they might affect heart disease.

3. **Model Selection:**

   Select machine learning methods that work well for problems involving categorization.

   Assess and contrast the effectiveness of various models by utilising suitable measures.

4. **Feature Selection:**

   Ascertain which characteristics have the greatest influence on the prognosis of cardiac disease.

   Consider the significance of each feature and, if needed, remove any unnecessary variables..

5. **Model Training:**

   Divide the dataset into sets for testing and training.

   Utilise the training set to train the chosen machine learning model..

6. **Model Evaluation:**

   Verify the model's generalizability to new, unseen data by evaluating its performance on the testing set using metrics like accuracy, precision, recall, and F1 score.

7. **Hyperparameter Tuning:**

   Adjust hyperparameters to improve the model's performance and prediction accuracy.

8. **Validation and Interpretation:**

   Interpret the model's predictions in light of healthcare decision-making.
   Verify the model's outcomes against accepted medical literature and guidelines.

Deliverables:

The project's goal is to produce a solid machine learning model that can forecast a person's risk of developing heart disease. The model's codebase, performance metrics, and methodology and conclusions documentation will be made available for additional study and possible implementation in a healthcare environment.

3.1.2 Problem Analysis

Analysing the issue of heart disease detection requires taking into account a number of different factors and difficulties specific to this area of study. This is a more thorough breakdown of the issue:

1. **Data Availability and Quality:**

   Obtaining a variety of annotated patient datasets can present testing and training challenges for the models.
   The data must be of a high quality and should include a variety of information, including test results and previous medical records..

2. **Health Significance:**

   Heart disease and other cardiovascular illnesses are a serious global health concern. In order to enable prompt interventions and individualised healthcare, the initiative addresses a fundamental need for early detection and risk assessment.

3. **Data Availability and Quality:**

   The dataset for the project is a CSV file. Accurate predictions depend on addressing missing values, ensuring data quality, and confirming the information's dependability.

4. **Data Exploration:**

   Exploratory Data Analysis (EDA) is essential to understanding the properties of the dataset. A well-informed approach to model construction involves recognising possible relationships and comprehending the distribution of features.

5. **Feature Selection:**

   Determining the most significant characteristics that impact the prognosis of cardiac disease is essential. Techniques for feature selection increase model efficiency by emphasising pertinent properties.

6. **Model Selection:**

   Selecting the right machine learning algorithms is crucial when working on categorization jobs. The chosen models ought to be able to handle the properties of the dataset and produce precise forecasts.

7. **Model Evaluation Metrics:**

   It is essential to use appropriate evaluation metrics (such as accuracy, precision, recall, and F1 score) in order to evaluate the model's performance. The metrics selected should be in line with the objectives of the project and the importance of false positives and false negatives in the healthcare industry.

8. **Ethical Considerations:**

   Ethics must be given first priority when handling sensitive health data. The project's success depends on protecting patient privacy, getting the required consents, and putting secure data practises in place.

9. **Interpretability of Results:**

   To acquire the confidence of medical professionals, the machine learning model's outcomes must be comprehensible. Comprehending the model's prediction process is essential to its acceptance and use in clinical environments.

10. **Hyperparameter Tuning:**

    Improving predicted accuracy necessitates optimising model hyperparameters. Tuning the hyperparameters systematically guarantees that the model is optimised for best results.

11. **Validation against Medical Guidelines:**

    It is necessary to verify the machine learning model's predictions against accepted medical practises and published research. To guarantee the accuracy of the forecasts, the project should be in line with the body of information already known in the field.

12. **Generalizability:**

    It is critical to evaluate how well the model generalises to new, untested data. The model's capacity to generate precise forecasts for a varied population outside of the training dataset is essential to the project's success.

13. **Scalability:**

    It is crucial to take the model's potential scalability into account, particularly if the project intends to be implemented more widely. A scalable model that can handle bigger datasets and a range of healthcare settings should be selected.

Through consideration of these factors throughout the problem analysis, the heart disease prediction project will be better equipped to handle obstacles, improve the project's precision, and offer insightful information to the medical community. The analysis serves as the cornerstone for a methodical and knowledgeable approach to resolving the current issue.

3.1.2 Solution

A multidisciplinary strategy including data collecting, feature engineering, model building, and ethical concerns is needed to address the problems with disease detection. Here are a few possible ways to deal with the difficulties in the illness detection project:

1. **Data Preprocessing:**

   Use techniques for imputation or removal to deal with missing data.

   Scale or normalise numerical features to provide reliable model output.

   Properly encode categorical variables to ensure compatibility with methods for machine learning.

2. **Exploratory Data Analysis (EDA):**

   Use EDA to comprehend the variable distribution.

   Determine the relationships between various characteristics and how they might affect heart disease.

   Illustrate important concepts for improved comprehension and dialogue.

3. **Feature Selection:**

   To determine which features are most important, apply methods like recursive feature removal, correlation analysis, or feature importance from tree-based models.

   Take domain expertise and medical literature into account to direct the selection procedure.

4. **Model Selection:**

   Play around with different classification techniques, like Gradient Boosting, Random Forest, Support Vector Machines, and Logistic Regression, that are appropriate for use in the healthcare industry.

   Analyse and contrast model performances by employing cross-validation methods.

5. **Model Training:**

   Divide the dataset into sets for testing and training.

   Utilising optimised hyperparameters, train a subset of machine learning models on the training dataset.

6. **Model Evaluation:**

   Use metrics such as accuracy, precision, recall, F1 score, and area under the receiver operating characteristic curve (AUC-ROC) to evaluate the models' performance on the testing set.

   Analyse data while taking into account the importance of false positives and false negatives in the context of predicting cardiac disease.

7. **Hyperparameter Tuning:**

   Use methods such as random or grid search to adjust the hyperparameters of particular models.

   Improve models so they can more effectively generalise to new data.

8. **Validation and Interpretation:**

   Check model outputs against accepted medical guidelines and literature.

   Explain model predictions with an emphasis on transparency and interpretability.

   Consult medical experts to get their opinions and validation.

9. **Ethical Considerations:**

   Put strong data privacy safeguards in place to safeguard patient data.

   Verify adherence to pertinent laws governing healthcare data and secure required consents.

   Openly explain to stakeholders how data is used and the implications of the model.

10. Scalability:

Take into account the model's potential scalability for wider application. Select models and technology that are able to manage more diverse healthcare contexts and greater datasets.

11. Documentation:

Record all aspects of the process, such as the data pretreatment procedures, model selection standards, specifics of hyperparameter tuning, and metrics for the model evaluation.

Provide a thorough manual so that the project can be replicated and referred to in the future.

12. Communication and Deployment:

Effectively convey the findings to stakeholders, both technical and non-technical. If appropriate, think about implementing the model in a healthcare setting and making sure that it integrates seamlessly with the current clinical workflows.

Through a methodical approach to these stages, the Heart Disease Prediction Project can create a dependable and comprehensible model, offering significant contributions to the healthcare industry and facilitating the early identification and management of heart disease risk factors in individuals.

3.2 Requirements

3.2.1 Functional Requirements

Functional requirements for disease detection typically include:

1. Data Input: Patient health data from external sources, such as CSV files or data entry interfaces, should be able to be entered into the system.

2. Data Preprocessing: To manage missing values, normalise numerical features, and encode categorical variables, implement data pretreatment functionalities.
3. Exploratory Data Analysis (EDA): To comprehend feature distribution, find relationships, and visualise important insights from the dataset, do exploratory data analysis.

4. Feature Selection: Use feature selection strategies to determine which characteristics are most important for predicting heart disease.

5. Model Selection: Use a variety of machine learning models, including Gradient Boosting, Random Forest, Support Vector Machines, and Logistic Regression, to predict cardiac disease.

6. Model Training: Utilising a training dataset created from the input data, train a chosen set of machine learning models.

7. Model Evaluation: Using metrics such as accuracy, precision, recall, F1 score, and AUC-ROC, assess how well trained models perform on a different testing dataset.

8. Hyperparameter Tuning: Optimise the chosen machine learning models through hyperparameter adjustment for improved generalisation.

9. Ethical Considerations: Include elements that address ethical issues, like consent management and data anonymization methods.

10. Interpretability: Offer tools for analysing model predictions, such as justifications for specific forecasts.

11. User Feedback and Usability: Provide tools for conducting usability tests, getting user input, and iteratively refining the user interface in response to advice from medical experts.

12. Integration with Clinical Workflows: Provide a system that is easy to integrate with current clinical workflows to minimise interference with healthcare procedures.

13. Scalability: Make sure the system can grow to handle bigger datasets and future additions to cover a wider spectrum of illnesses.

14. Comparative Analysis: Provide features for evaluating the produced model's performance against current diagnostic techniques or instruments.

15. Validation Against Medical Guidelines: Add functionality to validate model outputs against accepted medical recommendations and literature.

3.2.2 Non-Functional Requirements

Non-functional requirements are as significant for disease detection because they specify the system's performance in areas other than functionality. These are a few non- functional needs:

1. Performance: Effective and timely forecasts should be provided by the system, particularly in real-time clinical contexts.

2. Accuracy: Based on pertinent health parameters, the model should be highly accurate in forecasting the onset of heart disease.

3. Privacy and Security: Strict privacy regulations must be followed by the system to guarantee the security and privacy of patient health data.

4. Interpretability: Put a high priority on model interpretability to win over medical experts and promote better decision-making.

5. Usability: Healthcare personnel with different levels of technological competence should be able to quickly navigate and use the user interface. It should be simple to use and intuitive.
6. Scalability: The system ought to be expandable to accommodate future growth to accommodate more diseases and a growing amount of patient data.

7. Reliability: In order to guarantee constant availability in clinical settings, the system should be dependable and have little downtime.

8. Ethical Considerations: Respect the law regarding data protection, follow ethical guidelines while managing data, and make sure consent is given voluntarily.

9. Adaptability: The system must to be flexible enough to accommodate modifications in medical procedures, laws, and technology breakthroughs.

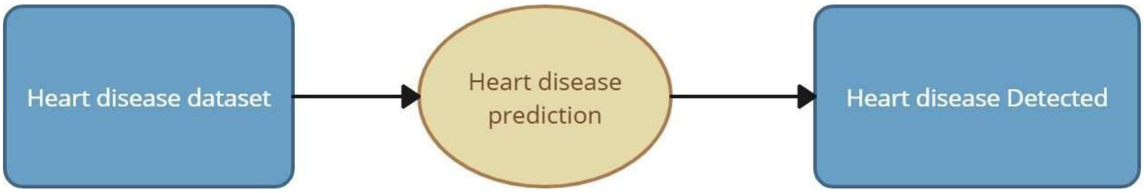
10. Documentation: There should be extensive documentation available, such as a user manual, technical specs, and an explanation on how to understand the results of the model.

11. Collaboration: Throughout the course of the project, encourage cooperation between data scientists, medical practitioners, and other stakeholders.

12. Compliance: Make sure that all rules and guidelines pertaining to the healthcare sector are followed, especially those concerning patient confidentiality and data security.

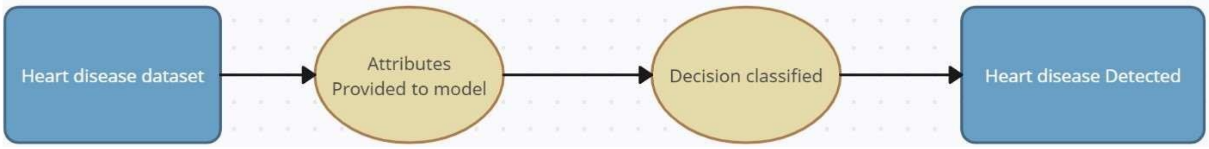
3.3 Data-Flow Diagram (DFD)

1. Level 0 DFD: A high-level overview of the entire system is given by Level 0 DFD, which displays the dataset, external entities (heart disease detection), and data flow between the various processes. It serves as a foundation for lower-level DFDs with greater complexity, in which the primary process is broken down into more specific subprocesses.



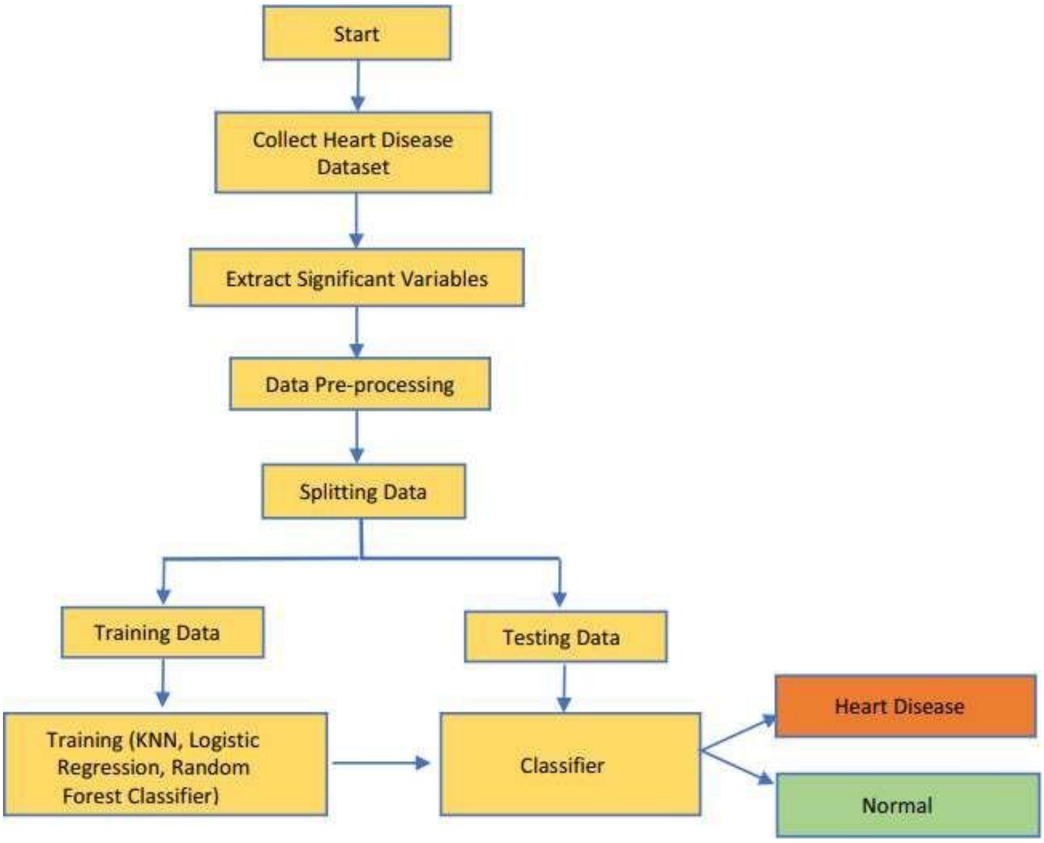
Level 0 DFD

2. Level 1 DFD: By dividing the primary process from the Level 0 DFD into smaller processes—providing data inputs to the model and decision categorization carried out by the model—Level 1 DFD offers a more thorough understanding of the system. It aids in comprehending the many roles played by the system and their interactions. At lower levels, this decomposition process can be carried out to produce even more thorough DFDs.



Level 1 DFD

3. Level 2 DFD: By dissecting the subprocesses from the Level 1 DFD into sub- subprocesses—feature extraction, model training, and testing—Level 2 DFD offers a more detailed picture of the system. Until a sufficient degree of information is obtained, this hierarchical decomposition is carried out, enabling a thorough comprehension of the system's operations and their interactions.



Start
Collect Heart Disease Dataset
Extract Significant Variables
Data Pre-processing
Splitting Data
Training Data
Testing Data
Training (KNN, Logistic Regression, Random Forest Classifier)
Classifier
Heart Disease
Normal

Level 2 DFD

Chapter 04: Implementation

4.1 Dataset Used

4.1.1 Dataset Overview

The dataset contains 1025 samples with 14 attributes listed below.

4.1.2 Dataset Features

0. **age**: Age of the individual.
1. **sex**: Gender of the

   individual. 1 = Male

   0 = Female
2. **cp** (Chest-Pain Type): Type of chest pain experienced.

   0 = Typical angina

   1 = Atypical angina

   2 = Non-anginal pain

   3 = Asymptomatic
3. **trestbps** (Resting Blood Pressure): Resting blood pressure in mmHg.
4. **chol** (Serum Cholesterol): Serum cholesterol in mg/dl.
5. **fbs** (Fasting Blood Sugar): Fasting blood sugar compared to 120mg/dl.
6. **restecg** (Resting ECG): Resting electrocardiographic results.

   0 = Normal

   1 = ST-T wave abnormality

   2 = Left ventricular hypertrophy
7. **thalach** (Max Heart Rate Achieved): Max heart rate achieved.
8. **exang** (Exercise Induced Angina): Exercise-induced angina.

   1 = Yes

   0 = No

9. **oldpeak** (ST Depression Induced by Exercise Relative to Rest): Value of ST depression induced by exercise relative to rest.

10. **slope** (Peak Exercise ST Segment): Peak exercise ST segment.

    0 = Upsloping

    1 = Flat

    2 = Down sloping

11. **ca** (Number of Major Vessels Colored by Fluoroscopy): Number of major vessels (0-3) colored by fluoroscopy.

12. **thal** (Thalassemia): Thalassemia condition.

    0 = Normal

    1 = Fixed defect

    2 = Reversible defect

13. **target** (Diagnosis of Heart Disease): Presence or absence of heart disease. 0 = Absence

    1 = Presence

4.2 Algorithm / Pseudo code of the Project Problem

- **Step 1: Data Preparation**

```
Load dataset
Preprocess data (cleaning, handling missing values, feature scaling, etc.)
Split data into training and testing sets
```

- **Step 2: Define Decision Tree Algorithm**

```
Function DecisionTree:
    if stopping_condition: # Check if a stopping condition is met
    return leaf_node_prediction
```

```
else:
    Select best attribute to split on
    Split data based on the best
    attribute Create a node for the
    best attribute For each split:
        Recursively call DecisionTree on the split data
        Attach the result as a child node to the current
        node
```

- **Step 3: Training the Model**

```
Call DecisionTree function with training data
```

- **Step 4: Predictions**

```
Function Predict:
      For each data point in the testing set:
 Traverse the decision tree using the features of the data point. Predict the class
label at the leaf node reached
```

- **Step 5: Evaluate the Model**

```
Evaluate predictions using metrics (e.g., accuracy, precision, recall, F1-score)
```

- **Step 6: Make Predictions for New Data**

```
Function Make Prediction:
    Given new data:
    Traverse the trained decision tree using the features of the new data
    Output the predicted class label
```

- **Step 7: Save and Deploy the Model**

```
Save the trained decision tree model
```

Deploy the model for future predictions

- **Step 8: Model Tuning and Improvement**

  (Optional) Fine-tune hyperparameters

  Consider feature selection or engineering for better performance

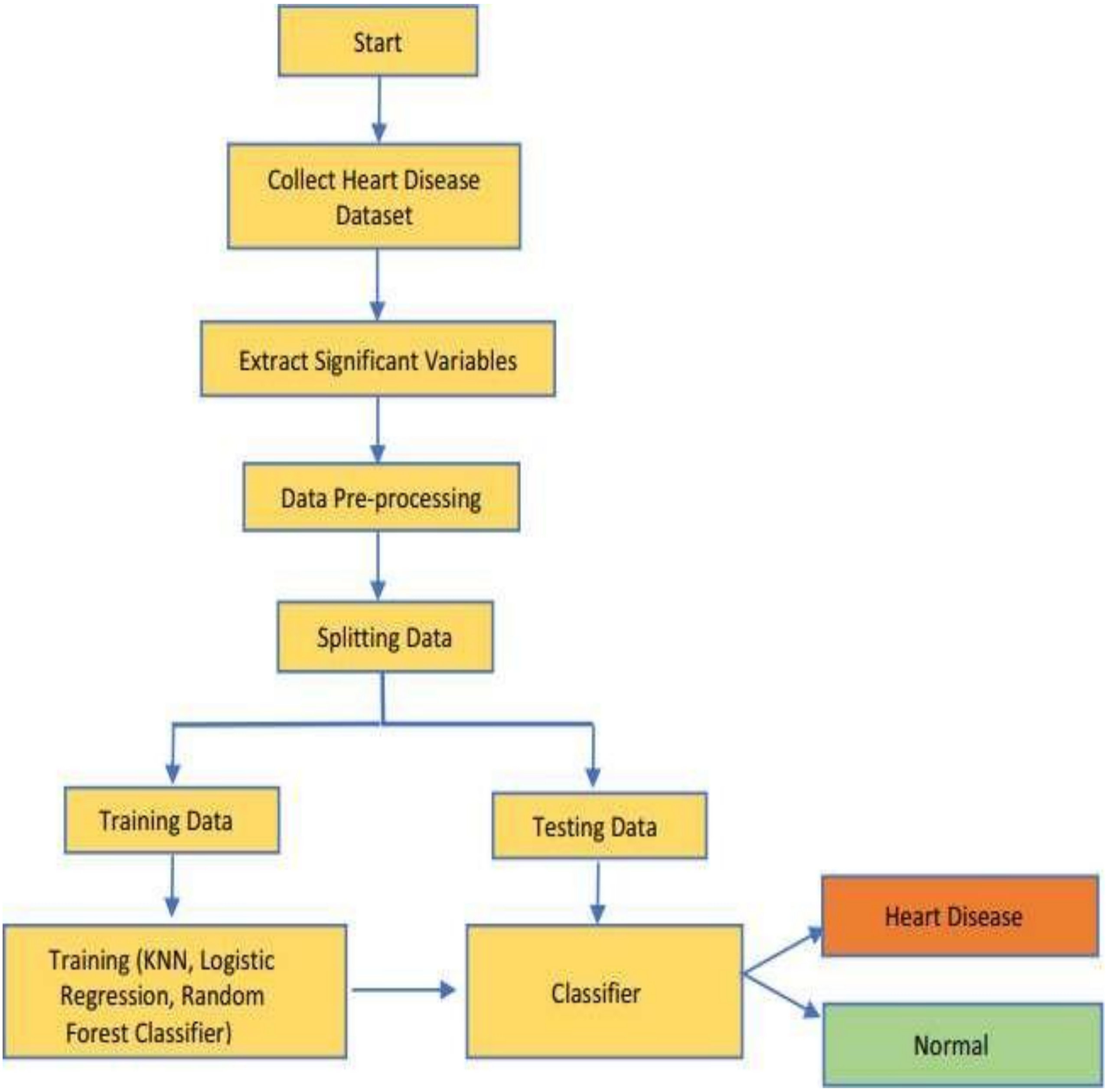
- **Step 9: Android Application for Mobile**

  Using Java and Kotlin And python

  as a back incorporated final

  implement.

4.3 Flow graph


Start
Collect Heart Disease Dataset
Extract Significant Variables
Data Pre-processing
Splitting Data
Training Data
Testing Data
Training (KNN, Logistic Regression, Random Forest Classifier)
Classifier
Heart Disease
Normal

Flow Graph of Major Project Problem

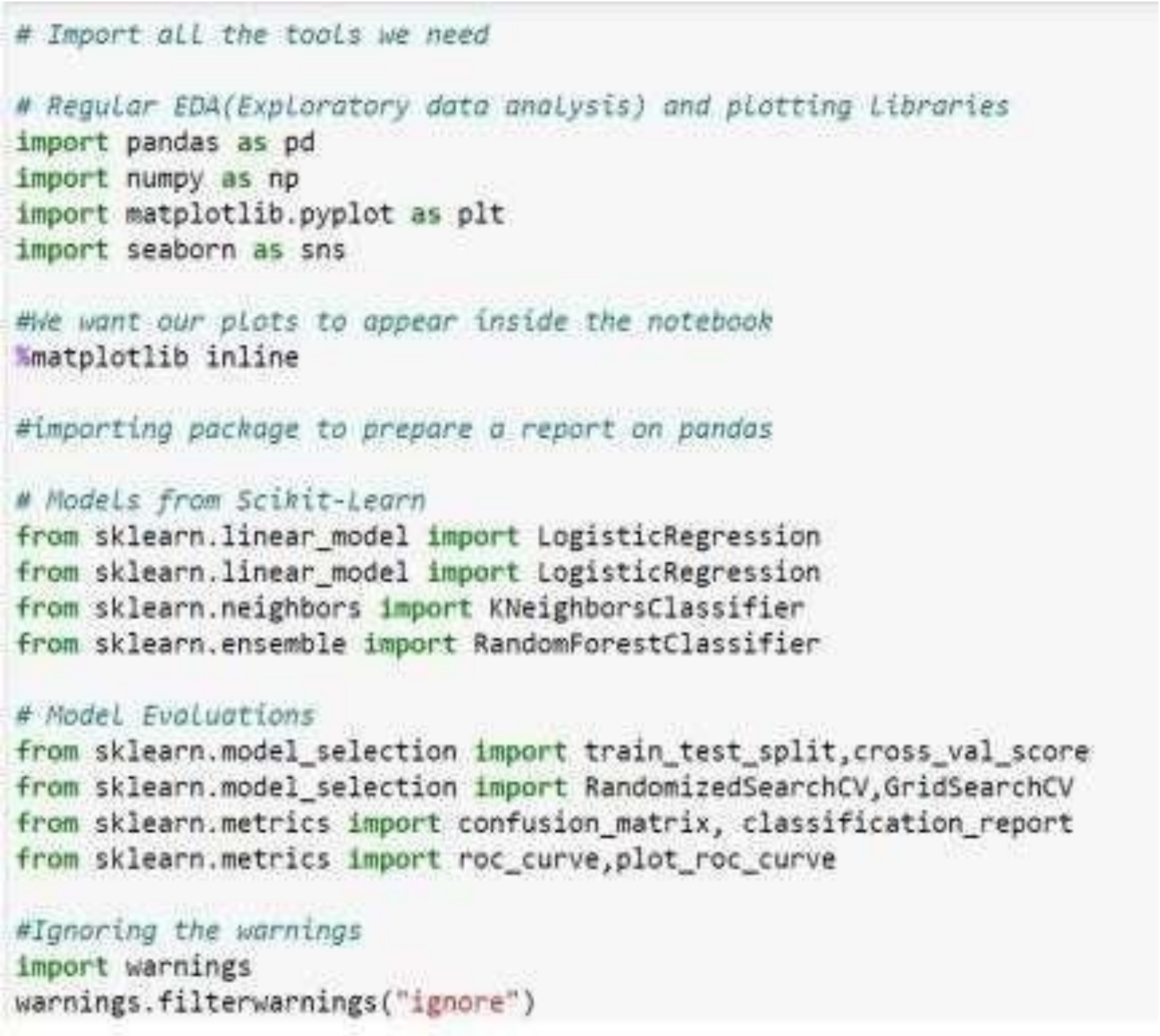
4.4 Screen shots of the various stages of the Project

STEP 1: DATA PREPARATION AND ITS TOOLS

Pandas & Numpy for Data Analysis and Manipulation

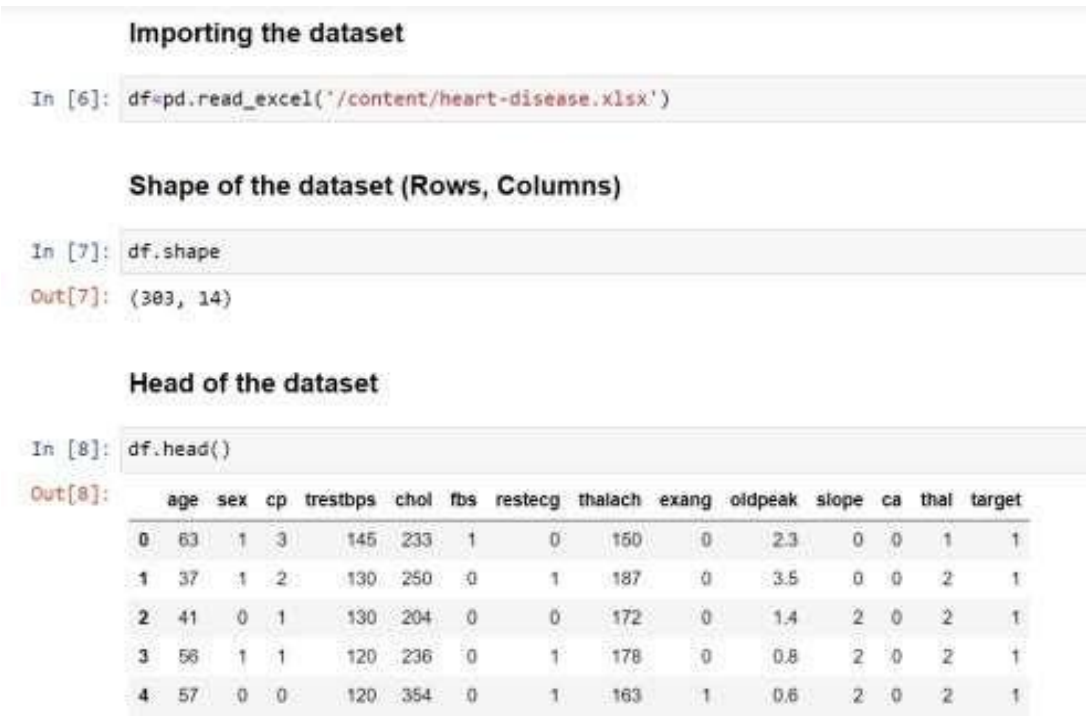
Matplotlib and Seaborn for Data Visualisation

Scikit-Learn for the Modelling and Evaluation



```
# Import all the tools we need

# Regular EDA(Exploratory data analysis) and plotting Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

#We want our plots to appear inside the notebook
%matplotlib inline

#importing package to prepare a report on pandas

# Models from Scikit-Learn
from sklearn.linear_model import LogisticRegression
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier

# Model Evaluations
from sklearn.model_selection import train_test_split,cross_val_score
from sklearn.model_selection import RandomizedSearchCV,GridSearchCV
from sklearn.metrics import confusion_matrix, classification_report
from sklearn.metrics import roc_curve,plot_roc_curve

#Ignoring the warnings
import warnings
warnings.filterwarnings("ignore")
```

Figure 1: Necessary Imports



Importing the dataset

```
In [6]: df=pd.read_excel('/content/heart-disease.xlsx')
```

Shape of the dataset (Rows, Columns)

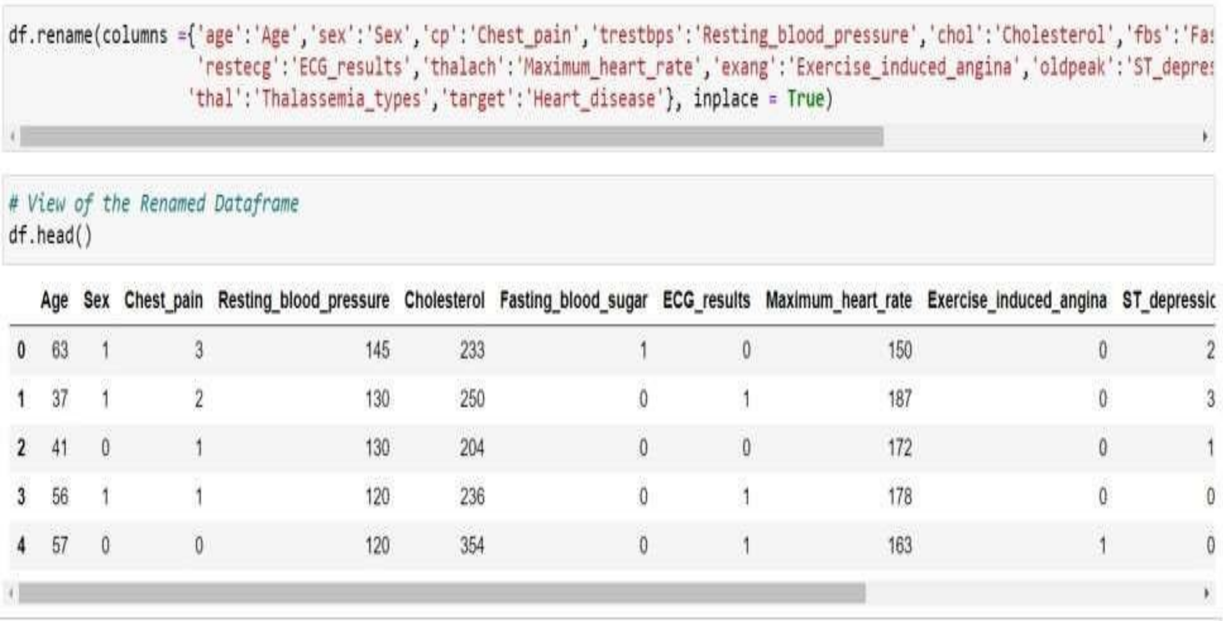
```
In [7]: df.shape
Out[7]: (303, 14)
```

Head of the dataset

```
In [8]: df.head()
Out[8]:
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

Figure 2: Importing the dataset from the local file, then Counting the total number of rows and columns and finally returns a specified number of rows, string from the top.



```
df.rename(columns ={'age':'Age','sex':'Sex','cp':'Chest_pain','trestbps':'Resting_blood_pressure','chol':'Cholesterol','fbs':'Fa
                    'restecg':'ECG_results','thalach':'Maximum_heart_rate','exang':'Exercise_induced_angina','oldpeak':'ST_depre
                   'thal':'Thalassemia_types','target':'Heart_disease'}, inplace = True)
```

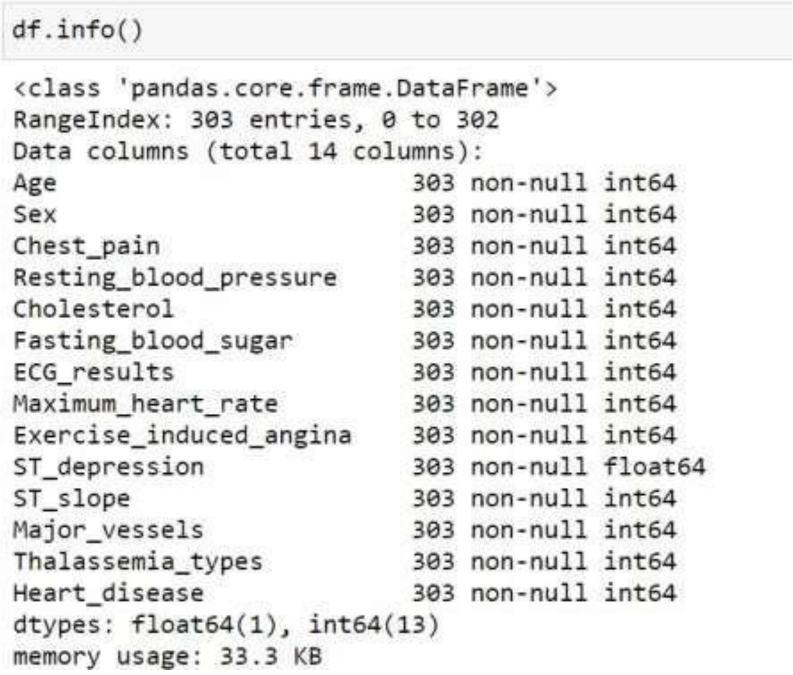
```
# View of the Renamed Dataframe
df.head()
```

	Age	Sex	Chest_pain	Resting_blood_pressure	Cholesterol	Fasting_blood_sugar	ECG_results	Maximum_heart_rate	Exercise_induced_angina	ST_depressic
0	63	1	3	145	233	1	0	150	0	2
1	37	1	2	130	250	0	1	187	0	3
2	41	0	1	130	204	0	0	172	0	1
3	56	1	1	120	236	0	1	178	0	0
4	57	0	0	120	354	0	1	163	1	0

Figure 3: For Better understanding remaining columns of the dataset used in our code.

STEP 2: EXPLORATORY DATA ANALYSIS

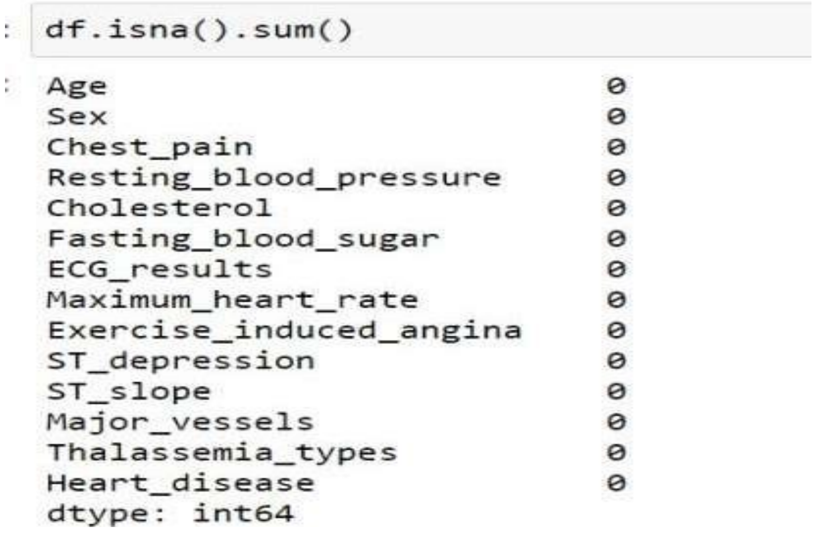
conducting exploratory data analysis (EDA) helps understand the dataset's characteristics, distributions, relationships, and potential insights.



```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
Age                        303 non-null int64
Sex                        303 non-null int64
Chest_pain                 303 non-null int64
Resting_blood_pressure     303 non-null int64
Cholesterol                303 non-null int64
Fasting_blood_sugar        303 non-null int64
ECG_results                303 non-null int64
Maximum_heart_rate         303 non-null int64
Exercise_induced_angina    303 non-null int64
ST_depression              303 non-null float64
ST_slope                   303 non-null int64
Major_vessels              303 non-null int64
Thalassemia_types          303 non-null int64
Heart_disease              303 non-null int64
dtypes: float64(1), int64(13)
memory usage: 33.3 KB
```

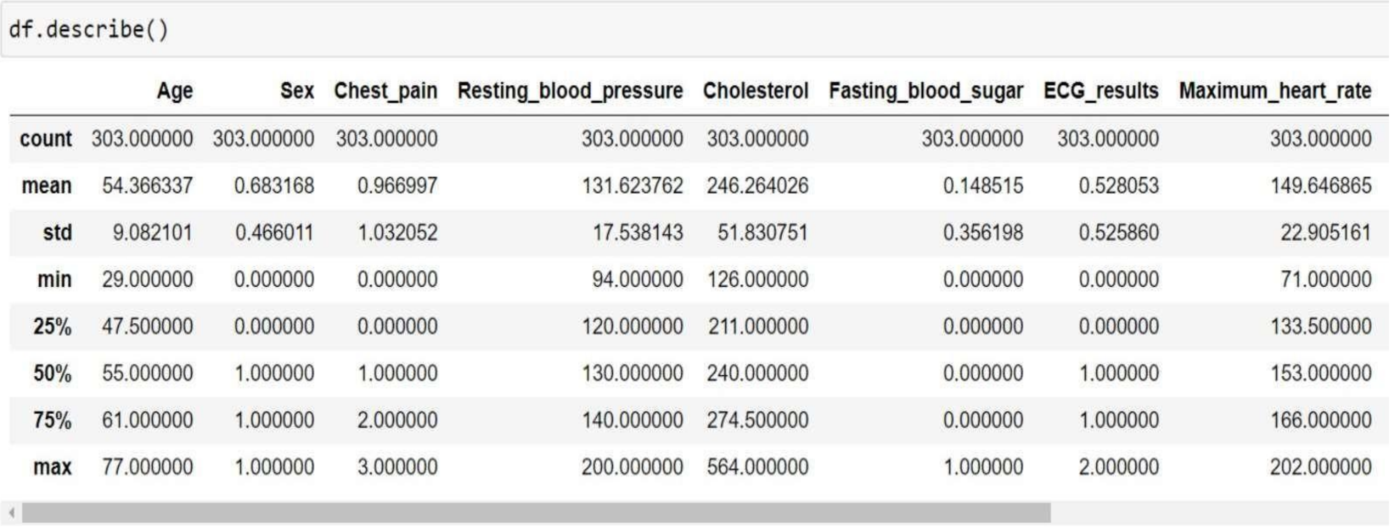
Figure 4: Information about the data



```
df.isna().sum()
```

```
Age                        0
Sex                        0
Chest_pain                 0
Resting_blood_pressure     0
Cholesterol                0
Fasting_blood_sugar        0
ECG_results                0
Maximum_heart_rate         0
Exercise_induced_angina    0
ST_depression              0
ST_slope                   0
Major_vessels              0
Thalassemia_types          0
Heart_disease              0
dtype: int64
```

Figure 5: Checking is there any field left empty?



```
df.describe()
```

```
              Age         Sex  Chest_pain  Resting_blood_pressure  Cholesterol  Fasting_blood_sugar  ECG_results  Maximum_heart_rate
count  303.000000  303.000000  303.000000              303.000000   303.000000           303.000000   303.000000          303.000000
mean    54.366337    0.683168    0.966997              131.623762   246.264026             0.148515     0.528053          149.646865
std      9.082101    0.466011    1.032052               17.538143    51.830751             0.356198     0.525860           22.905161
min     29.000000    0.000000    0.000000               94.000000   126.000000             0.000000     0.000000           71.000000
25%     47.500000    0.000000    0.000000              120.000000   211.000000             0.000000     0.000000          133.500000
50%     55.000000    1.000000    1.000000              130.000000   240.000000             0.000000     1.000000          153.000000
75%     61.000000    1.000000    2.000000              140.000000   274.500000             0.000000     1.000000          166.000000
max     77.000000    1.000000    3.000000              200.000000   564.000000             1.000000     2.000000          202.000000
```

Figure 6: Getting Description about the Dataset.

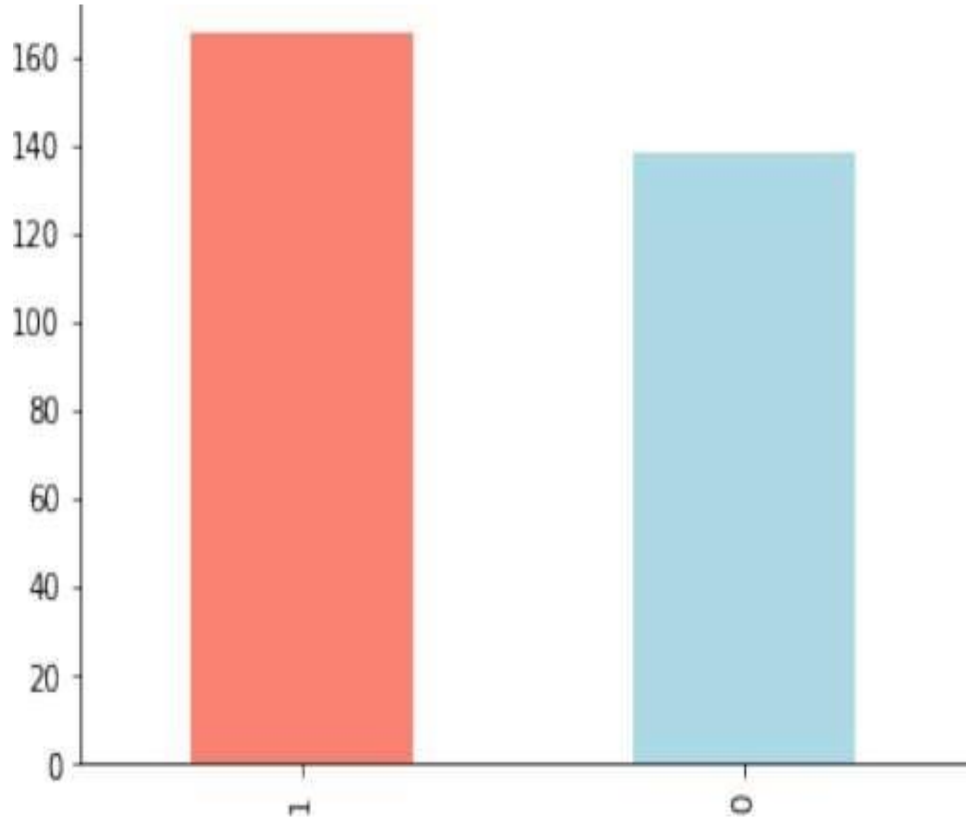
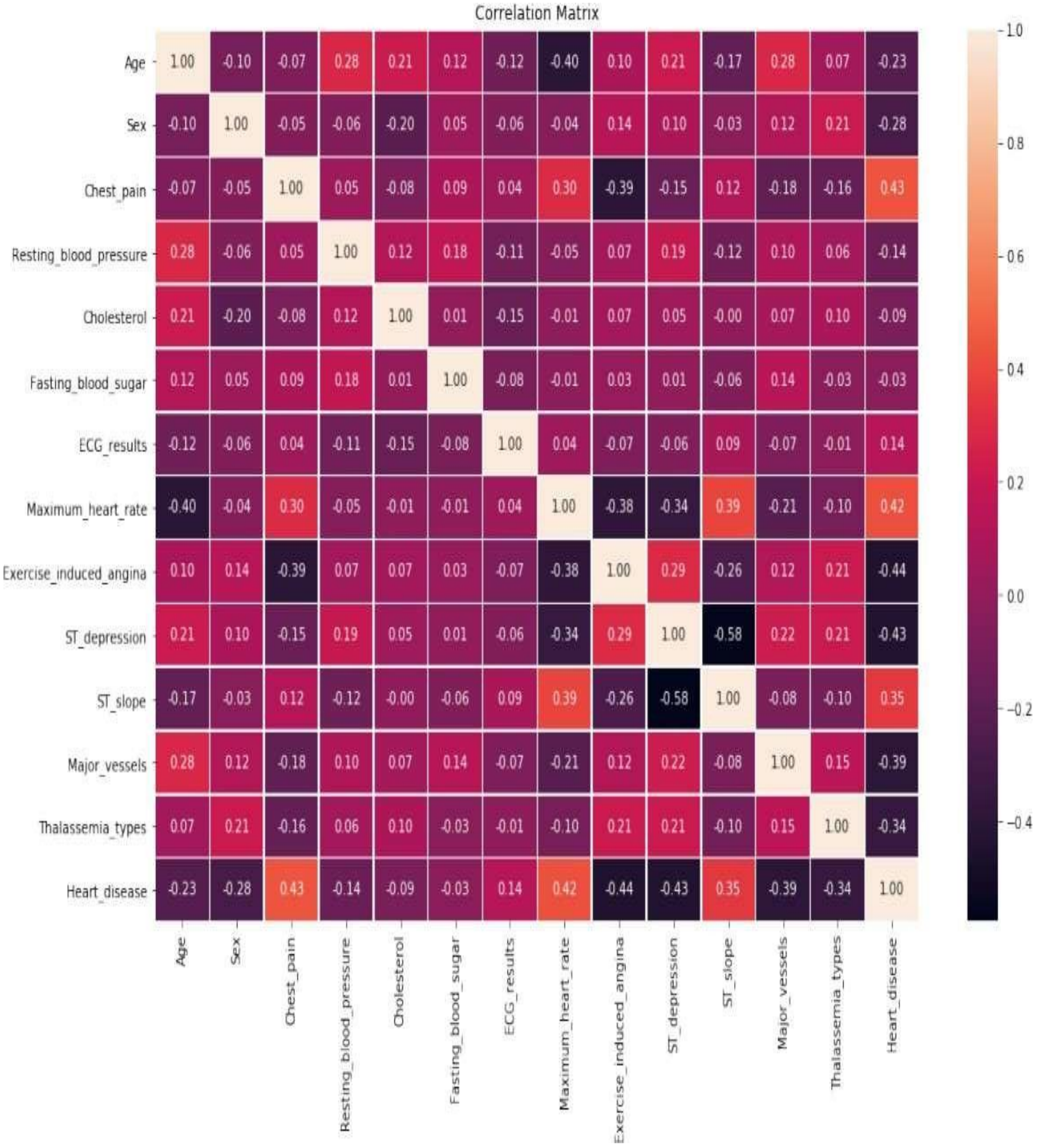


Figure 7: Count of the Diagnosis of Heart Disease

Male (1) and Female (0) respectively.



	Age	Sex	Chest_pain	Resting_blood_pressure	Cholesterol	Fasting_blood_sugar	ECG_results	Maximum_heart_rate	Exercise_induced_angina	ST_depression	ST_slope	Major_vessels	Thalassemia_types	Heart_disease
Age	1.00	-0.10	-0.07	0.28	0.21	0.12	-0.12	-0.40	0.10	0.21	-0.17	0.28	0.07	-0.23
Sex	-0.10	1.00	-0.05	-0.06	-0.20	0.05	-0.06	-0.04	0.14	0.10	-0.03	0.12	0.21	-0.28
Chest_pain	-0.07	-0.05	1.00	0.05	-0.08	0.09	0.04	0.30	-0.39	-0.15	0.12	-0.18	-0.16	0.43
Resting_blood_pressure	0.28	-0.06	0.05	1.00	0.12	0.18	-0.11	-0.05	0.07	0.19	-0.12	0.10	0.06	-0.14
Cholesterol	0.21	-0.20	-0.08	0.12	1.00	0.01	-0.15	-0.01	0.07	0.05	-0.00	0.07	0.10	-0.09
Fasting_blood_sugar	0.12	0.05	0.09	0.18	0.01	1.00	-0.08	-0.01	0.03	0.01	-0.06	0.14	-0.03	-0.03
ECG_results	-0.12	-0.06	0.04	-0.11	-0.15	-0.08	1.00	0.04	-0.07	-0.06	0.09	-0.07	-0.01	0.14
Maximum_heart_rate	-0.40	-0.04	0.30	-0.05	-0.01	-0.01	0.04	1.00	-0.38	-0.34	0.39	-0.21	-0.10	0.42
Exercise_induced_angina	0.10	0.14	-0.39	0.07	0.07	0.03	-0.07	-0.38	1.00	0.29	-0.26	0.12	0.21	-0.44
ST_depression	0.21	0.10	-0.15	0.19	0.05	0.01	-0.06	-0.34	0.29	1.00	-0.58	0.22	0.21	-0.43
ST_slope	-0.17	-0.03	0.12	-0.12	-0.00	-0.06	0.09	0.39	-0.26	-0.58	1.00	-0.08	-0.10	0.35
Major_vessels	0.28	0.12	-0.18	0.10	0.07	0.14	-0.07	-0.21	0.12	0.22	-0.08	1.00	0.15	-0.39
Thalassemia_types	0.07	0.21	-0.16	0.06	0.10	-0.03	-0.01	-0.10	0.21	0.21	-0.10	0.15	1.00	-0.34
Heart_disease	-0.23	-0.28	0.43	-0.14	-0.09	-0.03	0.14	0.42	-0.44	-0.43	0.35	-0.39	-0.34	1.00

Figure 8: Correlation Matrix

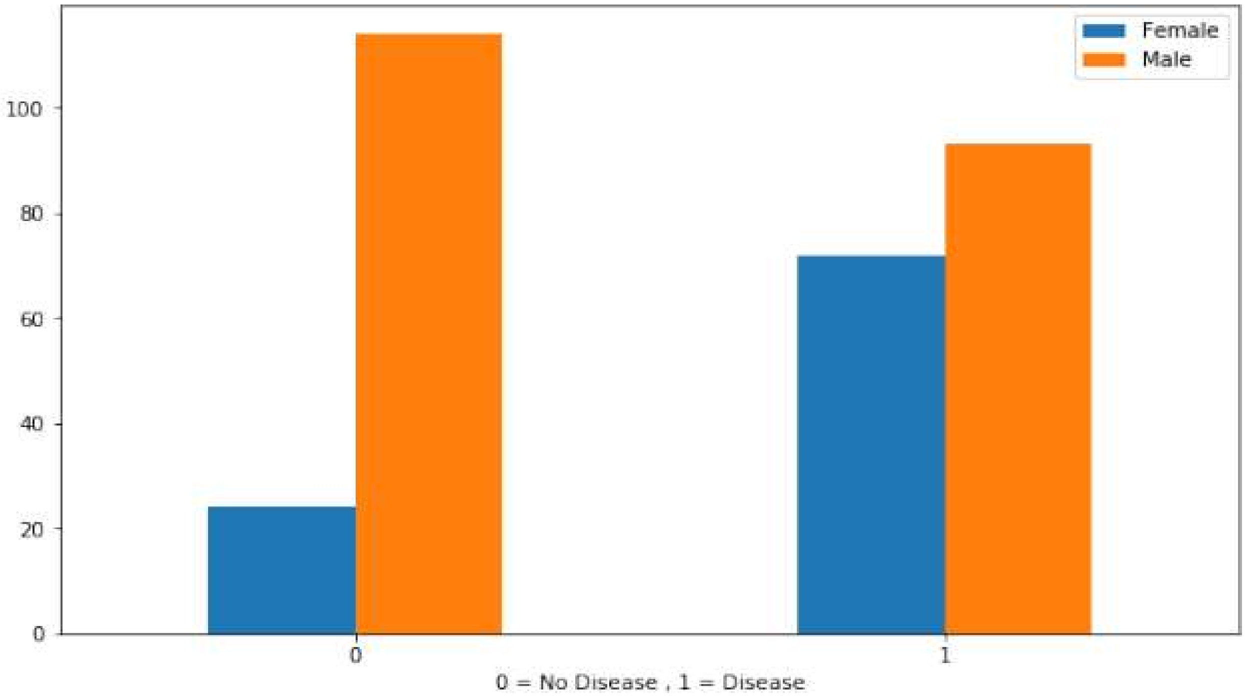


Figure 9: Heart Disease Frequency vs Sex

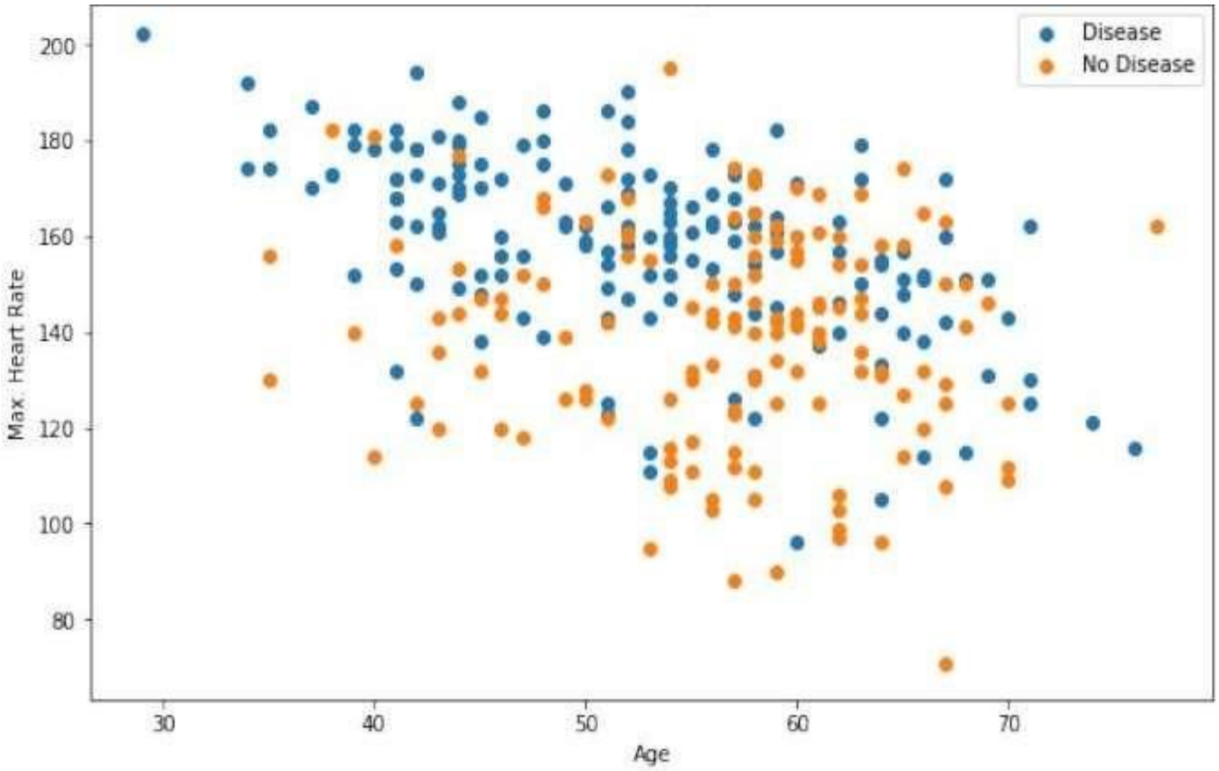


Figure 10: Heart Disease in function of Age and Max Heart Rate

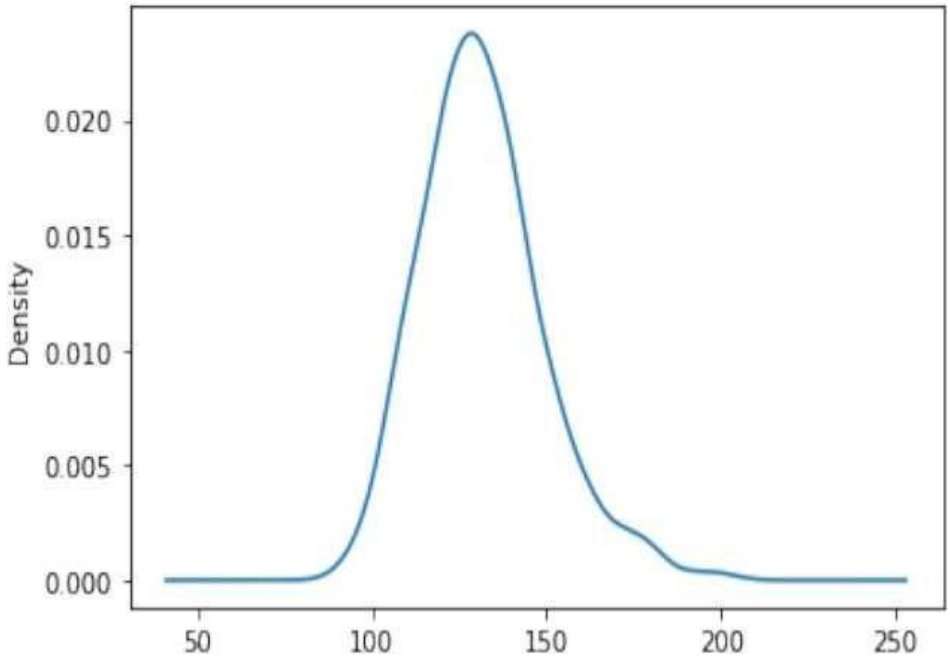


Figure 11: Resting Blood Pressure

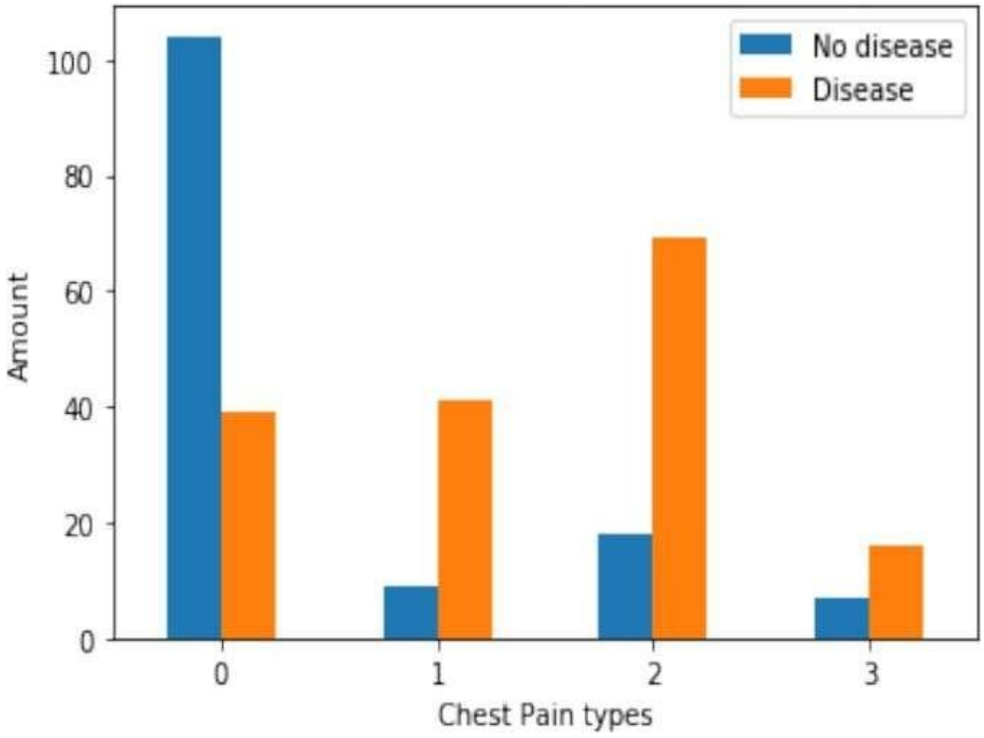


Figure 12: Heart Disease Frequency per Chest Pain Type

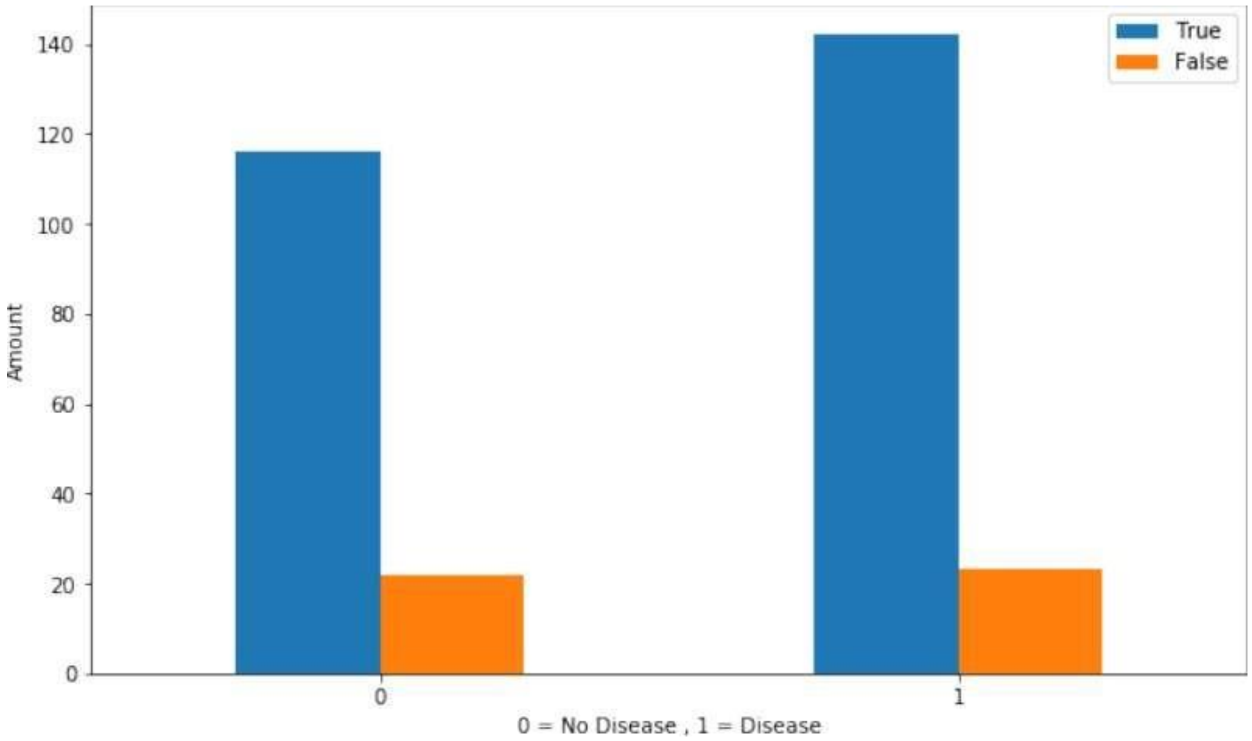


Figure 13: Heart Disease Frequency vs Fasting Blood Sugar

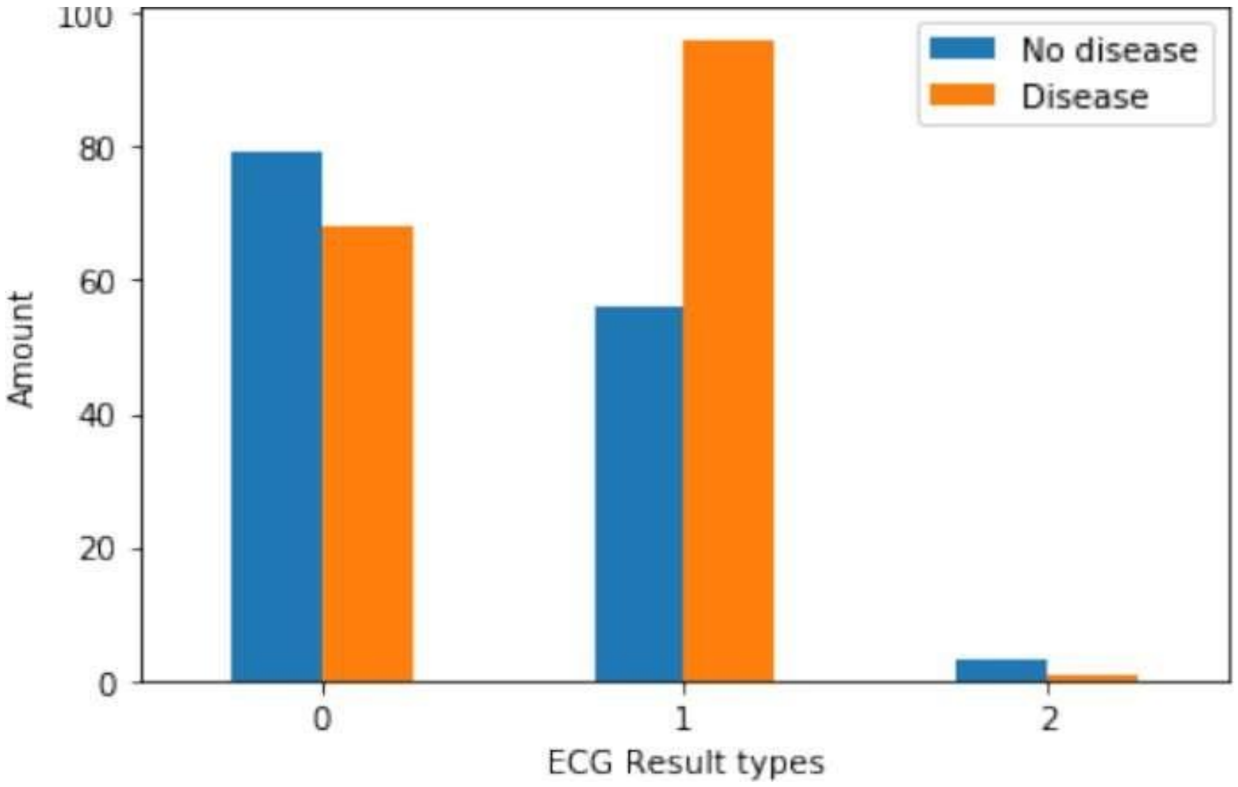


Figure 14: Heart Disease Frequency per ECG Results

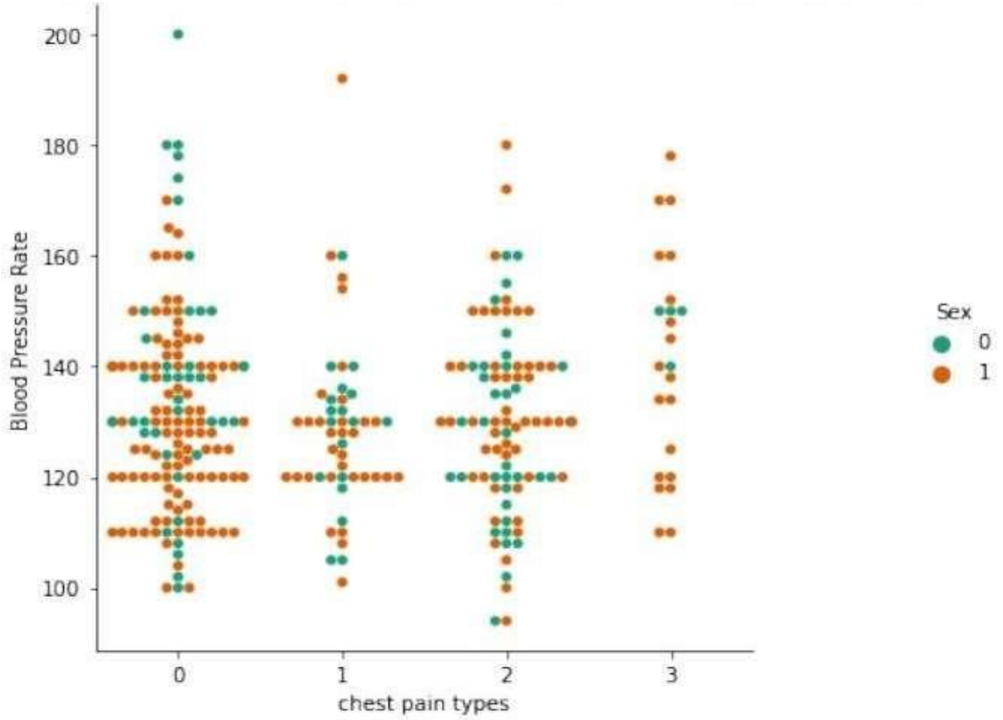


Figure 15: Relation Between resting chest pain types and Blood Pressure Rate with respect to gender.

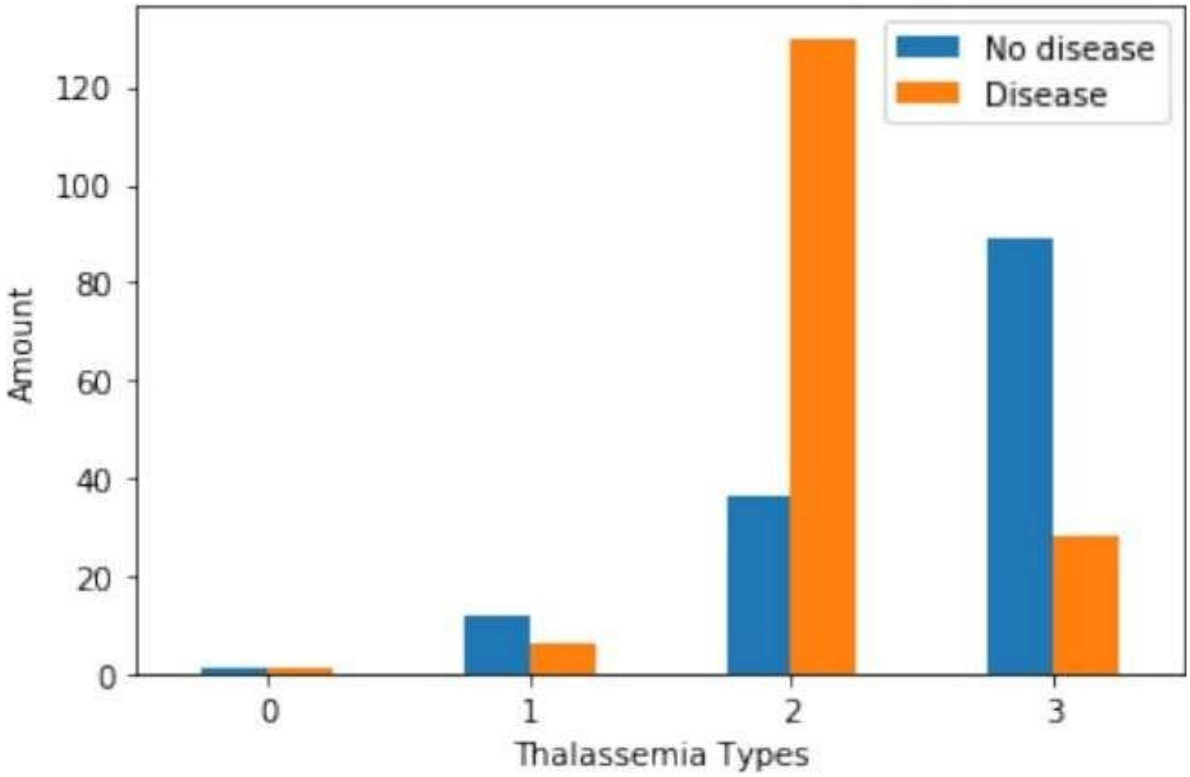


Figure 16: Heart Disease Frequency per Thalassemia Types

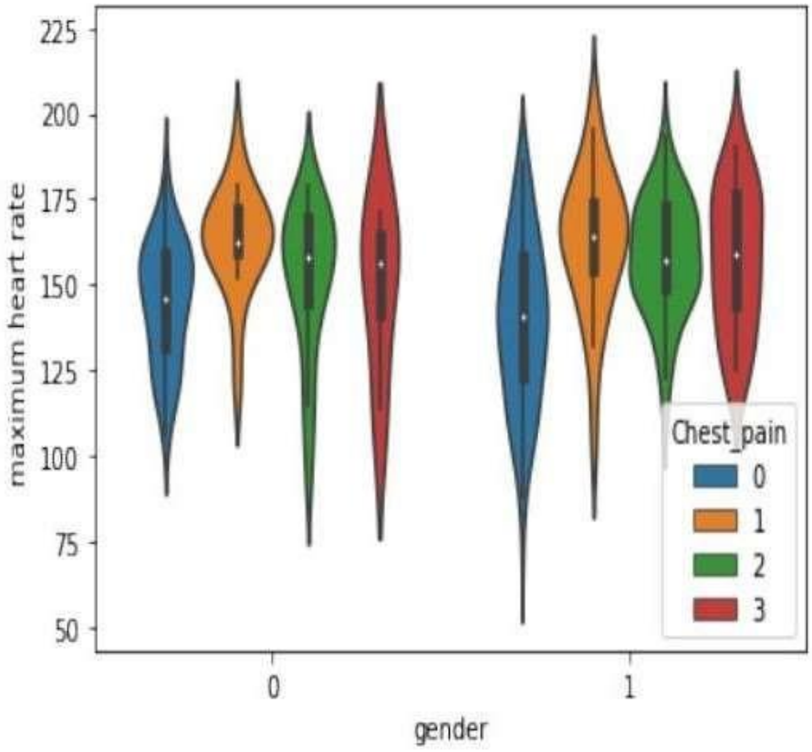


Figure 17: Relation Between maximum heart rate and chest pain types with respect to gender

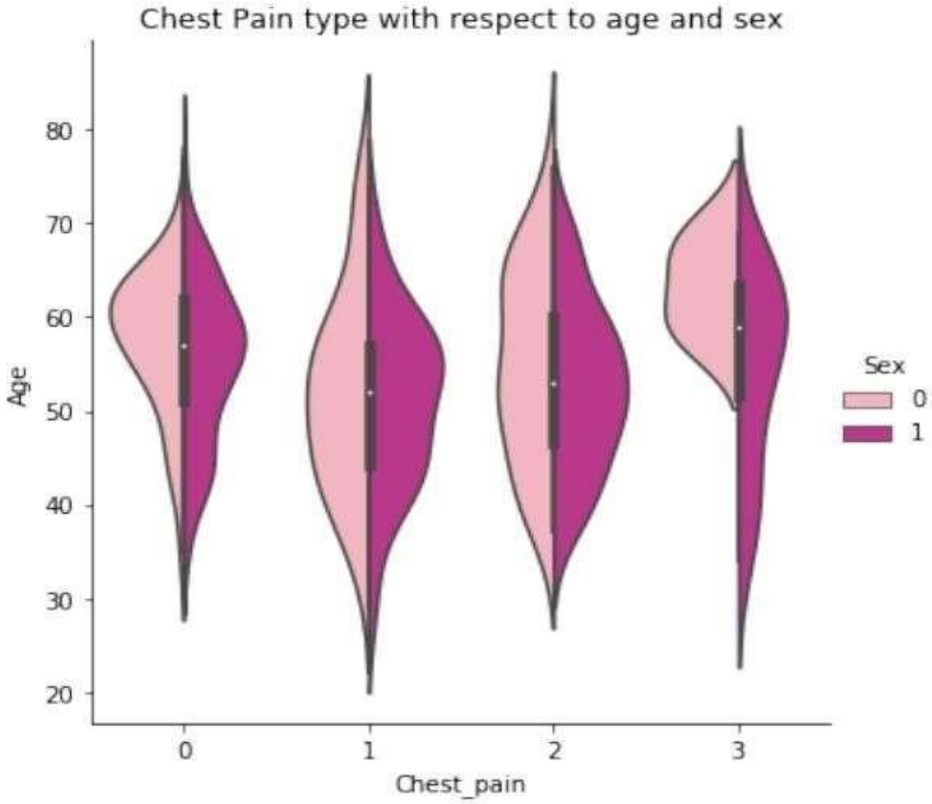


Figure 18: Chest Pain type with respect to age and sex

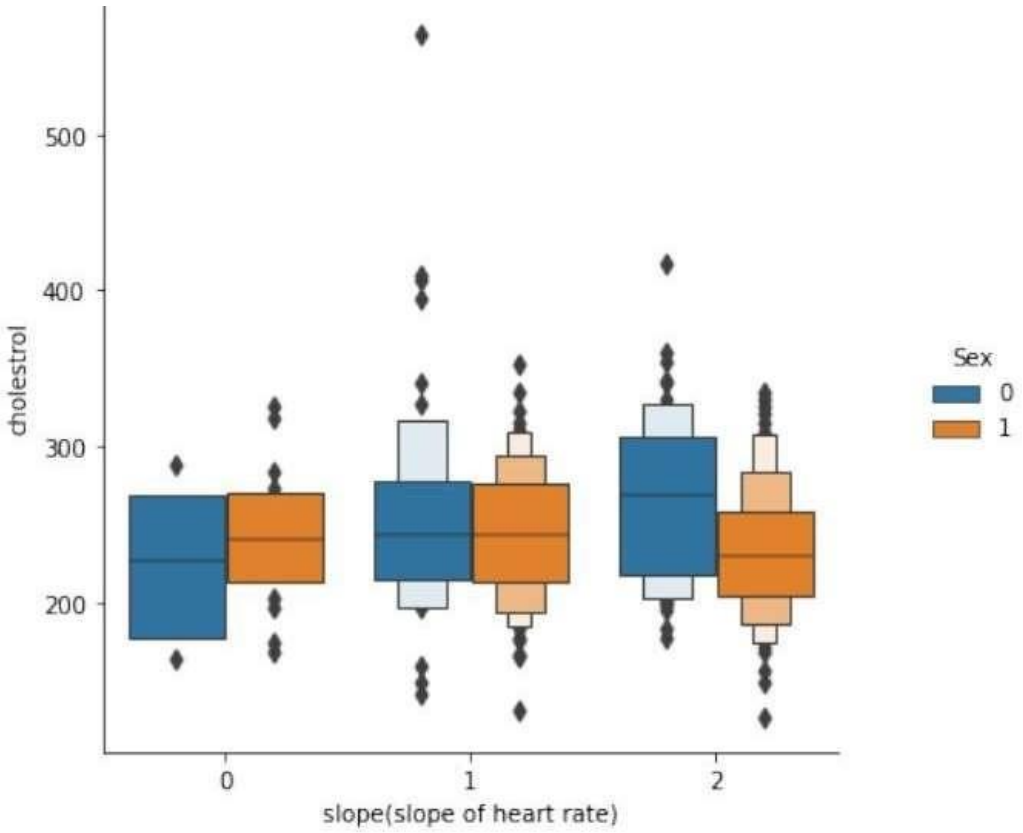
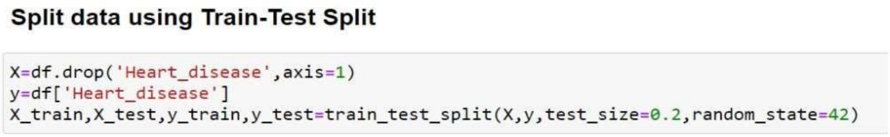


Figure 19: Relation Between heart rate slope and cholesterol level with respect to gender

STEP 3: MODELLING

We will experiment with the models, trying 3 different models and getting the results from them and comparing them later



Split data using Train-Test Split

```
X=df.drop('Heart_disease',axis=1)
y=df['Heart_disease']
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=42)
```

Now we have got our data split into training and test sets, it is time to build a Machine Learning model.

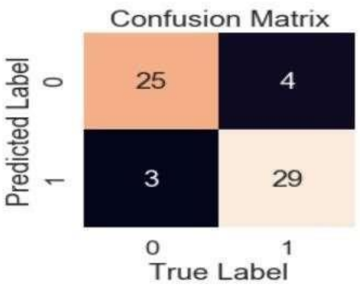
We will train it (find the patterns) on the training set.

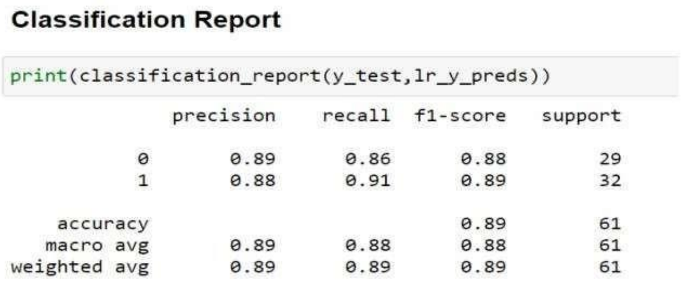
And we will test it (use the patterns) on the test set.

We're going to try 3 different Machine Learning models:

1. Logistic Regression
2. K-Nearest Neighbours Classifier
3. Random Forest Classifier

1. Logistic Regression (Accuracy of 88.5%)



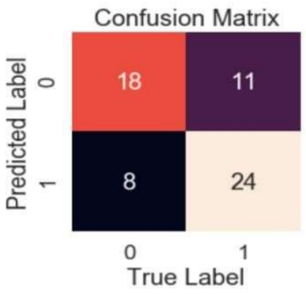


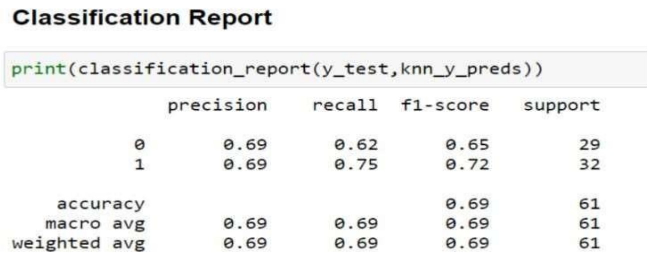
Classification Report

```
print(classification_report(y_test,lr_y_preds))
```

```
              precision    recall  f1-score   support

           0       0.89      0.86      0.88        29
           1       0.88      0.91      0.89        32

    accuracy                           0.89        61
   macro avg       0.89      0.88      0.88        61
weighted avg       0.89      0.89      0.89        61
```

2. K-Nearest Neighbour Classifier (Accuracy of 68.8%)



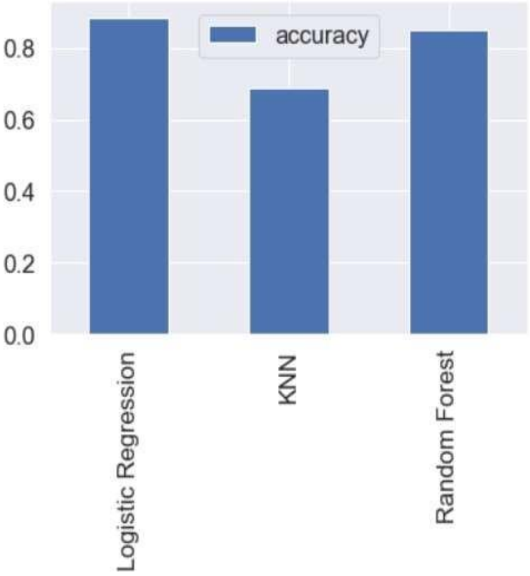


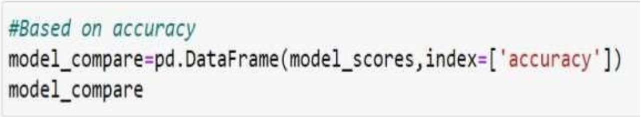
Classification Report

```
print(classification_report(y_test,knn_y_preds))
```

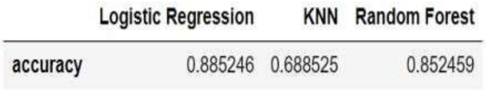
```
              precision    recall  f1-score   support

           0       0.69      0.62      0.65        29
           1       0.69      0.75      0.72        32

    accuracy                           0.69        61
   macro avg       0.69      0.69      0.69        61
weighted avg       0.69      0.69      0.69        61
```

Comparison Among Models



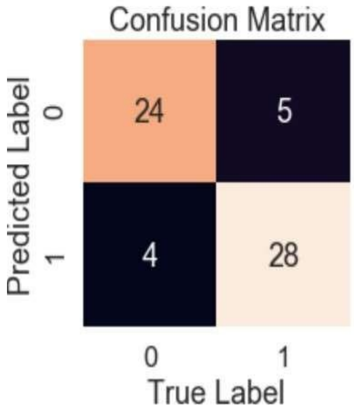


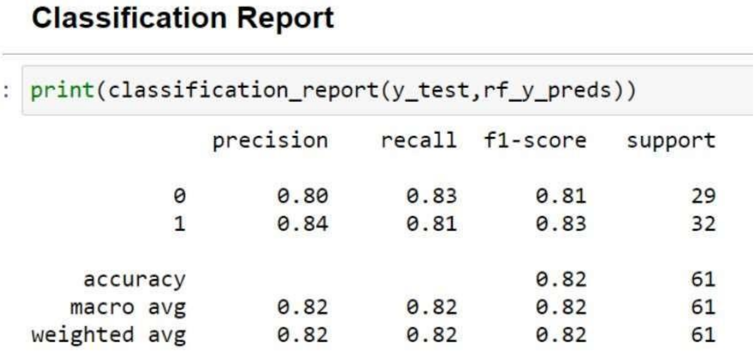
```
#Based on accuracy
model_compare=pd.DataFrame(model_scores,index=['accuracy'])
model_compare
```



	Logistic Regression	KNN	Random Forest
accuracy	0.885246	0.688525	0.852459

3. Random Forest Classifier (Accuracy of 85.2%)





Classification Report

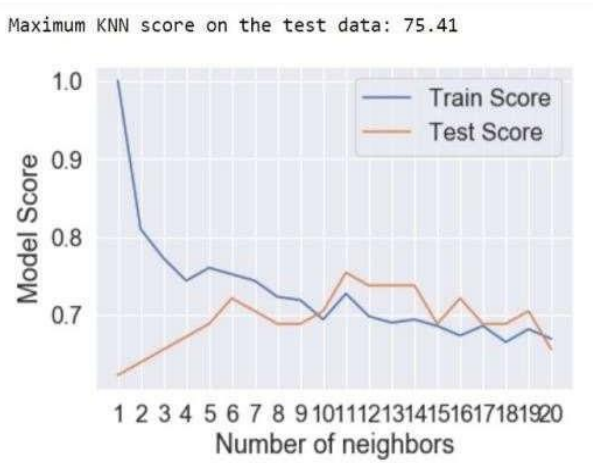
```
: print(classification_report(y_test,rf_y_preds))

              precision    recall  f1-score   support

           0       0.80      0.83      0.81        29
           1       0.84      0.81      0.83        32

    accuracy                           0.82        61
   macro avg       0.82      0.82      0.82        61
weighted avg       0.82      0.82      0.82        61
```

Now we have got baseline model...and we know a model's first prediction aren't always based our next steps off. What should we do?

Let's look at the following:

- HyperParameter tuning
- Feature Importance
- Confusion Matrix
- Cross-Validation
- Precision
- Recall
- F1-Score
- Classification Report
- ROC Curve
- Area under the curve(AUC)

HyperParameter Tuning (Manually) - K-Nearest Neighbour



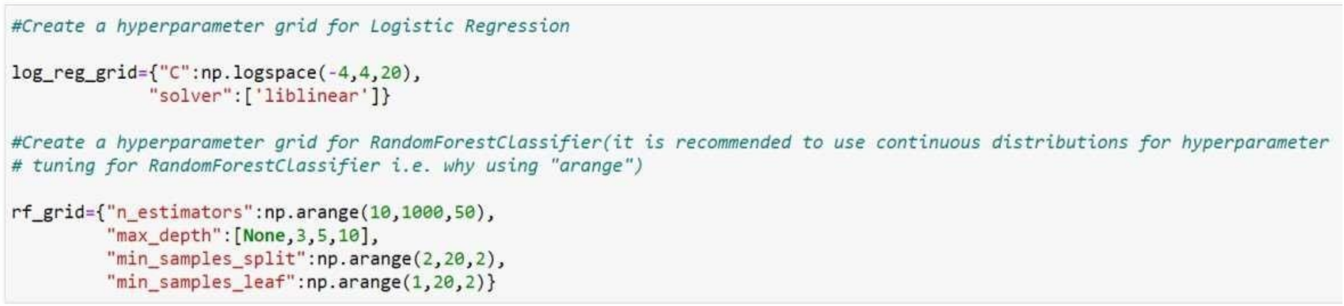
```
Maximum KNN score on the test data: 75.41
```

After KNN tuning also, KNN model got improved but still it is not predicting better than Random Forest and Logistic Regression. So we will discard it

HyperParameter tuning with Randomized Search CV

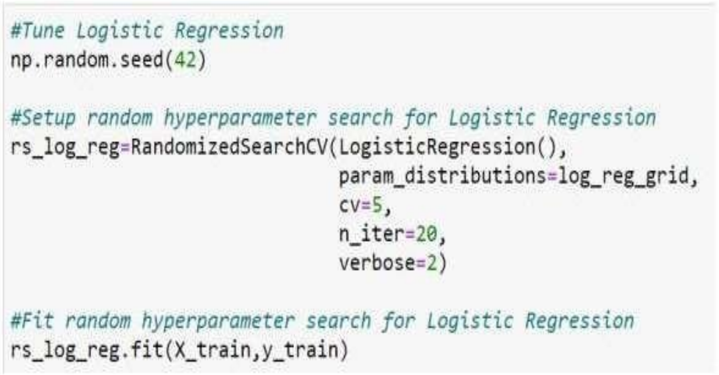
We are going to tune:

- Logistic Regression
- Random Forest Classifier

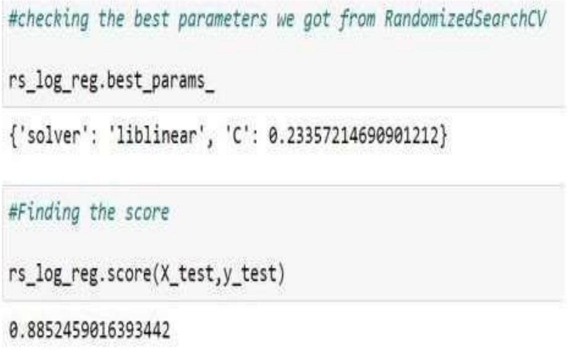


```
#Create a hyperparameter grid for Logistic Regression

log_reg_grid={"C":np.logspace(-4,4,20),
             "solver":['liblinear']}

#Create a hyperparameter grid for RandomForestClassifier(it is recommended to use continuous distributions for hyperparameter
# tuning for RandomForestClassifier i.e. why using "arange")

rf_grid={"n_estimators":np.arange(10,1000,50),
         "max_depth":[None,3,5,10],
         "min_samples_split":np.arange(2,20,2),
         "min_samples_leaf":np.arange(1,20,2)}
```

Logistic Regression



```
#Tune Logistic Regression
np.random.seed(42)

#Setup random hyperparameter search for Logistic Regression
rs_log_reg=RandomizedSearchCV(LogisticRegression(),
                              param_distributions=log_reg_grid,
                              cv=5,
                              n_iter=20,
                              verbose=2)

#Fit random hyperparameter search for Logistic Regression
rs_log_reg.fit(X_train,y_train)
```



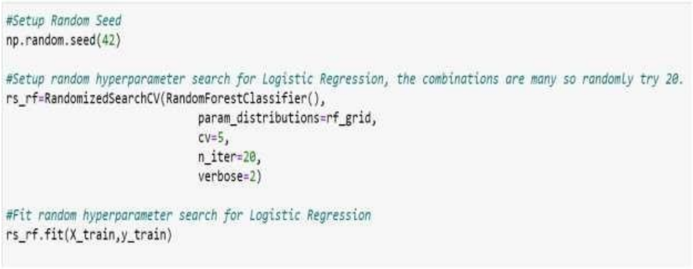
```
#checking the best parameters we got from RandomizedSearchCV

rs_log_reg.best_params_
```

```
{'solver': 'liblinear', 'C': 0.23357214690901212}
```

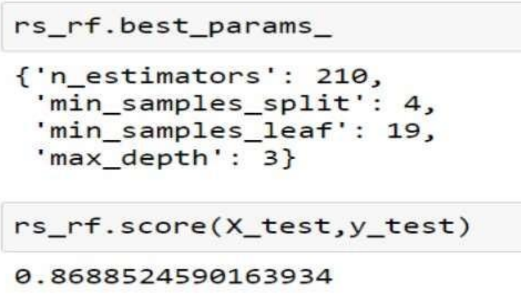
```
#Finding the score

rs_log_reg.score(X_test,y_test)
```

```
0.8852459016393442
```

Random Forest



```
#Setup Random Seed
np.random.seed(42)

#Setup random hyperparameter search for Logistic Regression, the combinations are many so randomly try 20.
rs_rf=RandomizedSearchCV(RandomForestClassifier(),
                         param_distributions=rf_grid,
                         cv=5,
                         n_iter=20,
                         verbose=2)

#Fit random hyperparameter search for Logistic Regression
rs_rf.fit(X_train,y_train)
```

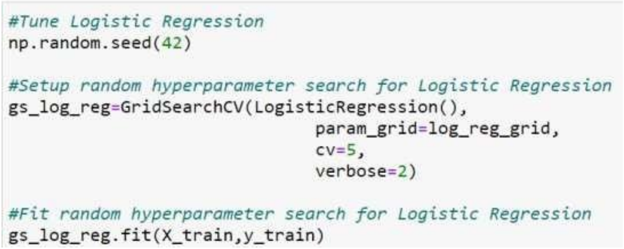


```
rs_rf.best_params_

{'n_estimators': 210,
 'min_samples_split': 4,
 'min_samples_leaf': 19,
 'max_depth': 3}

rs_rf.score(X_test,y_test)

0.8688524590163934
```

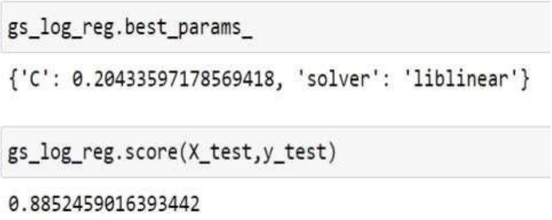
So now we have done RandomizedSearchCV, we will **eliminate RandomForest** as it's score is not much as compared to logistic Regression

HyperParameter tuning with GridSearchCV

Since our Logistic Regression model provides the best scores so far, we will try and improve it again using GridSearchCV.



```
#Tune Logistic Regression
np.random.seed(42)

#Setup random hyperparameter search for Logistic Regression
gs_log_reg=GridSearchCV(LogisticRegression(),
                        param_grid=log_reg_grid,
                        cv=5,
                        verbose=2)

#Fit random hyperparameter search for Logistic Regression
gs_log_reg.fit(X_train,y_train)
```

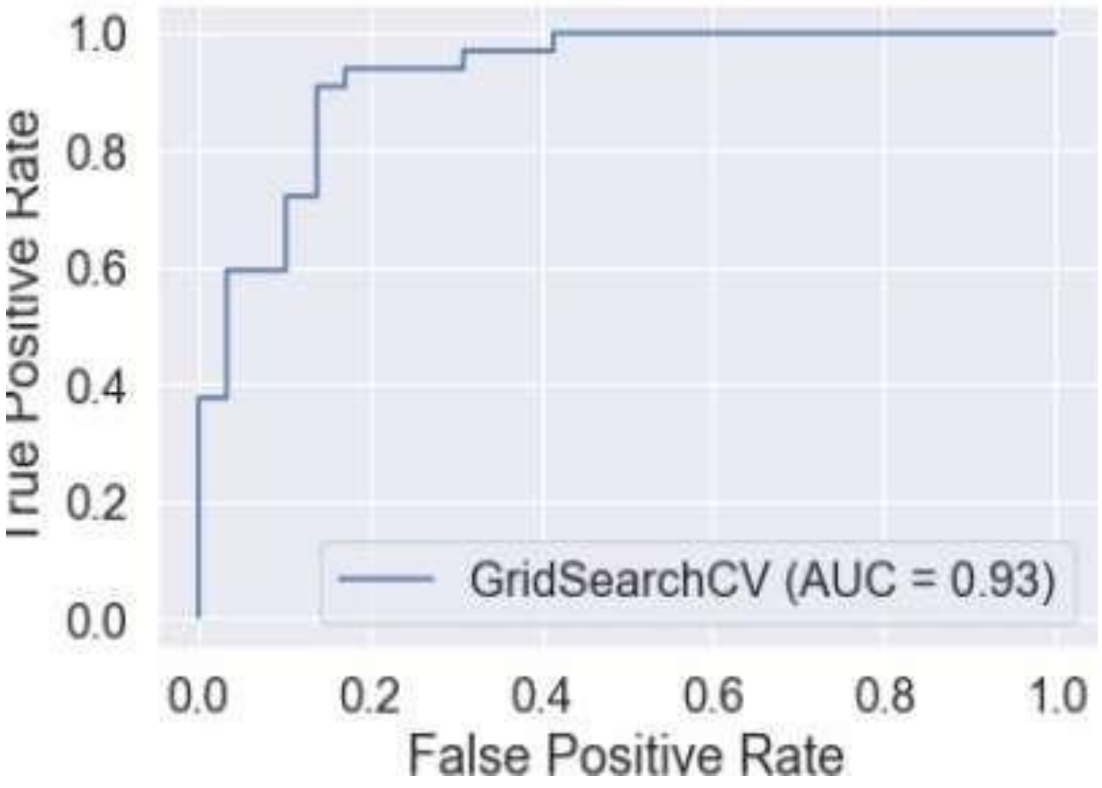


```
gs_log_reg.best_params_

{'C': 0.20433597178569418, 'solver': 'liblinear'}

gs_log_reg.score(X_test,y_test)

0.8852459016393442
```

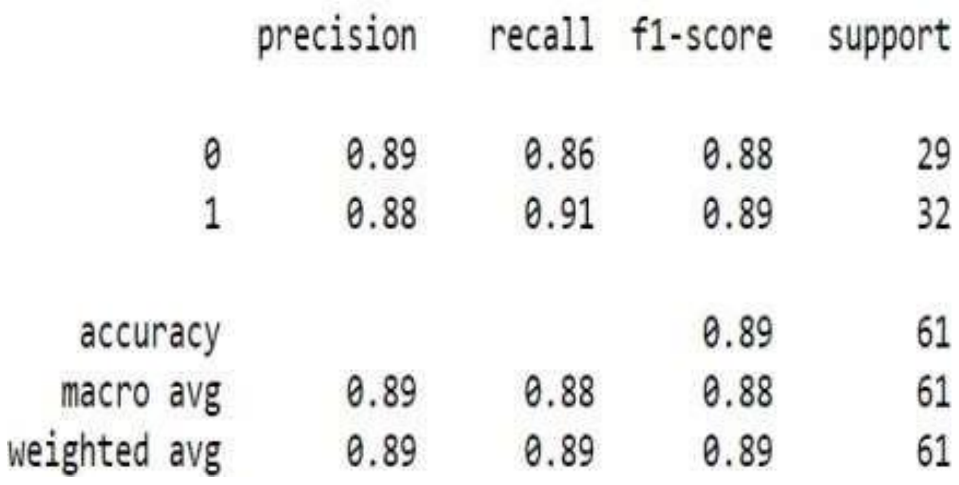
Step 4: Evaluating our tuned Logistic Regression model, beyond accuracy

- ROC curve and AUC score
- Confusion matrix
- Classification report
- precision
- recall
- f1-score
- Cross Validation

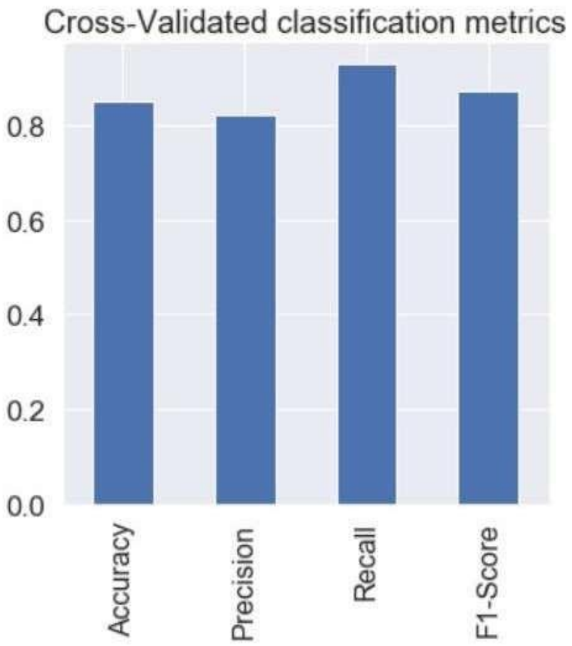
ROC and AUC

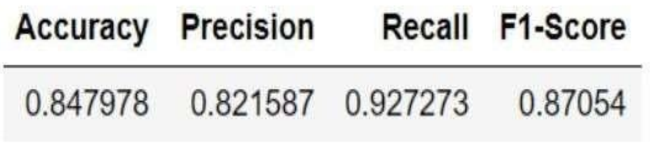


Classification Report



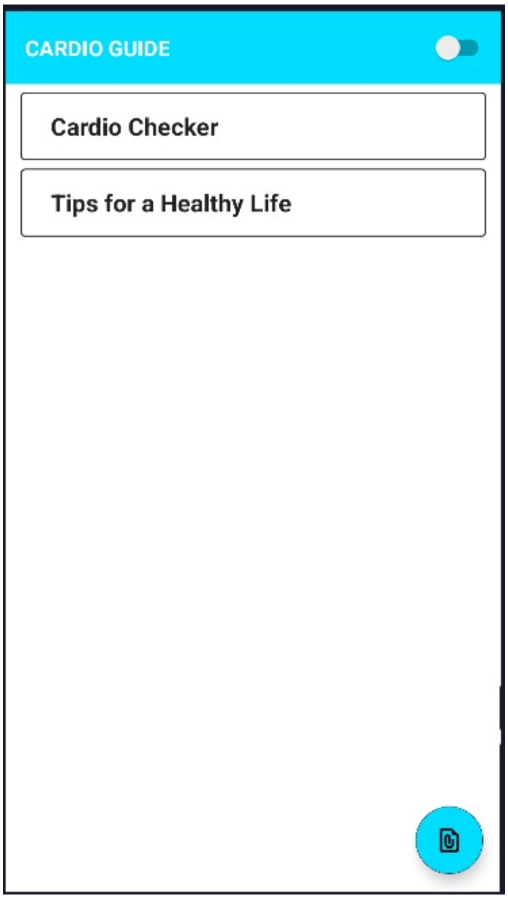
```
              precision    recall  f1-score   support

           0       0.89      0.86      0.88        29
           1       0.88      0.91      0.89        32

    accuracy                           0.89        61
   macro avg       0.89      0.88      0.88        61
weighted avg       0.89      0.89      0.89        61
```

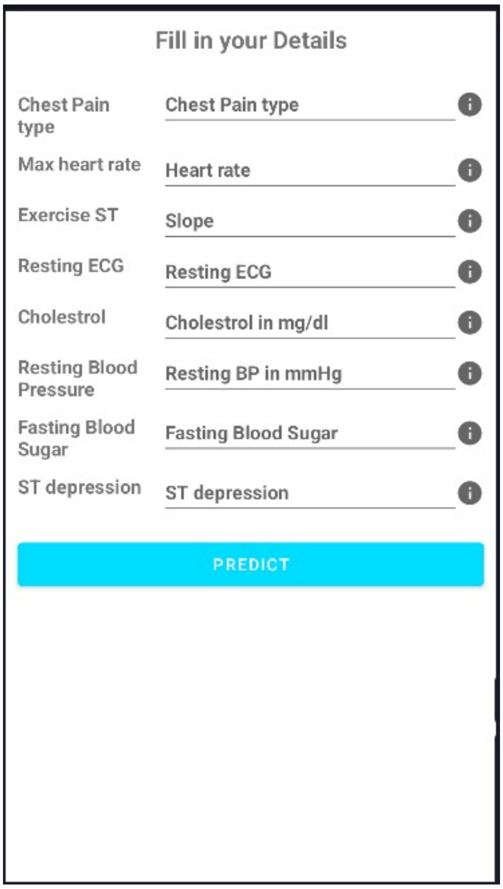




Accuracy	Precision	Recall	F1-Score
0.847978	0.821587	0.927273	0.87054

Step 5: Developing application front-end and back-end for users.





Fill in your Details

Chest Pain type	Chest Pain type
Max heart rate	Heart rate
Exercise ST	Slope
Resting ECG	Resting ECG
Cholestrol	Cholestrol in mg/dl
Resting Blood Pressure	Resting BP in mmHg
Fasting Blood Sugar	Fasting Blood Sugar
ST depression	ST depression

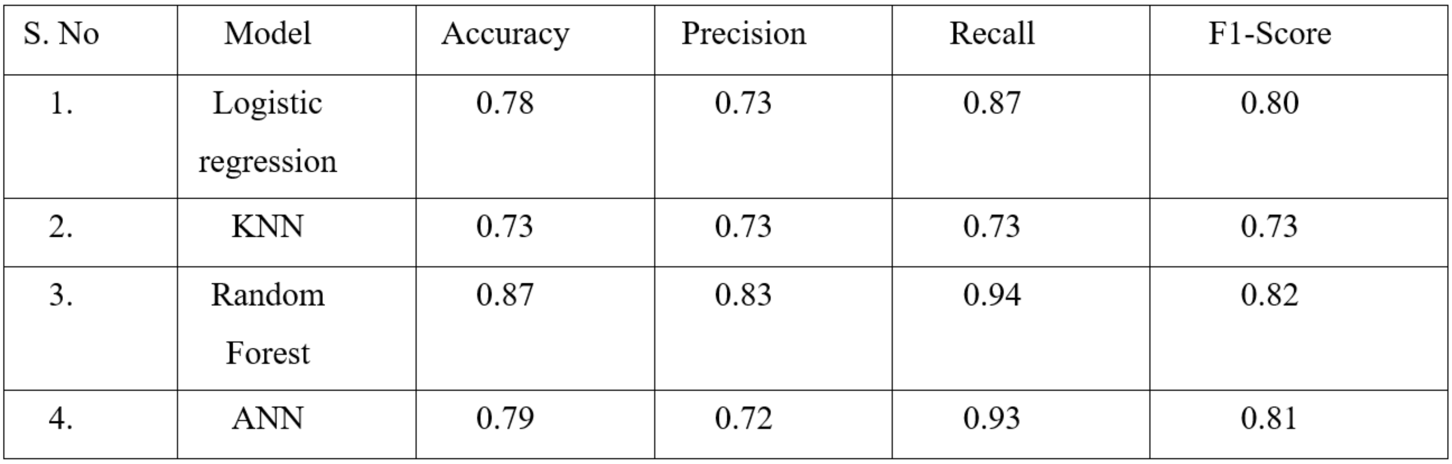
Chapter 05: Results

5.1 Discussion on the Results Achieved

For our "Heart Disease Prediction" project, we trained our model using four distinct models: random forest classifier, KNN classifier, ANN and logistic regression.

When it came to predicting heart disease, the Logistic Regression model that was trained on the dataset performed admirably. With an overall accuracy of 78%, the model demonstrated its capacity to accurately categorise people as having heart disease or not based on a variety of factors, including age, gender, type of chest discomfort, and other health markers. The model's precision, recall, and F1-score are also disclosed in the classification report, offering extra information about the predictive power of the model for each class. The model exhibits balanced performance across classes, with precision denoting the fraction of genuine positive predictions among all positive predictions, recall representing the fraction of true positives accurately detected, and F1-score representing the harmonic mean of precision and recall. Additionally, the confusion matrix shows how well the model classifies cases and offers a thorough analysis of its predicted strengths and shortcomings.

Table 2: Performance Report



S. No	Model	Accuracy	Precision	Recall	F1-Score
1.	Logistic regression	0.78	0.73	0.87	0.80
2.	KNN	0.73	0.73	0.73	0.73
3.	Random Forest	0.87	0.83	0.94	0.82
4.	ANN	0.79	0.72	0.93	0.81

General Observations:

1. Both the Random Forest and ANN models exhibit higher training accuracy but a significant drop in validation accuracy, suggesting overfitting.
2. The Random Forest model outperforms the other two models in terms of the best accuracy achieved.

5.2 Application of the Major Project

Numerous uses of the machine learning-based heart disease prediction study might enhance personal health and wellness. Here are some important uses:Early Isolation of Diseases: Determine who is at risk for heart disease before symptoms appear so that preventive and early intervention can be implemented.

1. Personalized Healthcare: Customise medical interventions according to each patient's risk profile to enable tailored treatment regimens and lifestyle advice.
2. Resource Optimization: Reduce the number of pointless tests performed on low- risk patients and give priority to those who pose a greater risk to help healthcare practitioners manage resources more effectively.
3. Telemedicine and Remote Monitoring: Enable remote cardiac health monitoring for patients to support telemedicine programmes by enabling medical experts to intervene as needed.
4. Health Insurance Risk Assessment: Help insurance firms determine how risky clients are for heart-related problems so they may make recommendations for policies and determine premiums.
5. Population Health Management: Assist in managing health at the population level by recognising patterns and trends in the risk of heart disease in various demographic groups and geographical areas.

6. Patient Education and Awareness: Help patients learn by giving them information about their risk factors and motivating them to make proactive lifestyle and health management decisions.
7. Clinical Decision Support: offer further information about a patient's risk of developing heart disease, assisting medical practitioners in making well-informed decisions.
8. Public Health Research: By offering a useful dataset to researchers so they can examine and comprehend the prevalence and risk factors of heart disease, you may support epidemiological studies.
9. Continuous Monitoring for Chronic Patients: Give people with chronic heart problems the option of ongoing monitoring so that treatment plans can be changed in real time and conditions can be identified early and before they deteriorate.
10. Wellness Programs: Encourage preventive steps and give staff personalised information into their heart health to support corporate wellness programmes.
11. Health Dashboards and Wearable Integration: Integrate with wearable technology and health dashboards to give people immediate feedback on their heart health and motivate them to lead healthier lives.
12. Integration with Electronic Health Records (EHR): Predictions and risk assessments can be seamlessly integrated into electronic health records to give medical professionals a complete picture of a patient's medical history.
13. Clinical Trials and Research Studies: Provide prediction models and risk assessments to clinical trials and research investigations aimed at treating and preventing heart disease.
14. Education and Training: act as an important teaching resource for healthcare practitioners, enabling them to comprehend and apply machine learning models to the prediction of cardiac disease.

5.3 Limitation of the Project

Although the machine learning-based cardiac disease prediction project has many advantages, there are some drawbacks that should be taken into account:

1. Data Quality and Availability: Biassed or small datasets may affect the model's generalizability and accuracy. Predictions that are biassed may result from missing or erroneous data.
2. Data Privacy Concerns: Adhering strictly to privacy standards is necessary while handling sensitive health data. Getting the right consent and protecting patient privacy can be difficult tasks.
3. Ethical Challenges: We must pay close attention to ethical issues, such as possible discrimination based on prediction models. Maintaining equity while averting unforeseen outcomes is a continuous endeavour.
4. Interpretability: The interpretability of highly complicated machine learning models may be compromised, making it difficult for medical practitioners to comprehend the logic underlying the predictions.
5. Dynamic Nature of Health Data: Risk factors and health conditions are subject to change. A model developed using past data might not be able to sufficiently adjust to changing health-related trends and behaviours.
6. Limited Causation Inference: One does not infer causation from correlation. The model can show correlations between specific characteristics and heart disease, but it is not always able to prove causation.
7. Overfitting and Generalization: The model's performance in real-world situations may be impacted by overfitting to the training set and difficulties generalising predictions to fresh, unforeseen data.
8. False Positives and Negatives: It is important to balance false positives and false negatives. While a high false negative rate could result in opportunities for preventative measures being overlooked, a high false positive rate could lead to needless treatments.
9. Dependency on Input Features: The availability and precision of input features have a major impact on prediction accuracy. Inaccurate or missing data in important features can affect how well a model performs.
10. Inherent Bias in Data: Predictions made by the model may be biassed due to bias in historical data, which could result in differences in risk ratings between various demographic groups.

11. Limited Adoption by Healthcare Professionals: Potential barriers to the implementation of AI-based technologies in clinical decision-making among healthcare professionals could restrict the model's applicability in real-world healthcare environments.
12. Algorithmic Complexity: Scalability may be limited by complex machine learning models due to their high computational cost and difficulty to implement in contexts with limited resources.
13. Lack of Patient Engagement: Patients may be reluctant to use prediction models because they have doubts about the security of their data, their confidence in the technology, and the possible psychological effects of using predictive health information.
14. Regulatory Compliance: Ensuring adherence to healthcare norms and laws, especially with relation to AI's application in healthcare, necessitates constant legal framework navigation.
15. Unforeseen External Factors: Over time, the model's relevance and applicability may be affected by external factors like societal changes, technological improvements, or changes in healthcare policies.

It is essential to comprehend these constraints in order to ethically create, implement, and use machine learning models for the prediction of cardiac disease. To overcome these obstacles and increase the efficacy of these predictive systems, ongoing research, openness, and cooperation with medical experts are crucial.

5.4 Future Work

Future work includes:

- Training more models and choose the best performing.
- Adding more features to the android app.
- Add more valid data resulting in higher prediction power.

For the heart disease prediction project, we intend to add features to Android app, there are many opportunities for further development and improvement. These are some ideas:

1. Real-Time Monitoring: Incorporate real-time monitoring features into the app so that it may be used to continuously evaluate and update a user's risk of heart disease based on changes in their health.
2. Integration with Wearable Devices: Connect the app to wearables (fitness trackers, smartwatches, etc.) to collect physiological data in real-time and provide the prediction model access to a wider range of features.
3. User-Friendly Interface: Concentrate on improving the user interface to make it more user-friendly and intuitive so that people can input and comprehend their health information with ease.
4. Personalized Health Recommendations: Expand the functionality of the app to offer individualised health advice based on the estimated risk, encouraging dietary adjustments and proposing precautionary steps.
5. Health Data Logging: Permit users to record more health-related information over time so that the app can adjust and enhance predictions based on data collected over time.
6. Secure User Authentication: To safeguard private health information, implement secure user authentication procedures to make sure that only people with permission can access and utilise the app.
7. Feedback Mechanism: Include a feedback system to gather user opinions and recommendations, enabling the app to be improved and improved over time.
8. Gamification Elements: To encourage users to use the app frequently and make the process of tracking and improving heart health more pleasurable, incorporate gamification aspects.
9. Multilingual Support: To serve a varied user base and make the software useable and accessible to people with varying linguistic backgrounds, offer multilingual assistance.

References

[1] Ivan Bratchenko, Lyudmela Bratchenko, Yulia Khristoforova. Classification of skin cancer using CNN analysis of Raman Spectra. ScienceDirect, November 2021.

[2] Tschandl, P., Rosendahl, C. & Kittler, H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Sci Data* 5, 180161 (2018). https://doi.org/10.1038/sdata.2018.161

[3] A Classification Method of Heart Disease Based on Heart Sound Signal. To cite this article: Lizhiyaun and Liuhaikuan 2019 J. Phys.: Conf. Ser. 1314 012028

[4] Prediction of Heart Diseases using Random Forest To cite this article: Madhumita Pal and Smita Parija 2021 J. Phys.: Conf. Ser. 1817 012009

[5] Joshua John, Mallia Galatti and Gillian Lee. Skin cancer detection using Convolutional and Artificial Neural Network. Journal of Computing Sciences. January 2020.

[6] Bagging Technique to Reduce Misclassification in Coronary Heart Disease Prediction Based on Random Forest To cite this article: A Saifudin et al 2020 J. Phys.: Conf. Ser. 1477 032009

[7] Ling Fang Lee, Xu Wang, Neal N. Xiaong and others. Deep Learning in Skin Disease Image Recognition. IEEE, November 2020.

[8] Vipul Dhabi, Vipul Goswami, Harshad Kumar. Skin Disease Classification from Image. IEEE, March 2020.

[9] Md Al Mamun, Mohammed Sharif. A Comparative Study Among Segmentation Techniques for Skin Disease Detection Systems. ResearchGate, January 2021.

[10] Akhtar Jamil, Merve Gun, Alaa Ali Hamid. Skin Lesions Segmentation and Classification for Medical Diagnosis. Researchgate April 2021.

[11] Yunendah Nur Fu'adah1, NK Caecar Pratiwi1, Muhammad Adnan Pramudito1 and Nur Ibrahim. Automatic Skin Cancer Classification System. IOP Science.

[12] Kamil Dililler, Boran Sekeroglu. Skin Lesion Classification Using CNN-based Transfer Learning Model. Journal of Science, January 2022.

[13] Akash kumar, Jalluri Rama, James Jing Khan. Classification of Skin Disease Using Deep Learning Neural Networks with MobileNet V2 and LSTM. mdpi, September 2021.

[14] Jessica Valesco, Cherry Pascheon, Jonathan Apuang. A Smartphone-Based Skin Disease Classification Using MobileNet CNN. ResearchGate, October 2019.

[15] C Pabitha, B Vinitha. Deep learning based severity grading for skin related issues, AIP Conference Proceedings, May 2022.

[16] Karthik R, Tejas Vaichole and Sanika Kulkarni. Channel Attention based Convolutional Network for skin disease classification. ScienceDirect, August 2021.

[17] Ridhi Arora , Balasubramanian Raman and Ruchi Awasthi. The Automated skin lesion segmentation using attention based deep Convolutional Neural Network. May 2020.

[18] Pawel Budura, Anna Platkowska and Joanna Czajowska. Deep learning approach to skin layer segmentation in inflammatory dermatoses. IEEE. July 2020.

[19] Raihan M, Mondal S, More A, Sagor M O F, Sikder G, Majumder M A & Ghosh K (2016, December). Smartphone based ischemic heart disease (heart attack) risk prediction using clinical data and data mining approaches, a prototype design. In 2016 19th International Conference on Computer and Information Technology (ICCIT) (pp. 299-303). IEEE.

[20] Buechler K F & McPherson P H (1999). U.S. Patent No. 5,947,124. Washington, DC: U.S. Patent and Trademark Office.

[21] Takci H (2018). Improvement of heart attack prediction by the feature selection methods. Turkish Journal of Electrical Engineering & Computer Sciences, 26(1), 1-10.

[22] Worthen W J, Evans S M, Winter S C & Balding D (2002). U.S. Patent No. 6,432, 124. Washington, DC: U.S. Patent and Trademark Office.

[23] Piller L B, Davis B R, Cutler J A, Cushman W C, Wright J T, Williamson J D & Haywood L J (2002). Validation of heart failure events in the Antihypertensive and Lipid Lowering Treatment to Prevent Heart Attack Trial (ALLHAT) participants assigned to doxazosin and chlorthalidone. Current controlled trials in cardiovascular medicine, 3(1), 10.

[24] Folsom A R, Prineas R J, Kaye S A & Soler J T (1989). Body fat distribution and self-reported prevalence of hypertension, heart attack, and other heart disease in older women. International journal of epidemiology, 18(2), 361-7.

[25] Kiyasu J Y (1982). U.S. Patent No. 4,338,396. Washington, DC: U.S. Patent and Trademark Office