LONDON METROPOLITAN UNIVERSITY

informatics college • pokhara

Module Code & Module Title

CU6051NP Artificial Intelligence

Assessment Weightage & Type

Coursework

Semester

2024 Spring

Student Name: Sisir Paudel

London Met ID: 23057059

College ID: NP04CP4S240056

Assignment Due Date: 21st Jan 2026

Assignment Submission Date: 21st Jan 2026

Supervisor Name: Jeevan Prakash Pant

I understand that my work needs to be submitted online via My Second Teacher under the relevant module page before the deadline for my assignment to be accepted and marked. I am aware that late submissions will be treated as non-submission and a zero mark will be awarded.

Table of Contents

Table of Figures

Table of Tables

1. Introduction

I have applied machine learning methods in power model development and comparison as my main task in this project. one of the main death causes in the world is cardiovascular diseases; therefore, the diagnosis that is trusted should be early and accurate. For this purpose, I have incorporated a data set with essential features as age, chest pain, cholesterol, blood pressure, plus lifestyle factors related to exercise. Via the trend as suggested by this data, I will provide a predictive model that will enable early detection and timely intervention in cardiovascular diseases.

My efforts in this project were mainly directed toward the exploration and evaluation of different prediction models in order to find the most appropriate one for the stated purpose. The three machine learning models I am using are: Logistic regression, decision tree and random forest classifier. The goal was to develop one model that could decide accurately, but also be trustworthy enough to be actually applied in the healthcare settings.

This whole project implied doing preparatory work cleaning the data, training the models properly, measuring their performance through evaluating, which were some of the challenges I managed to face. In addition, the dataset used in this project offered a large variety of useful features making it possible to reach high accuracy levels in the prediction process. The final outcome of this hard work was that I had been able to propose a predictive model that met the criteria of being both dependable and practicable for healthcare applications.

The project has been a hands-on experience for me about the great potential of the application of machine learning in the healthcare technology industry. The main contribution of this work to the development of the field might be in the form of a stepping stone towards the innovations in the early detection and treatment of cardiovascular diseases.

1.1. Topic and Concept Visualization

Cardiovascular diseases were, without a doubt, the number one killer globally in 2019 having caused 17.9 million deaths which represented 32% of the total deaths worldwide. Out of the total, 85% were attributed to heart attacks and strokes. Low- and middle-income nations were

responsible for over 75% of the affected population. The World Health Organization also reported that 38% of the total 17 million premature deaths - under 70 years - caused by non-communicable diseases, were due to heart diseases in 2019 (WHO, 2021).

People with heart-related issues has a good chance of recovering, if only they get rid of, or at least, cut down on, air pollution, have a balanced and nutritious diet, engage in physical activities, moderate their alcohol intake, and quit smoking. Moreover, early detection allows the management of cardiac disorders through counseling and medication.

Prediction of cardiovascular diseases is about knowing the connections among different health factors like age, gender, blood pressure, cholesterol, chest pain, maximum heart rate, and blood sugar on the basis of the likelihood of having heart problems. I am going to train many machine learning algorithms on a dataset that includes features related to health like these in this project.

Besides, the performance of the algorithms will be assessed in terms of their accuracy in predicting the occurrence of cardiovascular diseases.

.

1.1.1. AI association with the concept

Artificial Intelligence has been boosting the performance of various machine learning models, including logistic regression, decision trees, and random forests. These are some models that will be using AI to handle large datasets, find complicated patterns, and make proper predictions. AI may also be used in lots of automated tasks such as feature selection, hyperparameter tuning, and model performance evaluation in giving good performance without much interference made by humans.

The AI-based CVD prediction model will enable the derivation of knowledge from various health indicators, adapt changes in data distribution, and make accurate predictions to assist practitioners in the early-stage diagnosis and treatment planning process. This will help not just in the development of better healthcare outcomes but also contribute toward the advancement of medical technology.

1.2. Introduction to Problem Domain

Cardiovascular disease (CVD) is a global health crisis that affects society and the economy significantly. CVD is still the top cause of death in the USA and the same organization, the

World Health Organization, reported in 2021 that 17.7 million deaths are attributed to CVD all over the globe (WHO, 2021). Huge need for advanced CVD detection tools as diagnostic, risk factor reduction, and treatment options would be the area of impact.

CVD is mainly due to an interplay of risk factors. Early evaluating of lifetime risks can drastically minimize the number of deaths from CVD. Lifestyle changes are among the best methods to protect one's heart; quitting the smoking habit, controlling high blood pressure, and regular exercise are some examples (Jarett D. Berry, 2012).

The application of machine learning to predict CVD in health indicator analysis and to support health services in that area might be the solution. Health improvement through the use of inventive strategies should win the battle against diseases as a war.

The goal of this research project is to come up with a machine learning-based predictive model for CVD that will be usable in the healthcare sector. The early detection models will result in timely prevention and intervention, which in return will lead to better health outcomes and fewer CVD-related deaths, given that this is a growing global concern.

1.3. Project Motivation

My motivation for this project springs from deep interest in the combination of technology with health care in problem-solving. It is one of the most important critical health problems which affects millions of people worldwide, and I was looking forward to contributing towards its early-stage detection and prevention by using innovative techniques.

That is what impresses me the most with the use of machine learning in health data analysis; it aids in finding patterns that could be left undercover with traditional methods. This project presents an opportunity to apply my data analysis and machine learning skills in the making of something that will make a difference in people's lives.

Thirdly, the potential influence which this work will have motivates me: developing a predictive model of CVD places me in an exclusive position to provide a tool to health professionals to spot people who are in danger on time, hence opening a window to timely interference and saving lives

1.4. Aims & Objectives

1.4.1. Aims

In this specific research, the machine learning model's main goal is to predict a person's likelihood of having cardiovascular disease (CVD) based on the characteristics of a person's lifestyle and medical history. This, in turn, can be a great asset in the healthcare sector because it might enable the healthcare providers to spot the at-risk patients and then either lower the risk of CVD or mitigate the impact through timely intervention. The study will also stress the point of comparing different M.L. methods for predicting CVD risks along with their performance analysis based on accuracy, precision, and recall as evaluation metrics.

1.4.2. Objectives

The aim is to retrieve and delve into cardiovascular disease data.

- **Data Preprocessing:** Fill-in the missing data in the dataset, encode the categorical data, normalize the numeric data.
- **Selection of Features:** Selecting a subset of input variables CVD that are most informative.

  In order to utilize the machine learning algorithm.

  This is to compare the results in order to determine which model is best suited to predict the risk of CVD.

  To Single out the most functioning model to be put into practice in cardiovascular risk assessment.

1.5. Dataset

The prediction dataset includes patient health characteristics such as age, a significant risk factor for CVD, and the patient's sex, as the likelihood of developing CVD varies between men and women. Other features include chest discomfort or chest pain forms, serum cholesterol in mg/dl, and resting blood pressure in mm Hg as the necessary indications of heart health. Further, maximal heart rate achieved, resting electrocardiogram results, and fasting blood sugar level are other features indicating the health of the heart. Finally, it is the blood condition of thalassemia that influences cardiovascular health.

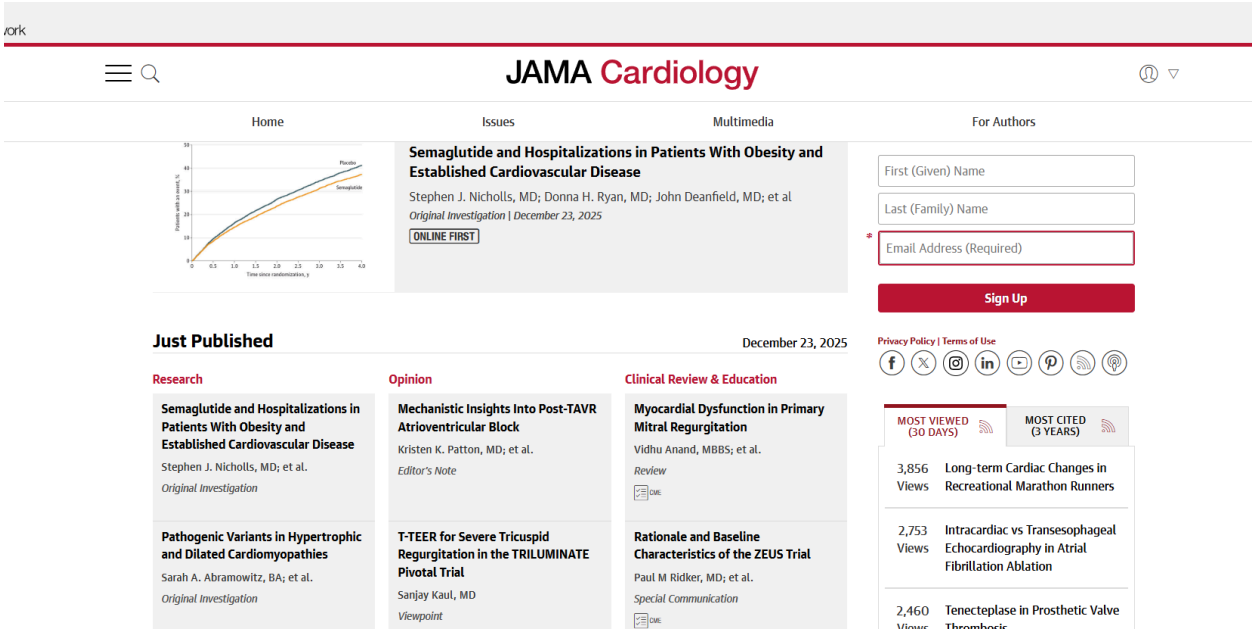
1.5.1. Data Features in the Dataset:

The following is a list of the features in the cardiovascular datasets.

- Age: The primary risk factor in heart diseases is age. Individuals aged above 60 years have an increased chance of developing a CVD.
- Sex: This factor shows the difference in the distribution of the risk of CVD between men and women.
- Type of Chest Pain: differentiates between a variety of chest pains caused because of a suspected heart disease. Resting Blood Pressure: It is a blood pressure of an individual who is not engaged in any activity. This is an extremely significant cardiovascular disease indicator.
- Serum Cholesterol: The likelihood of cardiovascular diseases is directly associated with the level of blood cholesterol of a patient. Fasting Blood Sugar: High levels of fasting blood sugar have already been identified to be a risk factor of heart disease.
- Electrocardiogram Results: The records the electrical activity of the heart and displays the possible abnormalities.
- Maximal Heart Rate Achieved: The work of the heart under stress and provides appropriate information on the fitness of the heart.
- Oldpeak: It is used to measure the performance of the heart in cases of stress by comparing exercise induced ST depression and rest.
- Slope of Peak Exercise ST Segment: It refers to the variations of the activity of the heart with exercise.
- Thalassemia: It is a blood condition that may impact the cardiovascular health.

2. Background and Literature Review

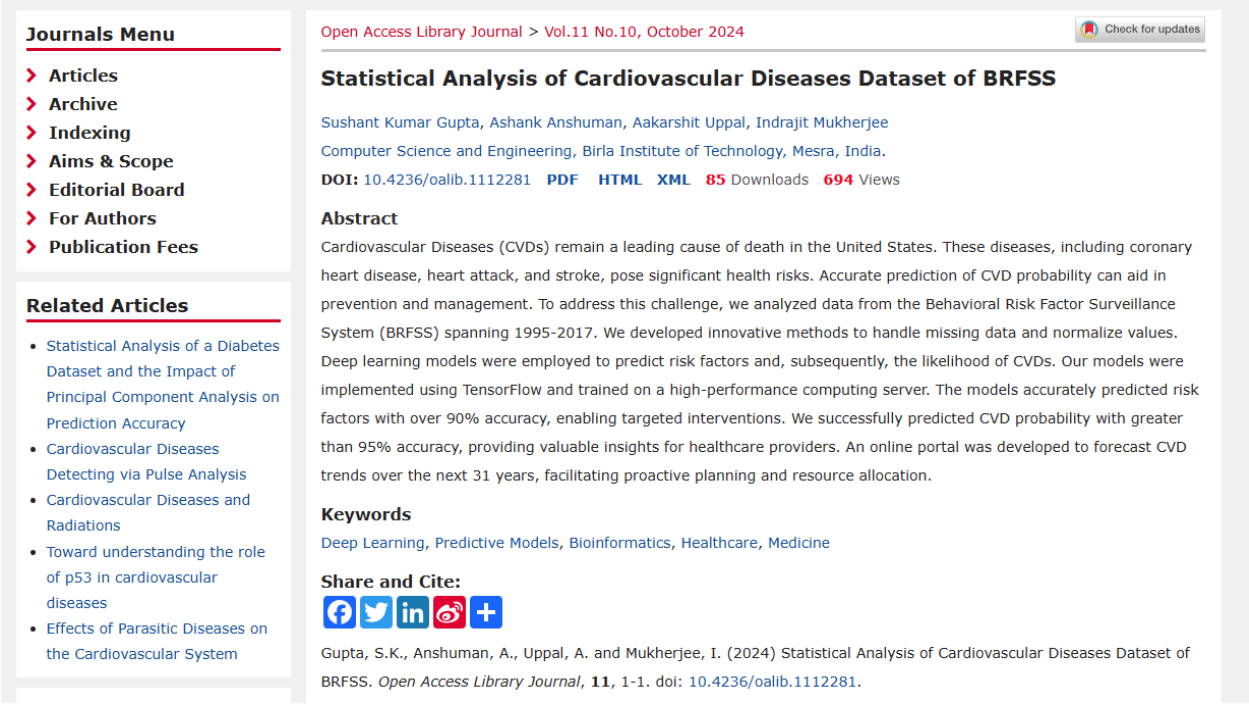
2.1. Journal 1



Link:https://jamanetwork.com/journals/jamacardiology

This review of the SELECT trial included a significant reduction in hospital admissions time in hospital with semaglutide once-weekly in cardiovascular disease and overweight or obese patients not diagnosed with diabetes. The gains were comparable to all the age, sex, and BMI groups, which proves that semaglutide can extend more benefits than cardiovascular risk reduction to reduce the overall hospital burden.

2.2. Journal 2



Journals Menu
- Articles
- Archive
- Indexing
- Aims & Scope
- Editorial Board
- For Authors
- Publication Fees

Related Articles
- Statistical Analysis of a Diabetes Dataset and the Impact of Principal Component Analysis on Prediction Accuracy
- Cardiovascular Diseases Detecting via Pulse Analysis
- Cardiovascular Diseases and Radiations
- Toward understanding the role of p53 in cardiovascular diseases
- Effects of Parasitic Diseases on the Cardiovascular System

Open Access Library Journal > Vol.11 No.10, October 2024

Check for updates

Statistical Analysis of Cardiovascular Diseases Dataset of BRFSS

Sushant Kumar Gupta, Ashank Anshuman, Aakarshit Uppal, Indrajit Mukherjee

Computer Science and Engineering, Birla Institute of Technology, Mesra, India.

DOI: 10.4236/oalib.1112281 **PDF** **HTML** **XML** **85** Downloads **694** Views

Abstract

Cardiovascular Diseases (CVDs) remain a leading cause of death in the United States. These diseases, including coronary heart disease, heart attack, and stroke, pose significant health risks. Accurate prediction of CVD probability can aid in prevention and management. To address this challenge, we analyzed data from the Behavioral Risk Factor Surveillance System (BRFSS) spanning 1995-2017. We developed innovative methods to handle missing data and normalize values. Deep learning models were employed to predict risk factors and, subsequently, the likelihood of CVDs. Our models were implemented using TensorFlow and trained on a high-performance computing server. The models accurately predicted risk factors with over 90% accuracy, enabling targeted interventions. We successfully predicted CVD probability with greater than 95% accuracy, providing valuable insights for healthcare providers. An online portal was developed to forecast CVD trends over the next 31 years, facilitating proactive planning and resource allocation.

Keywords

Deep Learning, Predictive Models, Bioinformatics, Healthcare, Medicine

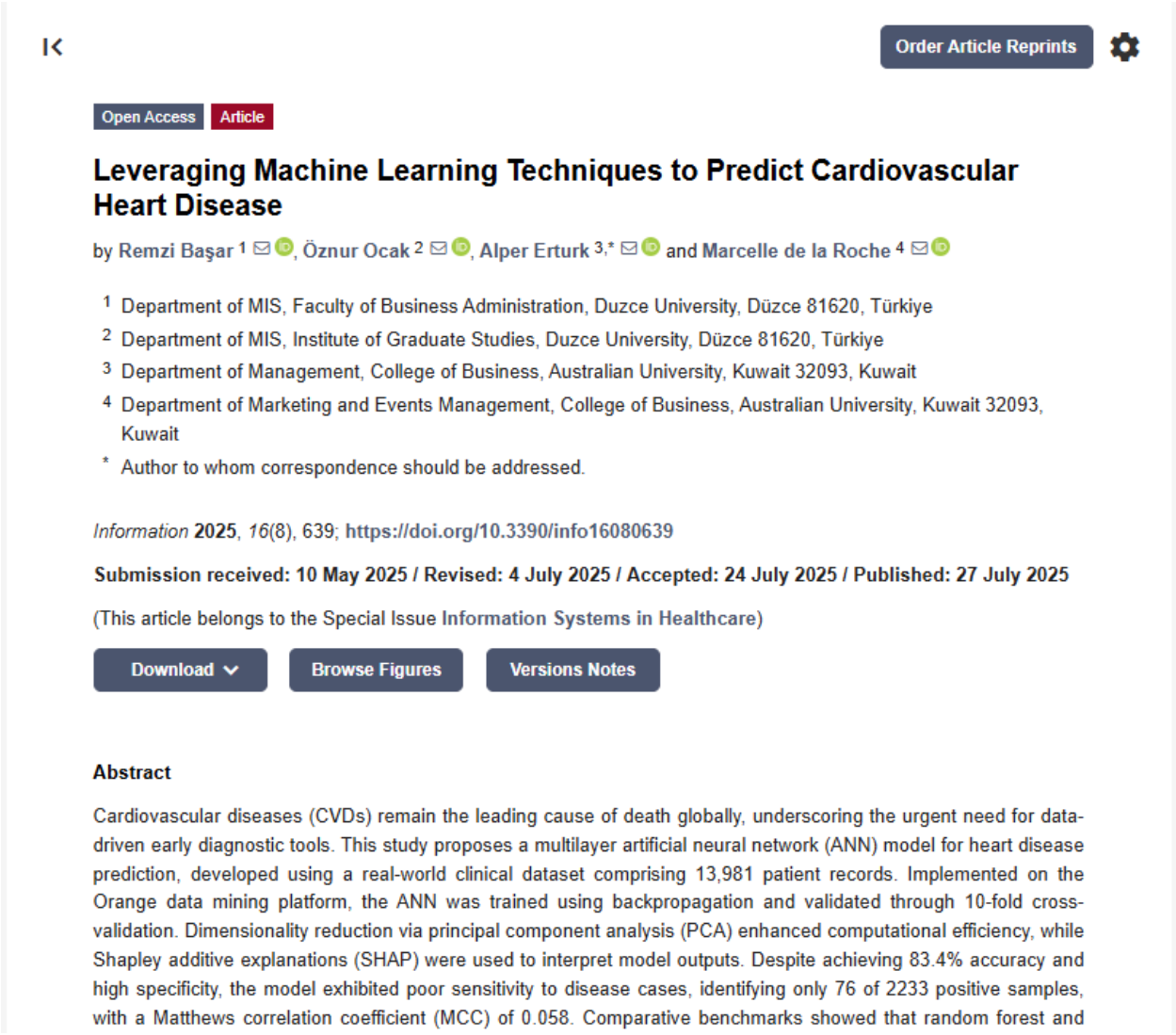
Share and Cite:

Gupta, S.K., Anshuman, A., Uppal, A. and Mukherjee, I. (2024) Statistical Analysis of Cardiovascular Diseases Dataset of BRFSS. *Open Access Library Journal*, **11**, 1-1. doi: 10.4236/oalib.1112281.

Link:https://www.scirp.org/journal/paperinformation?paperid=137075

One of the major causes of death in the United States is Cardiovascular Diseases (CVDs). The diseases such as coronary heart disease, heart attack as well as stroke are highly dangerous. CVD probability prediction can help prevent and manage the disease. To overcome this obstacle, we examined data provided by Behavioral Risk Factor Surveillance System (BRFSS), 1995-2017. We also devised new ways of dealing with missing data and normalization of values. The use of deep learning models was intended to forecast risk factors and, consequently, the probability of CVDs. We have used our models on a high-performance computing server through the implementation in TensorFlow.

2.3. Journal 3



Order Article Reprints

Open Access Article

Leveraging Machine Learning Techniques to Predict Cardiovascular Heart Disease

by Remzi Başar [1], Öznur Ocak [2], Alper Erturk [3,*] and Marcelle de la Roche [4]

[1] Department of MIS, Faculty of Business Administration, Duzce University, Düzce 81620, Türkiye
[2] Department of MIS, Institute of Graduate Studies, Duzce University, Düzce 81620, Türkiye
[3] Department of Management, College of Business, Australian University, Kuwait 32093, Kuwait
[4] Department of Marketing and Events Management, College of Business, Australian University, Kuwait 32093, Kuwait
* Author to whom correspondence should be addressed.

Information **2025**, *16*(8), 639; https://doi.org/10.3390/info16080639

Submission received: 10 May 2025 / Revised: 4 July 2025 / Accepted: 24 July 2025 / Published: 27 July 2025

(This article belongs to the Special Issue Information Systems in Healthcare)

Download Browse Figures Versions Notes

Abstract

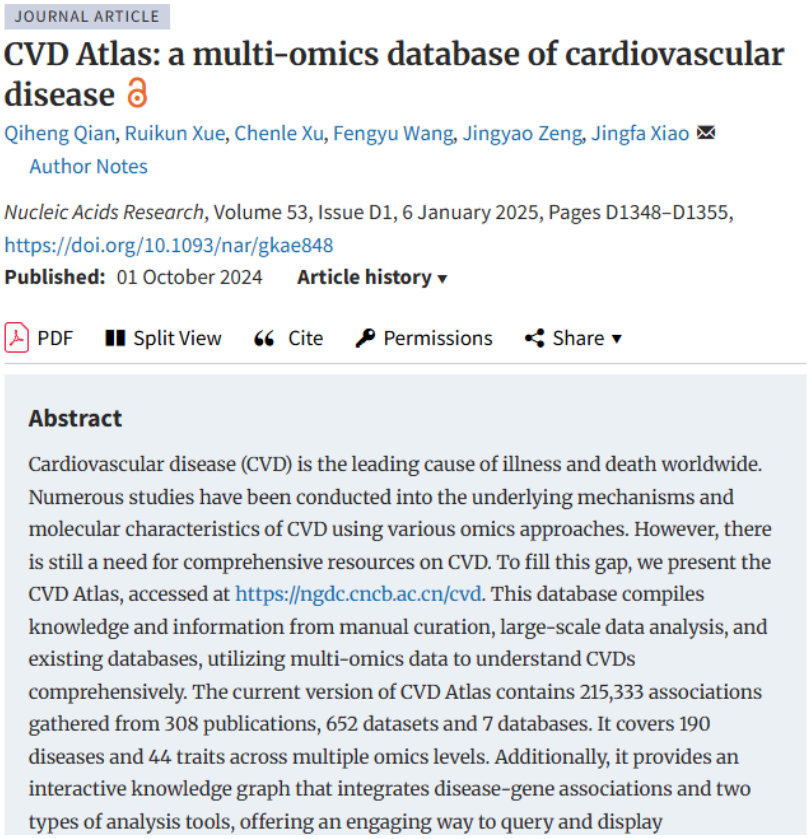
Cardiovascular diseases (CVDs) remain the leading cause of death globally, underscoring the urgent need for data-driven early diagnostic tools. This study proposes a multilayer artificial neural network (ANN) model for heart disease prediction, developed using a real-world clinical dataset comprising 13,981 patient records. Implemented on the Orange data mining platform, the ANN was trained using backpropagation and validated through 10-fold cross-validation. Dimensionality reduction via principal component analysis (PCA) enhanced computational efficiency, while Shapley additive explanations (SHAP) were used to interpret model outputs. Despite achieving 83.4% accuracy and high specificity, the model exhibited poor sensitivity to disease cases, identifying only 76 of 2233 positive samples, with a Matthews correlation coefficient (MCC) of 0.058. Comparative benchmarks showed that random forest and

Link:https://www.mdpi.com/2078-2489/16/8/639

The cardiovascular diseases (CVDs) have been the primary cause of mortality worldwide, which is why there is a pressing need to develop data-driven early diagnostic instruments. This paper presents a multilayer neural network (ANN) approach to heart disease prediction, which will be developed on a real-life clinical sample consisting of 13,981 patient cases. The ANN, which is implemented on the Orange data mining platform, is trained with the use of the backpropagation and evaluated by means of the 10-fold cross-validation. The usage of dimensionality reduction with the help of the principal component analysis (PCA) improved computational efficiency, and Shapley additive explanations (SHAP) were employed to elucidate the model outputs. Although

the model had an accuracy of 83.4 and high specificity, it had low sensitivity to disease patients and it could only identify 76 of 2233 positive samples with a Matthews correlation coefficient (MCC) of 0.058. Comparison with reference standards demonstrated that random forest and support vector machines were much more successful than ANN when it comes to discrimination (AUC up to 91.6%).

2.4. Journal 4



JOURNAL ARTICLE

CVD Atlas: a multi-omics database of cardiovascular disease

Qiheng Qian, Ruikun Xue, Chenle Xu, Fengyu Wang, Jingyao Zeng, Jingfa Xiao

Author Notes

Nucleic Acids Research, Volume 53, Issue D1, 6 January 2025, Pages D1348–D1355,
https://doi.org/10.1093/nar/gkae848

Published: 01 October 2024 **Article history**

PDF Split View Cite Permissions Share

Abstract

Cardiovascular disease (CVD) is the leading cause of illness and death worldwide. Numerous studies have been conducted into the underlying mechanisms and molecular characteristics of CVD using various omics approaches. However, there is still a need for comprehensive resources on CVD. To fill this gap, we present the CVD Atlas, accessed at https://ngdc.cncb.ac.cn/cvd. This database compiles knowledge and information from manual curation, large-scale data analysis, and existing databases, utilizing multi-omics data to understand CVDs comprehensively. The current version of CVD Atlas contains 215,333 associations gathered from 308 publications, 652 datasets and 7 databases. It covers 190 diseases and 44 traits across multiple omics levels. Additionally, it provides an interactive knowledge graph that integrates disease-gene associations and two types of analysis tools, offering an engaging way to query and display

Link:https://academic.oup.com/nar/article/53/D1/D1348/7798784

Cardiovascular disease (CVD) is the most common cause of morbidity and mortality all over the world. The mechanisms and molecular peculiarities underlying CVD have been studied numerous times and using various omics methods. However, there is still a need to have much CVD materials. This database consolidates the experience and the knowledge of manual curation, a large-scale data analysis, and existing databases to acquire knowledge about CVDs in a multi-omics way. The current version of CVD Atlas features 215, 333 associations that were

gathered with the help of 308 publications, 652 datasets and 7 databases. It consists of 190 diseases and 44 traits of different degrees of omics. It also provides an interactive knowledge graph, which is a combination of disease-gene associations and two types of analysis tools which is an appealing approach to querying and presenting relationships. The web interface can also be used to navigate easily in CVD Atlas, whereby any user is free to access, search and download all association data and research metadata and annotation information.

2.5. Journal 5

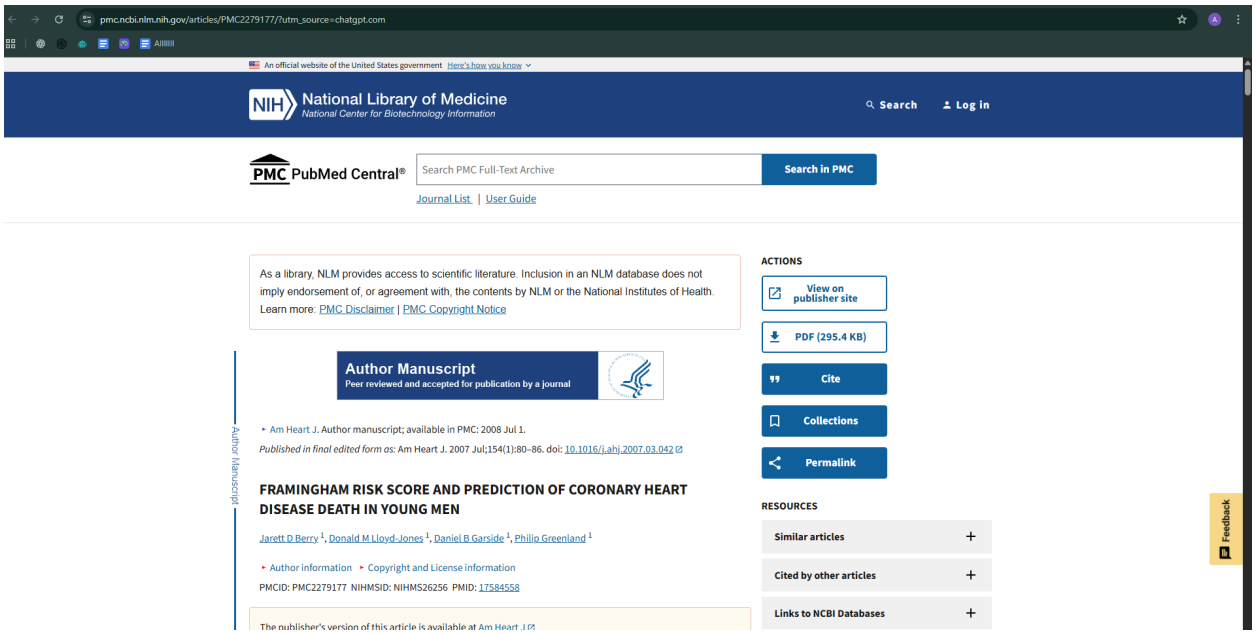
Link:https://link.springer.com/article/10.1007/s44174-025-00564-2

It has already been established that cardiovascular disease (CVD) is among the largest causes of mortality in the whole world and especially in the Indian subcontinent where the consumption of high cholesterol food is mostly blamed as the cause of the disease. A number of machine learning models have been used and fully cross-validated with stratified fivefold cross-validation after which the optimum parameters have been selected on the findings of the grid search cross-validation. The model performance was evaluated on the basis of accuracy and precision, recall, F1-score, AUC and ROC curve analysis. A SHapley Additive exPlanations (SHAP) analysis was used to enhance the insight of how the model makes decisions, and its emphasis

was placed on the feature significance on the whole. The best performance was observed in XGBoost with training and testing accuracy of 97.12 and 86.07 percent (AUC=0.91) respectively as seen in the performance it gave on the testing data. The second was the Random Forest, which has the best performance of highest AUC (0.92) and 83.08% testing accuracy. K-Nearest Neighbors and Decision Tree followed with the testing accuracy of 74.63 and 78 respectively. Logistic Regression and Support Vector machine were lower in overall accuracy (c. 66) but both with high recall (0.91 and 0.95) which expressed sensitivity to positive cases. Not only do these results indicate good XGBoost performance, but also they indicate the trade-offs of other models.

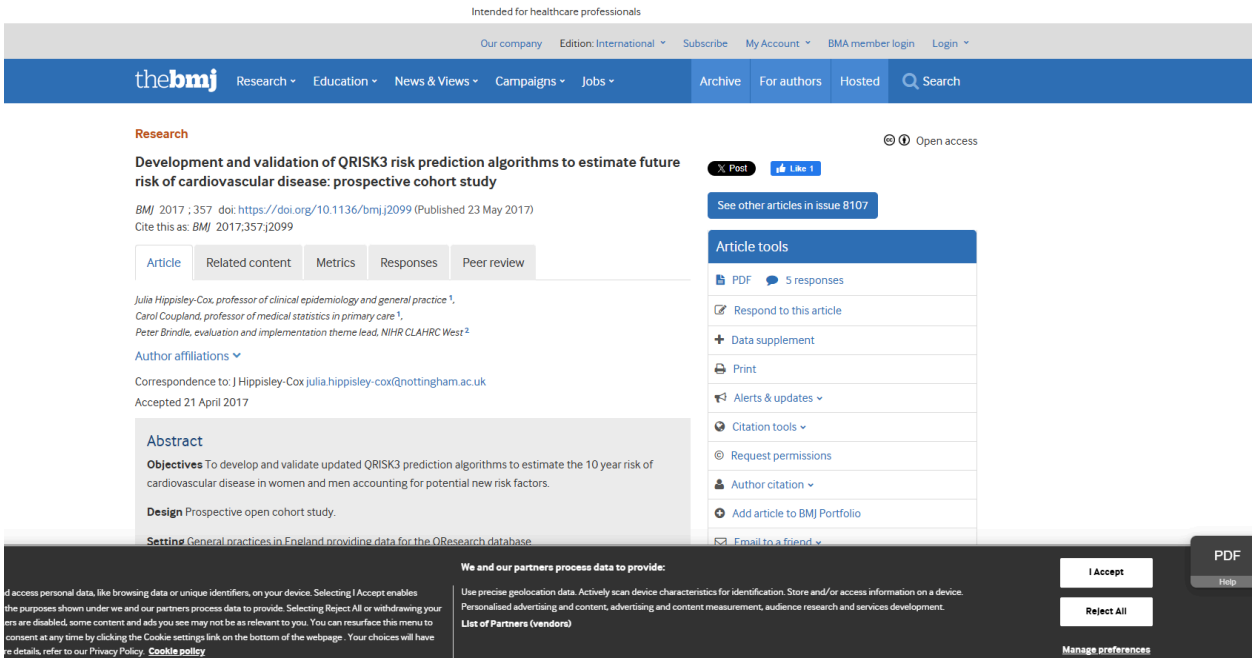
2.6. Research Gaps (Cardiovascular Disease) Existing Systems

Most nations already implement cardiovascular risk prediction tools to aid in the planning of early screening and prevention in clinics. An example of one is the ASCVD Risk Estimator Plus, developed by the American College of Cardiology and estimated a 10-year cardiovascular risk of a person, using the pooled cohort equation, but is primarily configured as a rule-based calculator rather than a learning system (ACC, n.d.; ClinCalc, 2025). Equally, QRISK3 is similarly common in the UK to project the risk of heart attack or stroke in 10 years, using large volume of primary care data and an encapsulated risk equation (Hippisley-Cox et al., 2017; QRISK, n.d.). Even older scoring methods like the Framingham risk functions are not forgotten to be used to estimate 10-year cardiovascular risk based on major risk factors like age, cholesterol, blood pressure and smoking status, but was constructed on selected cohorts and might not generalise as well with other populations and lifestyles nowadays (Berry et al., 2007; Framingham Heart Study, n.d.). In practice, they are useful in rapid decision support, though usually population-specific, based on the same kind of fixed statistical assumptions, and may be not very effective in reflecting complex non-linear interactions between risk factors (such as interactions between age, blood pressure, cholesterol patterns, and type of chest pain).



Link:https://pmc.ncbi.nlm.nih.gov/articles/PMC2279177/?utm

Recent studies indicate that supervised machine learning has been used more recently to predict cardiovascular disease as algorithms like Logistic Regression, Decision Trees, and Random Forests are able to model patterns in patient characteristics and sometimes even enhance performance by learning interactions within the data when they are not all linear (Hossain et al., 2024). Nonetheless, one weakness that is recurrent in the literature is that some of them only consider a single model, or they do not provide a fair, side-by-side comparison with the consideration of the same preprocessing procedures, the same train/test split strategy, and the same evaluation metrics-so it becomes difficult to claim what is, in actual fact, the best option to use in a particular dataset and use case (Qian, 2025). This leaves a research gap: there is a necessity to have one, systematic pipeline, which pre-processes the cardiovascular data in a consistent manner and subsequently compares a number of supervised algorithms collectively, applying similar metrics, to determine the most trustworthy model of predicting cardiovascular disease risk. Thus, the assigned task fills that gap by exploring a comparative supervised learning workflow (e.g. Logistic Regression vs Decision Tree vs Random Forest) to identify which model is the most helpful in cardiovascular prediction with the help of the chosen dataset.



Link:https://www.bmj.com/content/357/bmj.j2099?utm

3. Solution

3.1.. Proposed Solution Approach

The suggested method utilizes supervised machine learning with classification techniques to analyze a labeled medical dataset and predict the probability of the occurrence of the heart disease. The dataset with historical patient records includes labels that clearly define the outcome as either the presence or absence of cardiovascular disease, which is the reason why supervised learning is chosen as the appropriate method for this problem.

The first step is to get a dataset for cardiovascular diseases and next there will be a thorough preprocessing step where the irrelevant variables are eliminated and the missing values are treated in an appropriate manner. Selected for model training are both numerical and categorical features that are of clinical significance they include age, gender, blood pressure, cholesterol levels, body mass index, and lifestyle-related factors. These features are then used to train several classification models which are able to find out the hidden relationships between health indicators of the patients and the risk of heart diseases.

After the models have been trained, they can then predict if a person is likely to get cardiovascular disease. The predictions made by the models are very helpful in detecting patients with high risk early in the process thus promoting preventive healthcare practices and making clinical decisions based on the information.

3.2. Explanation of Algorithm Used

The prediction models selected for this project are:

3.2.1. (Model - Logistic Regression)

Logistic regression is among the most popular techniques applied in the case of binary classification tasks and is considered one of the mighty tools in solving predictive issues. It happens through the logistic function, which is an S-shaped curve mapping the weighted linear relationship of input features onto a value within the range of 0 and 1 (Saidi, 2021). This hence admits logistic regression to give the predicted probabilities without merely classified data points into categorical values.

Predicting the probability of an outcome involves giving more insights about the likelihood of a certain data point belonging to a particular class. The ability of this class, along with the interpretability of its output regarding the probability, makes logistic regression among the vital techniques in most of the predictive and classification tasks.

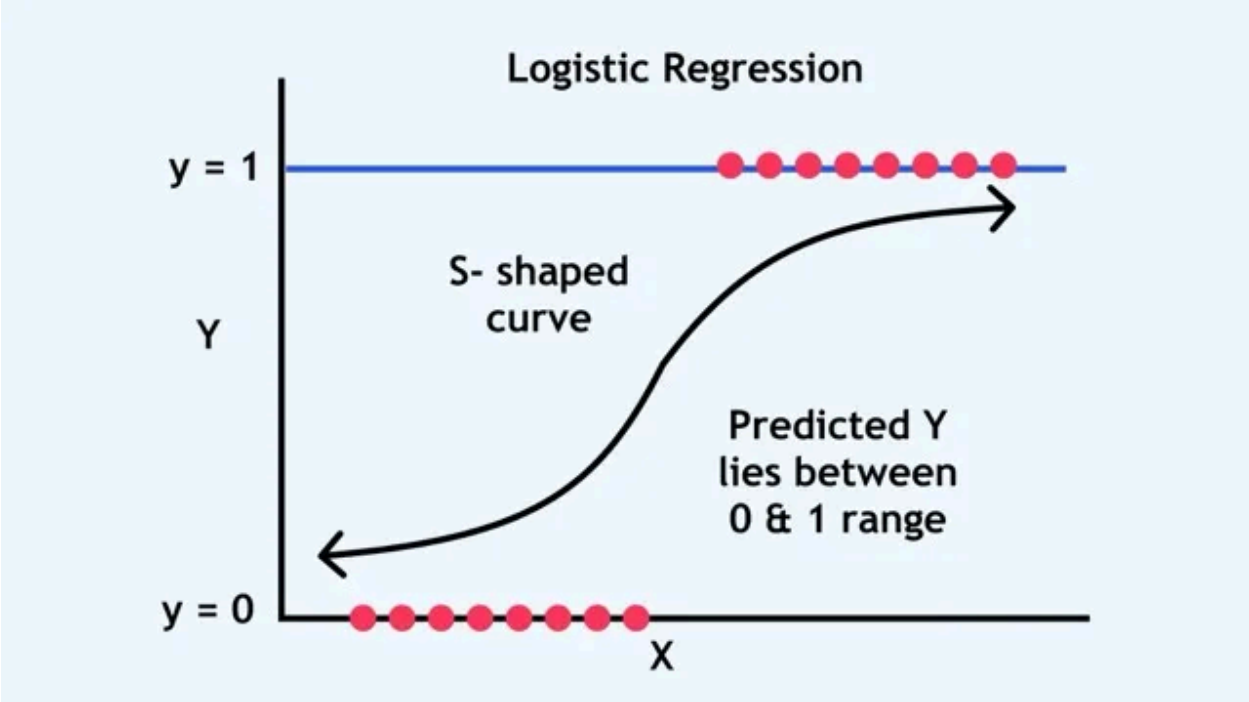


Figure 1: Logistic regression

Core Function: The Logistic Regression is used to estimate the likelihood of a sequence of cases to be of a specific type (heart disease in this case).

Mathematical Representation:

y = 1 / (1 + exp(-(wX + b))) where:

y: Predicted probability (between 0 and 1)

X: Input features (e.g., age, blood pressure, cholesterol)

w: Weights assigned to each feature

b: Bias term (intercept)

Training: The model is trained to find the most suitable value of weights (w) and bias (b) by minimizing the loss function which is the cross-entropy

3.2.2. (Model - Decision Tree Classifier):

One of the most used options during the development of every classification system or prediction of target variables using several factors in data mining is the decision trees. They operate using a dataset which is segmented into small branchy parts to form an inverted tree structure which includes a root node, internal nodes and leaf nodes.

A good thing about decision trees is that decision trees are non-parametric: they do not assume anything about the structure of the underlying data. Due to it, they have been efficient with large and intricate datasets (SONG, 2015).The decision tree model is trained using training data and the validation data may assist in deciding the optimal size of the final tree to prevent over and underfitting.

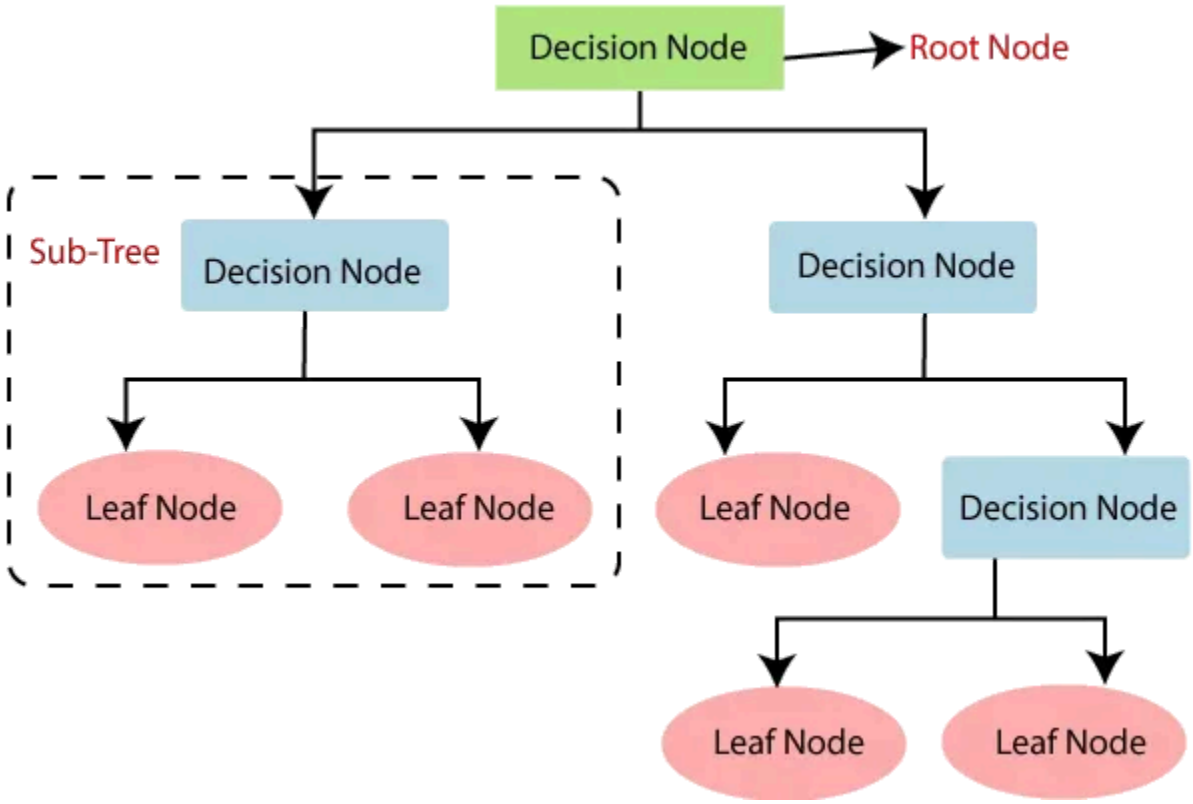


Figure 2: Decision Tree

The Idea: Decision Trees The decision trees are recursive algorithms that divide the data into a tree-like form on the basis of the values of features.

Decision Rules: Each one of the tree nodes is a choice rulein terms of a feature and a number (e.g. If age is less than 40, go left, do not go right).

Impurity Measures:

Gini Index: The Gini Index is used to measure the likelihood of making an error in classifying a randomly chosen sample.

Entropy: It is a measure of disorder or impurity of the data.

Training: The tree is constructed by choosing the feature and threshold at each node which gives the smallest impurity of the child nodes.

3.2.3. (Model - Random Forest Classifier):

The Random Forest Classifier is nothing but a collection of decision trees, each of which categorizes the data based on certain characteristics. However, despite their simplicity and ease of usage, the decision trees can lose their accuracy in working with large and complicated datasets where the variables are highly interactive with one another.

To get over this problem, random forests create a multitude of decision trees that use different subsets of data and predictor variables. Results from those trees are then summed up to make a final prediction (Jaime Lynn Speiser, 2019). The final prediction is then based on the results of all these trees. This method greatly enhances the accuracy of the model in comparison to a single decision tree, and at the same time, it allows one to see which predictors have the most significant impact on the outcome.

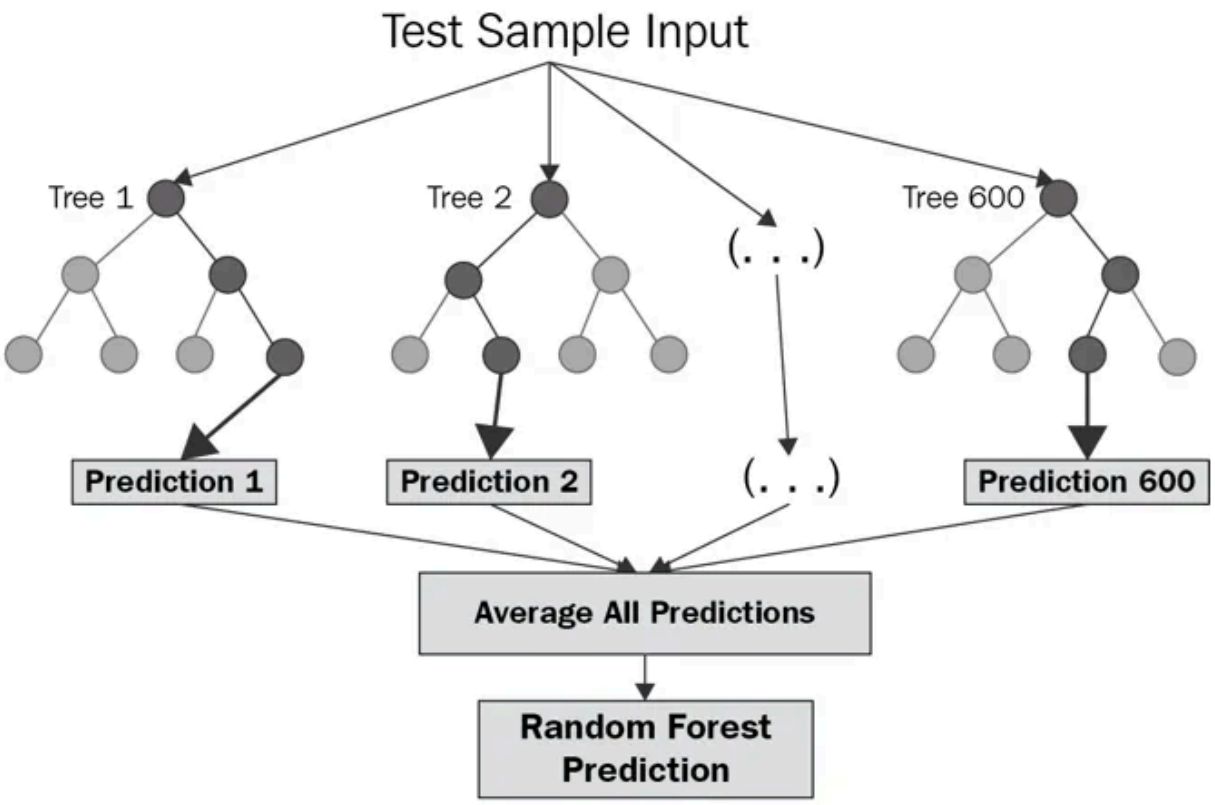


Figure 3: Random Forest Classifier

Ensemble Method: Random Forest It is ensemble learning that considers several decision trees in combination.

Key Features:

- **Bagging:** This is done by training the tree on each tree in the forest by randomly selecting a bootstrap sample of the training data.
- **Feature Randomness:** In splitting any node, it simply uses a random subset of the features. Forecasting: Prediction involves combining the predicted values of all the various trees; e.g. majority voting.

3.3. Pseudocode of Solution

```
START
LOAD dataset "cardiovascular_disease.csv"

DEFINE FUNCTION PREPROCESS_DATA(dataset)
  REMOVE unnecessary columns

  IF missing values exist THEN
    FOR EACH affected column DO
      IF column is numeric THEN
        FILL missing values with mean
      ELSE
        FILL missing values with mode
      END IF
    END FOR
  END IF

  REMOVE duplicate rows if any
  ENCODE categorical features
  RETURN cleaned_dataset
END FUNCTION
```

dataset_clean ← **PREPROCESS_DATA**(dataset)

PLOT feature distribution histogram
PLOT correlation heatmap

SPLIT dataset_clean into X (features) and y (target)

DEFINE FUNCTION SPLIT_DATA(X, y)
  SPLIT data into training (80%) and testing (20%)
  RETURN X_train, X_test, y_train, y_test
END FUNCTION

X_train, X_test, y_train, y_test ← **SPLIT_DATA**(X, y)

DEFINE FUNCTION SCALE_DATA(X_train, X_test)
  INITIALIZE StandardScaler
  FIT on X_train and **TRANSFORM** both sets
  RETURN scaled X_train and X_test
END FUNCTION

X_train_scaled, X_test_scaled ← **SCALE_DATA**(X_train, X_test)

DEFINE FUNCTION TRAIN_MODEL(model, X, y)
  TRAIN model
  RETURN trained_model
END FUNCTION

DEFINE FUNCTION EVALUATE_MODEL(model, X, y, name)
  PREDICT values
  CALCULATE accuracy, precision, recall, and F1-score
  PLOT confusion matrix

RETURN metrics
END FUNCTION

TRAIN and **EVALUATE** Logistic Regression
TRAIN and **EVALUATE** Decision Tree
TRAIN and **EVALUATE** Random Forest

DEFINE healthy_input and high_risk_input
SCALE inputs and **GENERATE** predictions
DISPLAY predictions

COMPARE model performance using charts
SELECT best model based on performance

IF best model exists **THEN**
DISPLAY model details and predictions
ELSE
DISPLAY error message
END IF

END

3.4. Data Description

The Cardiovascular Disease Dataset also has a range of attributes that are associated with cardiovascular health. These characteristics are gathered on behalf of every patient and are applied to foresee whether one is at risk of cardiovascular disease.

Table 1: Data Description

Feature	Description
patientid	Unique identifier for each patient (excluded from analysis)
age	Age of the patient
gender	Sex of the patient (0 = Female, 1 = Male)
chestpain	Type of chest pain experienced by the patient (values range from 0 to 3)
restingBP	Resting blood pressure measured in mm Hg
serumcholesterol	Serum cholesterol level measured in mg/dl
fastingbloodsugar	Indicates whether fasting blood sugar is greater than 120 mg/dl (1 = True, 0 = False)
restingelectro	Resting electrocardiographic results (0, 1, or 2)
maxheartrate	Maximum heart rate achieved during the exercise test
exerciseangia	Indicates presence of exercise-induced angina (1 = Yes, 0 = No)
oldpeak	ST depression induced by exercise relative to rest
slope	Slope of the peak exercise ST segment
noofmajorvessels	Number of major vessels colored by fluoroscopy
target	Indicates presence of cardiovascular disease (1 = Yes, 0 = No)

3.5. Diagrammatical Representation of the Solution

3.5.1. Flow Chart

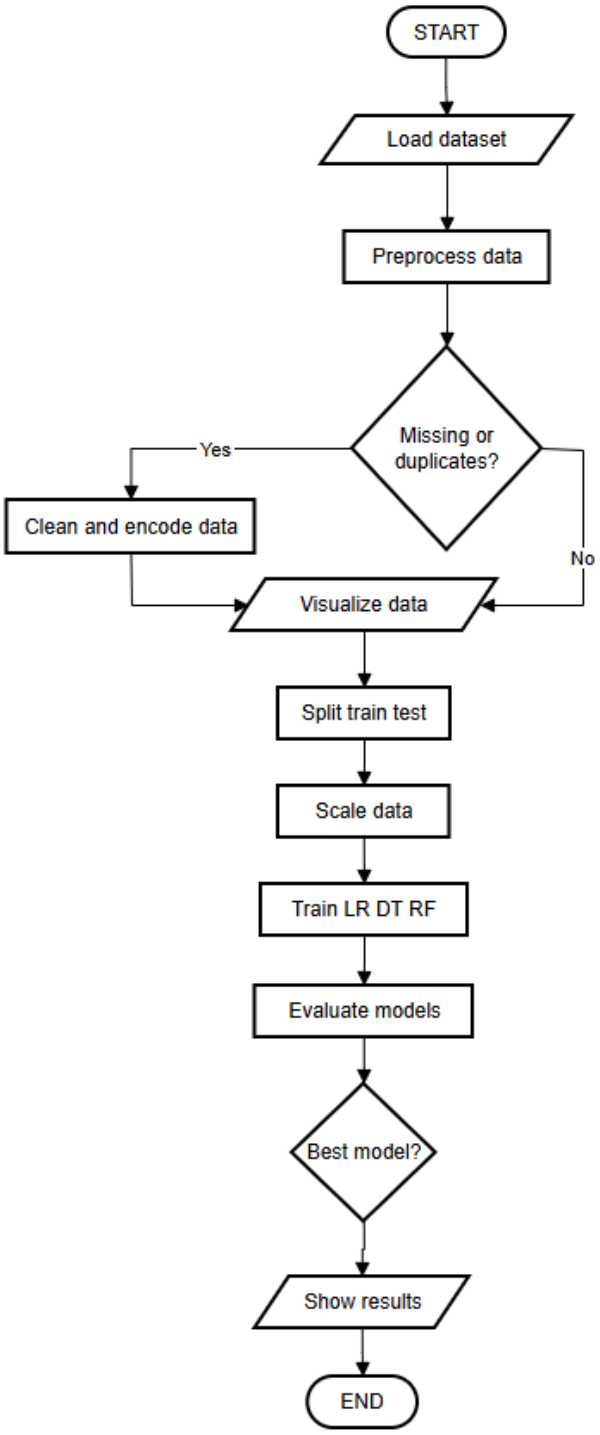


Figure 4: FlowChart

3.5.2. State Transition Diagram

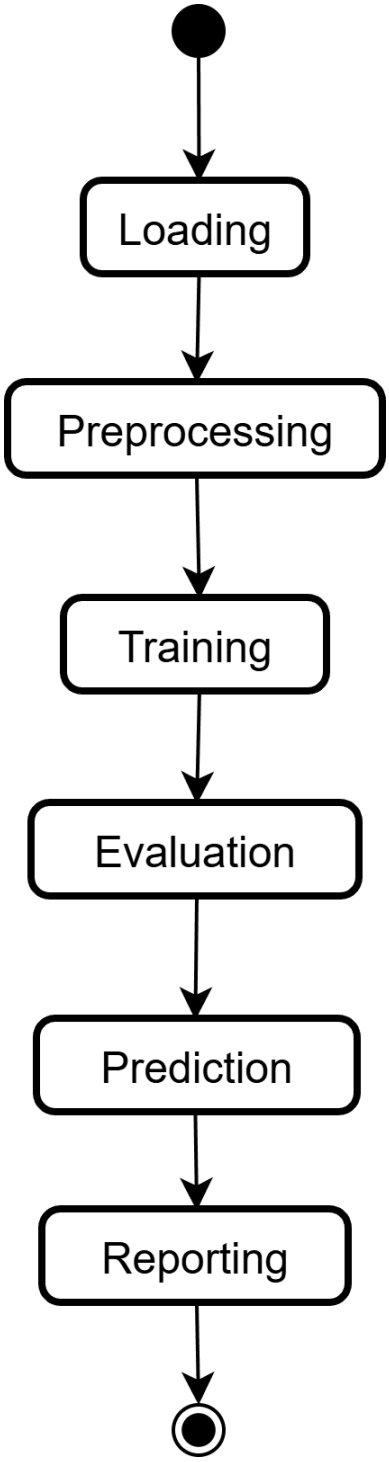


Figure 5: State Transition Diagram

3.6. Tools & Environment Used

3.6.1. Jupyter Notebook

The cardiovascular disease prediction system's installation along with the entire experimental analysis were performed on **Jupyter Notebook**. Jupyter Notebook was a digital laboratory that made possible, the interactive, easy and efficient data exploration, model development, result visualization, and machine learning workflow step by step execution.

Figure 6: Jupyter Notebook

3.6.2. Python Environment

The system was developed using Python, and the selection of Python was due to its entire ecosystem containing machine learning and data science libraries. Python was the one through which data went in and out pre-processing, feature engineering, training the model, evaluating its performance, and finally, visualization of results went through this language.

Figure 7: Python

Modules / Libraries Used

3.6.3. Pandas Library

Pandas library was the one of choice when it came to data manipulation and analysis. One of its strong points was that it allowed the efficient working with structured data, meaning in the case of cardiovascular data set loading for the analysis, cleaning missing values, removing duplicates and managing feature columns among other things, all this could be done tree-fast because of Pandas.

Figure 8: Pandas Library

Import: import pandas as pd

3.6.4. NumPy Library

The use of the **NumPy** library was a major breakthrough in the field of numerical computations and array operations. It not only guaranteed mathematical calculations but also rendered the handling of large portions of numerical data highly efficient, which was the case during preprocessing and model training.

Figure 9: NumPy Library

Import: import numpy as np

3.6.5. Matplotlib Library

Matplotlib was the library that facilitated the visual output of the dataset in the form of histograms as well as performance comparison plots. These visual representations were essential to see the data distribution and the outcomes of the model analysis.

Figure 10: Matplotlib Library

Import: import matplotlib.pyplot as plt

3.6.6. Seaborn Library

Through Seaborn, the Python library that was responsible for making the statistical visuals more advanced, one could plot correlation heatmaps and confusion matrixes. Thus, feature relationships and the classification performance were more easily interpreted..

Figure 11: Seaborn Library

Import: import seaborn as sns

3.6.7. Scikit-Learn Library

Scikit-Learn was the library that was chosen and used as the core of the development of the models and the evaluation of their performance. It presented a package of tools that comprised of preprocessing tools, classification model training tools as well as performance metrics.

Figure 12: Scikit-Learn Library

Submodules Used:

- **Traintestsplit -** In order to split the data into both training and testing sets.
- **LabelEncoder -** This is used to transform categorized variables into a number form.

- **MinimaxSelector -** To achieve standardization of the features and feature scaling.
- **LogisticRegression -** To construct the logistic regression classification model
- **Decision Tree Classifier -** To build the decision tree classifier.
- **RandomForestClassifier -** To develop the random forest ensemble classifier.
- **Accuracyscore -** To be used in determining the general classification accuracy.
- **Confusionmatrix -** To analyze true predictions and false predictions.
- **Classification report -** To compute precision, recall and F1-score.

4. Results Achieved & Development Process

4.1. Imports

Python libraries were imported in order to handle, visualize, train a machine learning model and evaluate it.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

import seaborn as sns
sns.set_theme(style='whitegrid', palette='Set2')
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import confusion_matrix, accuracy_score, classification_report
from sklearn.preprocessing import StandardScaler
```

Figure 13: Imports

4.2. Reading the Dataset

The cardiovascular disease data was read out of a CSV file into a Pandas DataFrame to analyze it.

```
data = pd.read_csv('Cardiovascular_Disease_Dataset.csv')
print("Dataset Preview:")
data.head()
```

Dataset Preview:

	patientid	age	gender	chestpain	restingBP	serumcholestrol	fastingbloodsugar	restingrelectro	maxheartrate	exerciseangia	oldpeak
0	103368	53	1	2	171	0	0	1	147	0	5.3
1	119250	40	1	0	94	229	0	1	115	0	3.7
2	119372	49	1	2	133	142	0	0	202	1	5.0
3	132514	43	1	0	138	295	1	1	153	0	3.2
4	146211	31	1	1	199	0	0	2	136	0	5.3

Next steps: Generate code with data | New interactive sheet

Figure 14: Dataset Loaded (CSV Preview)

4.3. Dropping Patient Identifier

The column patientid was deleted as this is a unique identifier and does not help in prediction.

```
if 'patientid' in data.columns:
    data = data.drop(['patientid'], axis=1)
data
```

	age	gender	chestpain	restingBP	serumcholestrol	fastingbloodsugar	restingrelectro	maxheartrate	exerciseangia	oldpeak	slope	no
0	53	1	2	171	0	0	1	147	0	5.3	3	
1	40	1	0	94	229	0	1	115	0	3.7	1	
2	49	1	2	133	142	0	0	202	1	5.0	1	
3	43	1	0	138	295	1	1	153	0	3.2	2	
4	31	1	1	199	0	0	2	136	0	5.3	3	
...	...	...	...	...	...	...	...	...	...	...	...	
995	48	1	2	139	349	0	2	183	1	5.6	2	
996	47	1	3	143	258	1	1	98	1	5.7	1	
997	69	1	0	156	434	1	0	196	0	1.4	3	
998	45	1	1	186	417	0	1	117	1	5.9	3	
999	25	1	0	158	270	0	0	143	1	4.7	0	

1000 rows × 13 columns

Next steps: Generate code with data | New interactive sheet

Figure 15: Dropping patient id Column

4.4. Missing Values Check

Isnull () and a heatmap visualization were used to verify the presence of missing or null values in the data set. No missing values were found.

```
missing_values = data.isnull().sum()

print("\nChecking for missing values:")
print(missing_values)


sns.heatmap(data.isnull(), cbar=False, cmap='plasma')
plt.title("Missing Values Heatmap")

# Check for missing values and handle accordingly
if missing_values.sum() == 0:
    print("No missing values were found")
else:
    print("There are missing values in the dataset.")
plt.show()
```

```
Checking for missing values:
age                 0
gender              0
chestpain           0
restingBP           0
serumcholestrol     0
fastingbloodsugar   0
restingrelectro     0
maxheartrate        0
exerciseangia       0
oldpeak             0
slope               0
noofmajorvessels    0
target              0
dtype: int64
No missing values were found
```

Figure 16: Missing Values Summary

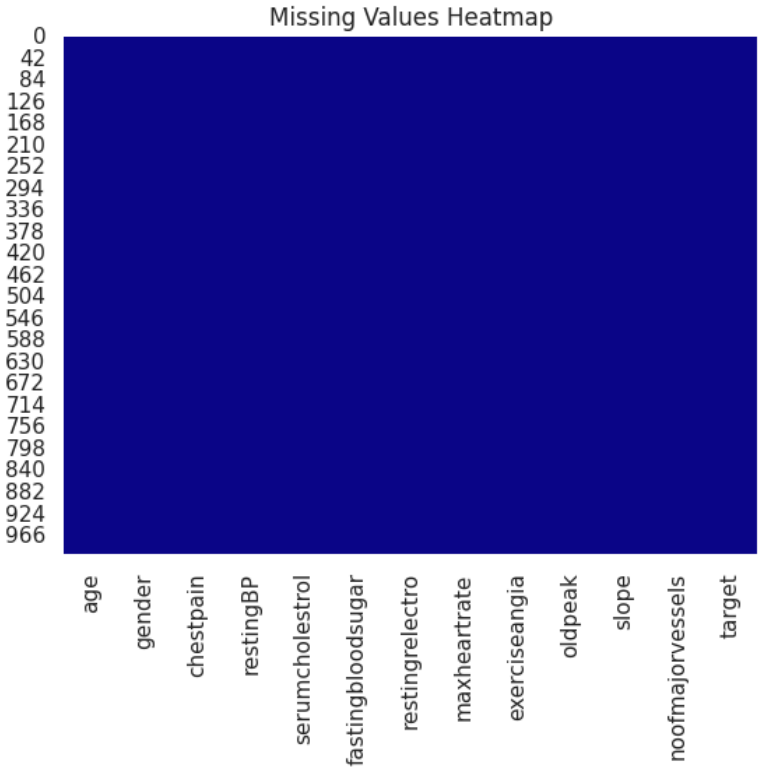


Figure 17: Missing Values Heatmap

4.5. Duplicate Values Check

Duplicated was used to check the duplicate rows to maintain quality data. No duplicate rows were found.

```
duplicate_rows = data.duplicated()

print("\nChecking for duplicate values:")
print(f"Total duplicate rows: {duplicate_rows.sum()}")


if duplicate_rows.sum() > 0:
    print("Duplicate rows found!")
    print(data[duplicate_rows])
else:
    print("No duplicate rows were found.")
```

```
Checking for duplicate values:
Total duplicate rows: 0
No duplicate rows were found.
```

Figure 18: Duplicate Rows Check

4.6. Feature Distribution

Histograms were created to see how each feature is distributed and to have a general idea on how the data is spread.

```
data.hist(bins=15, figsize=(15, 10), color='crimson', edgecolor='black')
plt.suptitle("Feature Distributions")
plt.show()
```

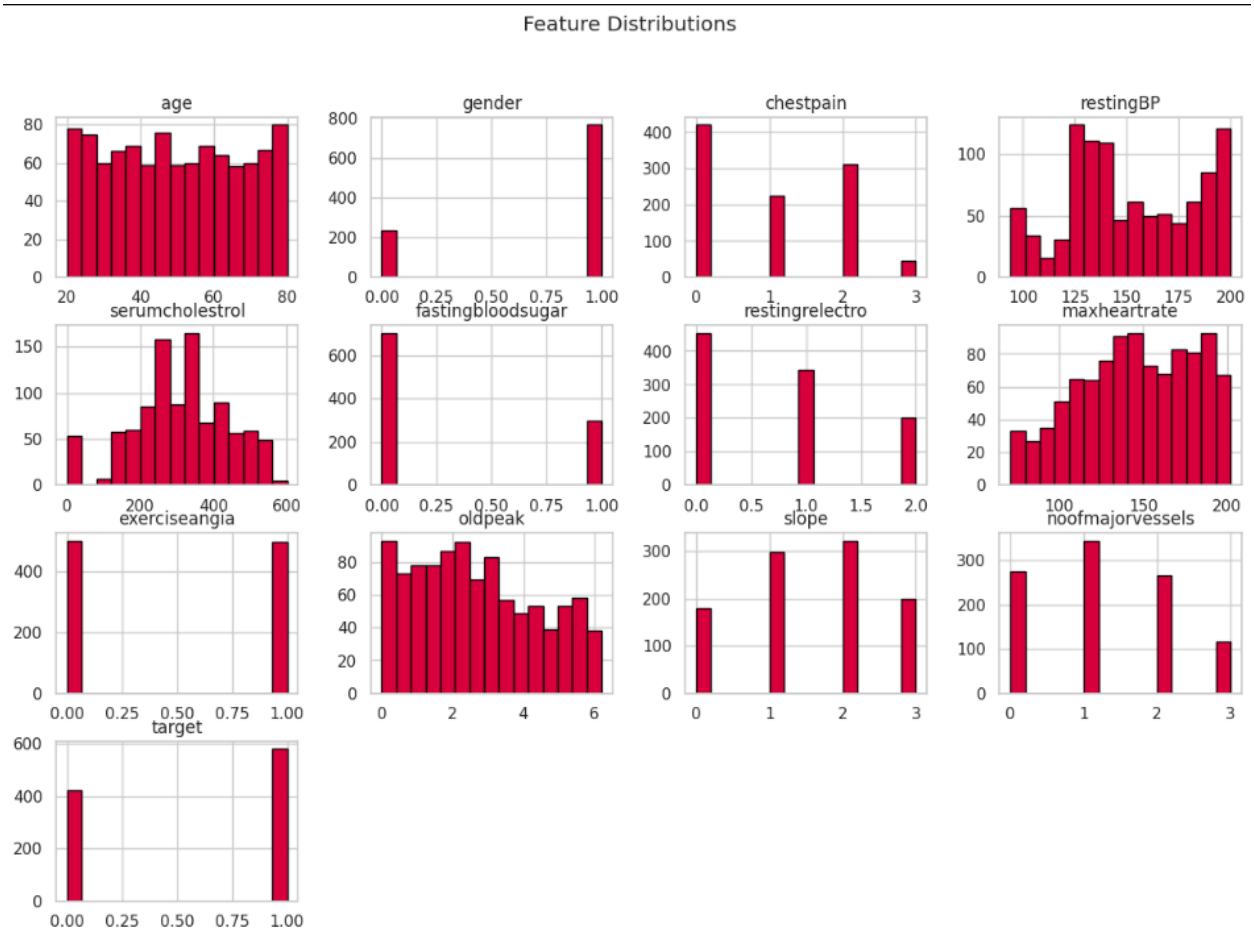


Figure 19: Feature Distributions

4.7. Correlation Heatmap

A heatmap was used to show the correlation among the features and the variables that are highly correlated.

```
plt.figure(figsize=(12, 8))
sns.heatmap(data.corr(), annot=True, cmap='plasma', fmt='.2f')
plt.title("Correlation Heatmap")
plt.show()
```

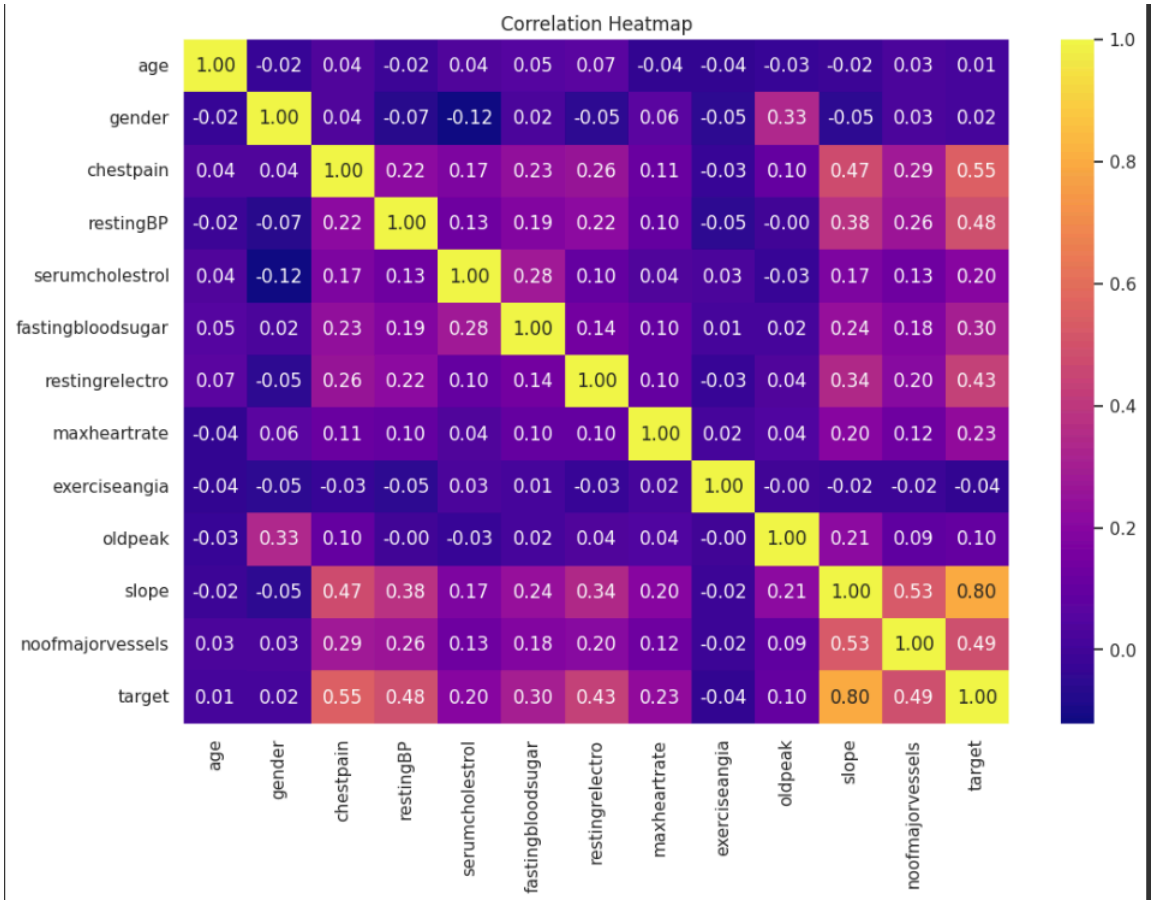


Figure 20: Correlation Heatmap

4.8. Train/Test Split

The data was split into two parts, the input features X and the target label y, and split into training and testing data at an 80/20 ratio to measure the performance of the model.

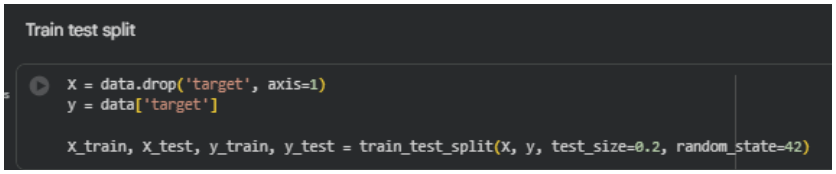


Figure 21: Train/Test Split

4.9. Scaling

The StandardScaler was used to standardize the values of the features then the Logistic Regression model was trained. Scaling is used to make sure the models are not learned by features with greater values.

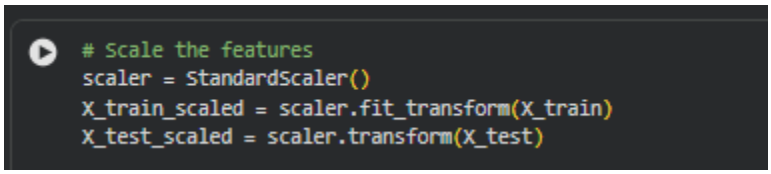


Figure 22: Feature Scaling using StandardScaler

4.10. Training Models

4.10.1. Logistic Regression

Logistic Regression classifier was prepared on the scaled training data with accuracy, confusion matrix and classification report as a measure of evaluation.

```
# Logistic Regression with scaled data
lr_model = LogisticRegression()
lr_model.fit(X_train_scaled, y_train)
lr_pred = lr_model.predict(X_test_scaled)

lr_accuracy = accuracy_score(y_test, lr_pred)
lr_cm = confusion_matrix(y_test, lr_pred)

print(f"Accuracy: {lr_accuracy:.2f}")
print("Classification Report:")
print(classification_report(y_test, lr_pred))

plt.figure(figsize=(6, 4))
sns.heatmap(lr_cm, annot=True, fmt='d', cmap='plasma', cbar=False)
plt.title("Logistic Regression - Confusion Matrix")
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.show()
```

```
Accuracy: 0.96
Classification Report:
              precision    recall  f1-score   support

           0       0.96      0.95      0.96        83
           1       0.97      0.97      0.97       117

    accuracy                           0.96       200
   macro avg       0.96      0.96      0.96       200
weighted avg       0.96      0.96      0.96       200
```

Figure 23: Logistic Regression Model Training

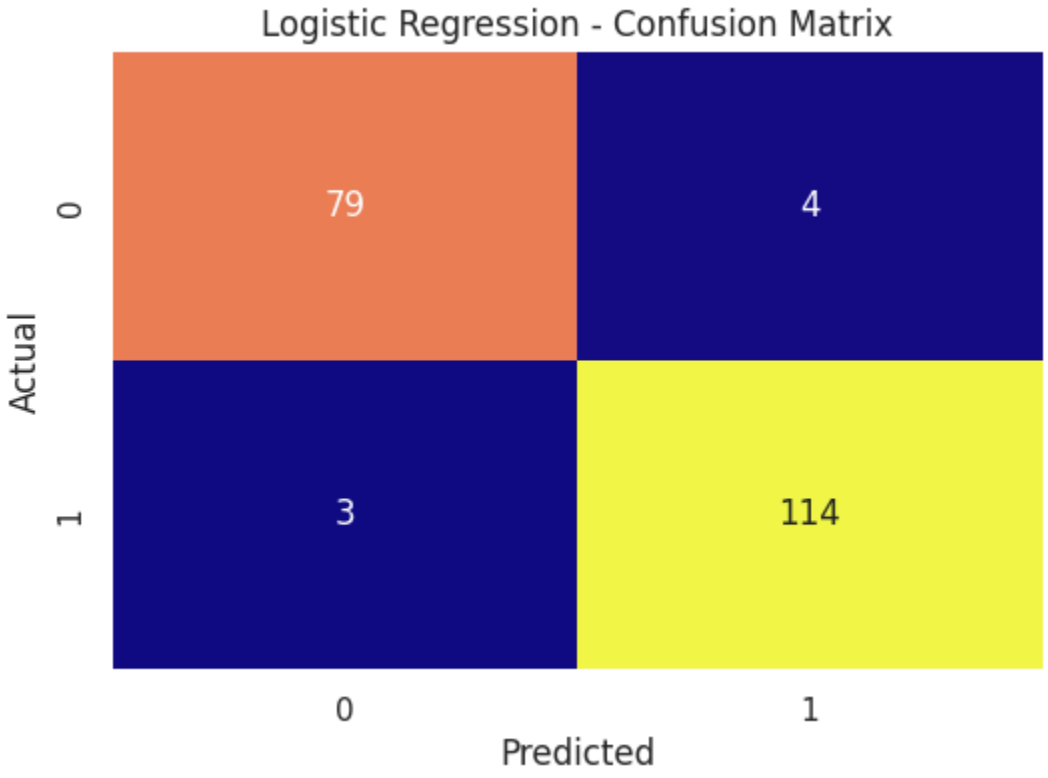


Figure 24: Logistic Regression Confusion Matrix

4.10.2. Decision Tree Classifier

The Decision Tree classifier was trained on the original (unscaled) data and tested with the help of the accuracy, confusion, and classification report.

```
# Decision Tree Classifier
print("Decision Tree:")
dt_model = DecisionTreeClassifier()
dt_model.fit(X_train, y_train)
dt_pred = dt_model.predict(X_test)

dt_accuracy = accuracy_score(y_test, dt_pred)
dt_cm = confusion_matrix(y_test, dt_pred)

print(f"Accuracy: {dt_accuracy:.2f}")
print("Classification Report:")
print(classification_report(y_test, dt_pred))

plt.figure(figsize=(6, 4))
sns.heatmap(dt_cm, annot=True, fmt='d', cmap='plasma', cbar=False)
plt.title("Decision Tree - Confusion Matrix")
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.show()
```

```
Decision Tree:
Accuracy: 0.96
Classification Report:
              precision    recall  f1-score   support

           0       0.97      0.93      0.95        83
           1       0.95      0.98      0.97       117

    accuracy                           0.96       200
   macro avg       0.96      0.96      0.96       200
weighted avg       0.96      0.96      0.96       200
```

Figure 25: Decision Tree Model Training

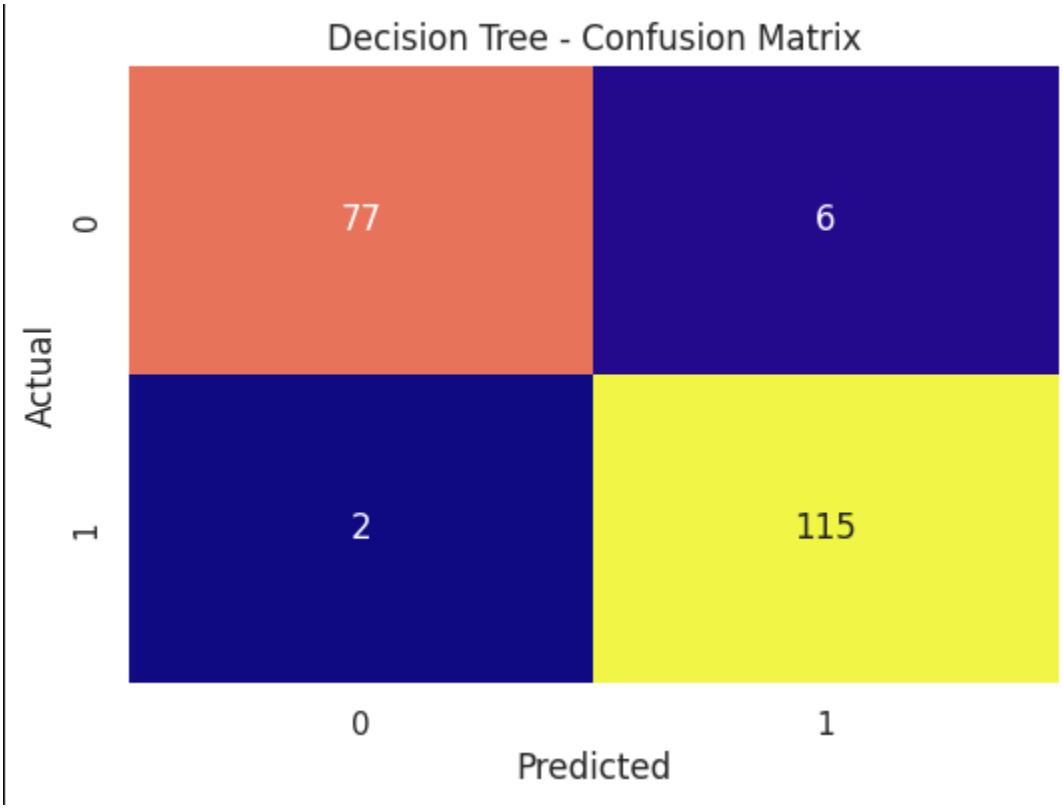


Figure 26: Decision Tree Confusion Matrix

4.10.3. Random Forest Classifier

Random Forest classifier was fitted on the data and measured in terms of accuracy, confusion matrix and classification report. This model has the highest accuracy compared to the tested models.

```
# Random Forest Classifier
print("Random Forest:")
rf_model = RandomForestClassifier()
rf_model.fit(X_train, y_train)
rf_pred = rf_model.predict(X_test)

rf_accuracy = accuracy_score(y_test, rf_pred)
rf_cm = confusion_matrix(y_test, rf_pred)

print(f"Accuracy: {rf_accuracy:.2f}")
print("Classification Report:")
print(classification_report(y_test, rf_pred))

plt.figure(figsize=(6, 4))
sns.heatmap(rf_cm, annot=True, fmt='d', cmap='plasma', cbar=False)
plt.title("Random Forest - Confusion Matrix")
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.show()
```

```
Random Forest:
Accuracy: 0.98
Classification Report:
              precision    recall  f1-score   support

           0       0.98      0.98      0.98        83
           1       0.98      0.98      0.98       117

    accuracy                           0.98       200
   macro avg       0.98      0.98      0.98       200
weighted avg       0.98      0.98      0.98       200
```

Figure 27: Random Forest Model Training

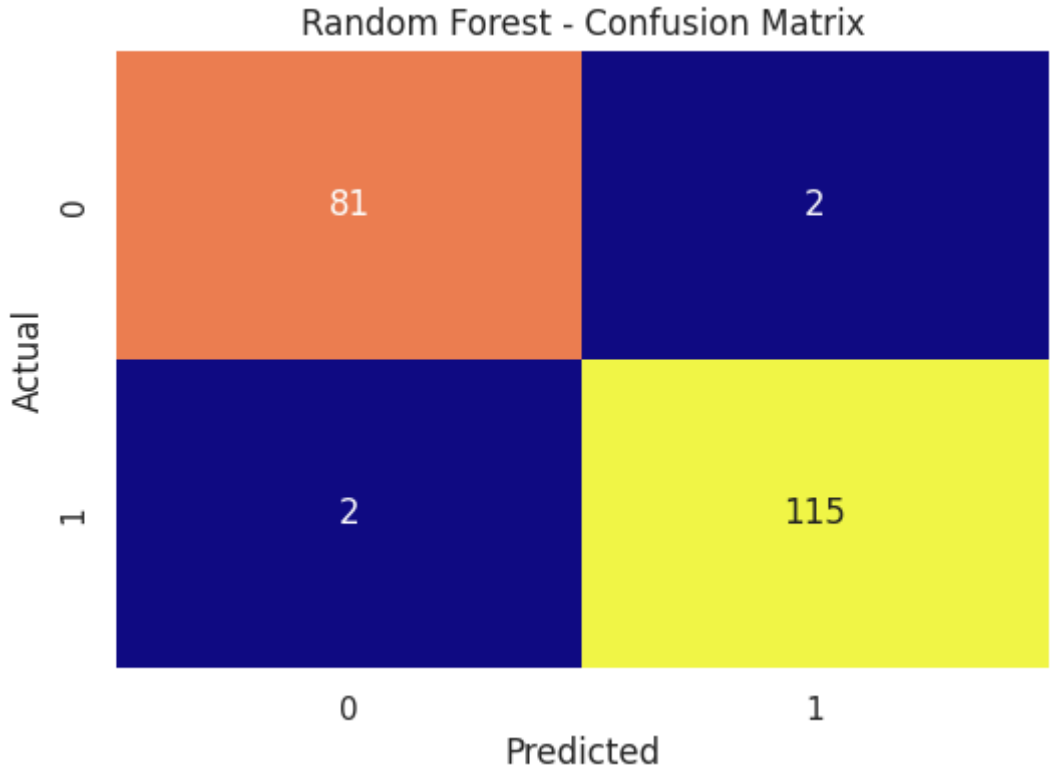


Figure 28: Random Forest Confusion Matrix

4.11. Sample Data for Testing

There were two sample inputs developed to test model predictions:

1. a normal healthy case, and
2. the case of a high-risk cardiovascular disease.

```
feature_names = ['age', 'gender', 'chestpain', 'restingBP', 'serumcholestrol', 'fastingbloodsugar',
                 'restingrelectro', 'maxheartrate','exerciseangia', 'oldpeak', 'slope', 'noofmajorvessels']

# Normal Data (Healthy Individual)
normal_input = pd.DataFrame([[30, 0, 0, 120, 200, 0, 0, 160, 0, 1.0, 1, 0]],
                            columns=feature_names)

# High-Risk Data (Cardiovascular Disease)
high_risk_input = pd.DataFrame([[60, 1, 3, 180, 300, 1, 2, 140, 1, 3.0, 2, 3]],
                               columns=feature_names)
```

Figure 29: Sample Input Data (Normal vs High-Risk)

4.12. Prediction on New Inputs

4.12.1. Logistic Regression Prediction

The same fitted scaler was used to rescaling the sample inputs and then the sample was predicted by use of the Logistic Regression.

```
# Logistic Regression Prediction

# Scale the input data
normal_input_scaled = scaler.transform(normal_input)
high_risk_input_scaled = scaler.transform(high_risk_input)

# Normal Data (Healthy Individual)
normal_prediction_lr = lr_model.predict(normal_input_scaled)
print(f"Logistic Regression - Normal Data Prediction: {'Cardiovascular Disease' if normal_prediction_lr[0] == 1 else 'No Disease'}")

# High-Risk Data (Cardiovascular Disease)
high_risk_prediction_lr = lr_model.predict(high_risk_input_scaled)
print(f"Logistic Regression - High-Risk Data Prediction: {'Cardiovascular Disease' if high_risk_prediction_lr[0] == 1 else 'No Disease'}")

Logistic Regression - Normal Data Prediction: No Disease
Logistic Regression - High-Risk Data Prediction: Cardiovascular Disease
```

Figure 30: Logistic Regression Predictions for New Inputs

4.12.2. Decision Tree Prediction

The sample inputs were predicted using the Decision Tree classifier.

```
# Decision Tree Classifier Prediction

# Normal Data (Healthy Individual)
normal_prediction_dt = dt_model.predict(normal_input)
print(f"Decision Tree - Normal Data Prediction: {'Cardiovascular Disease' if normal_prediction_dt[0] == 1 else 'No Disease'}")

# High-Risk Data (Cardiovascular Disease)
high_risk_prediction_dt = dt_model.predict(high_risk_input)
print(f"Decision Tree - High-Risk Data Prediction: {'Cardiovascular Disease' if high_risk_prediction_dt[0] == 1 else 'No Disease'}")

Decision Tree - Normal Data Prediction: No Disease
Decision Tree - High-Risk Data Prediction: Cardiovascular Disease
```

Figure 31: Decision Tree Predictions for New Inputs

4.12.3. Random Forest Prediction

The sample inputs were predicted using the Random Forest classifier.

```
# Random Forest Prediction

# Normal Data (Healthy Individual)
normal_prediction = rf_model.predict(normal_input)
print(f"Normal Data Prediction: {'Cardiovascular Disease' if normal_prediction[0] == 1 else 'No Disease'}")

# High-Risk Data (Cardiovascular Disease)
high_risk_prediction = rf_model.predict(high_risk_input)
print(f"High-Risk Data Prediction: {'Cardiovascular Disease' if high_risk_prediction[0] == 1 else 'No Disease'}")
```

```
Normal Data Prediction: No Disease
High-Risk Data Prediction: Cardiovascular Disease
```

Figure 32: Random Forest Predictions for New Inputs

4.13. Model Comparison using Bar Graph & Pie-Chart

The trained models were compared using their performance metrics (accuracy, precision, recall, and F1-score). A bar chart and pie chart were used to visualize the comparison.

```
# Define model names
models = ['Logistic Regression', 'Decision Tree', 'Random Forest']

# Metrics for each model
accuracy = [0.96, 0.96, 0.98]
precision_class_0 = [0.96, 0.97, 1.00]
recall_class_0 = [0.95, 0.94, 0.96]
f1_class_0 = [0.96, 0.96, 0.98]

precision_class_1 = [0.97, 0.96, 0.97]
recall_class_1 = [0.97, 0.98, 1.00]
f1_class_1 = [0.97, 0.97, 0.99]


x = np.arange(len(models))
width = 0.2

fig, ax = plt.subplots(figsize=(12, 6))


rects1 = ax.bar(x - width*1.5, accuracy, width, label='Accuracy', color='darkgreen')
rects2 = ax.bar(x - width/2, precision_class_1, width, label='Precision (Class 1)', color='darkgreen')
rects3 = ax.bar(x + width/2, recall_class_1, width, label='Recall (Class 1)', color='darkgreen')
rects4 = ax.bar(x + width*1.5, f1_class_1, width, label='F1-Score (Class 1)', color='darkgreen')


ax.set_xlabel('Models')
ax.set_ylabel('Scores')
ax.set_title('Model Comparison: Logistic Regression vs Decision Tree vs Random Forest')
ax.set_xticks(x)
ax.set_xticklabels(models)
ax.legend()

# Add value annotations
def add_annotations(rects):
    for rect in rects:
        height = rect.get_height()
        ax.annotate(f'{height:.2f}',
                    xy=(rect.get_x() + rect.get_width() / 2, height),
                    xytext=(0, 3),
                    textcoords="offset points",
                    ha='center', va='bottom')

add_annotations(rects1)
add_annotations(rects2)
add_annotations(rects3)
add_annotations(rects4)

# Show plot
plt.tight_layout()
plt.show()
```

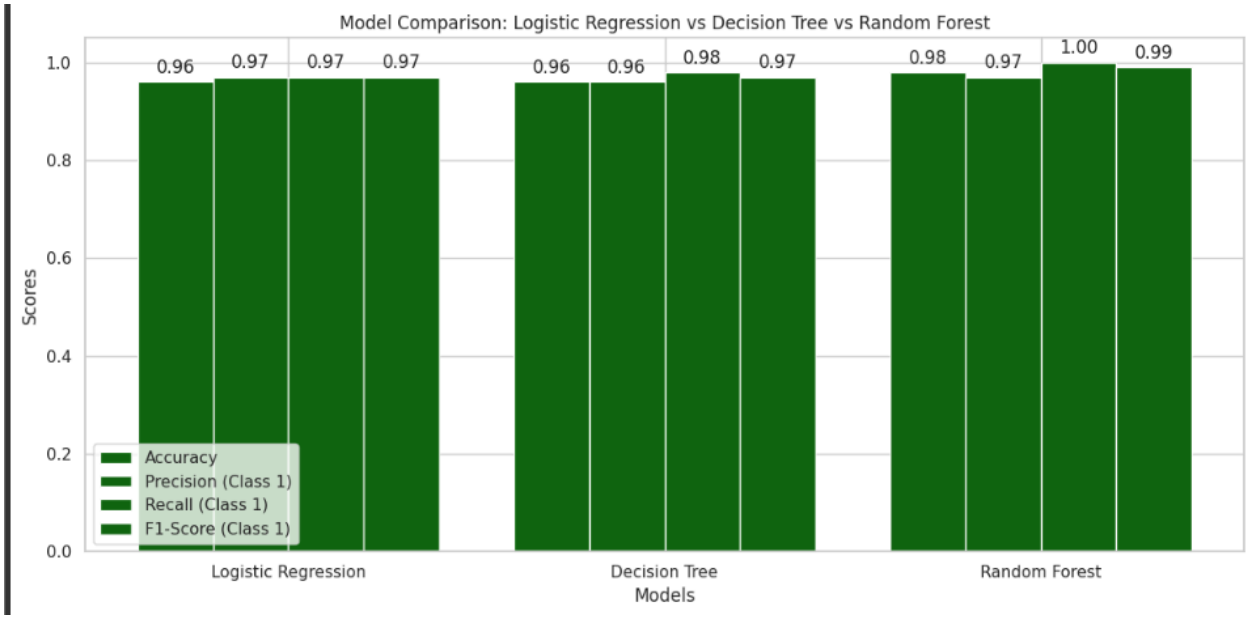


Figure 33: Model Comparison Bar Chart

```
models = ['Logistic Regression', 'Decision Tree', 'Random Forest']
accuracy = [0.96, 0.96, 0.98]


plt.figure(figsize=(8, 6))
plt.pie(accuracy, labels=models, autopct='%1.1f%%', startangle=90, colors=['darkgreen', 'crimson', 'salmon'])
plt.title('Model Accuracy Comparison')
plt.axis('equal')


plt.show()
```

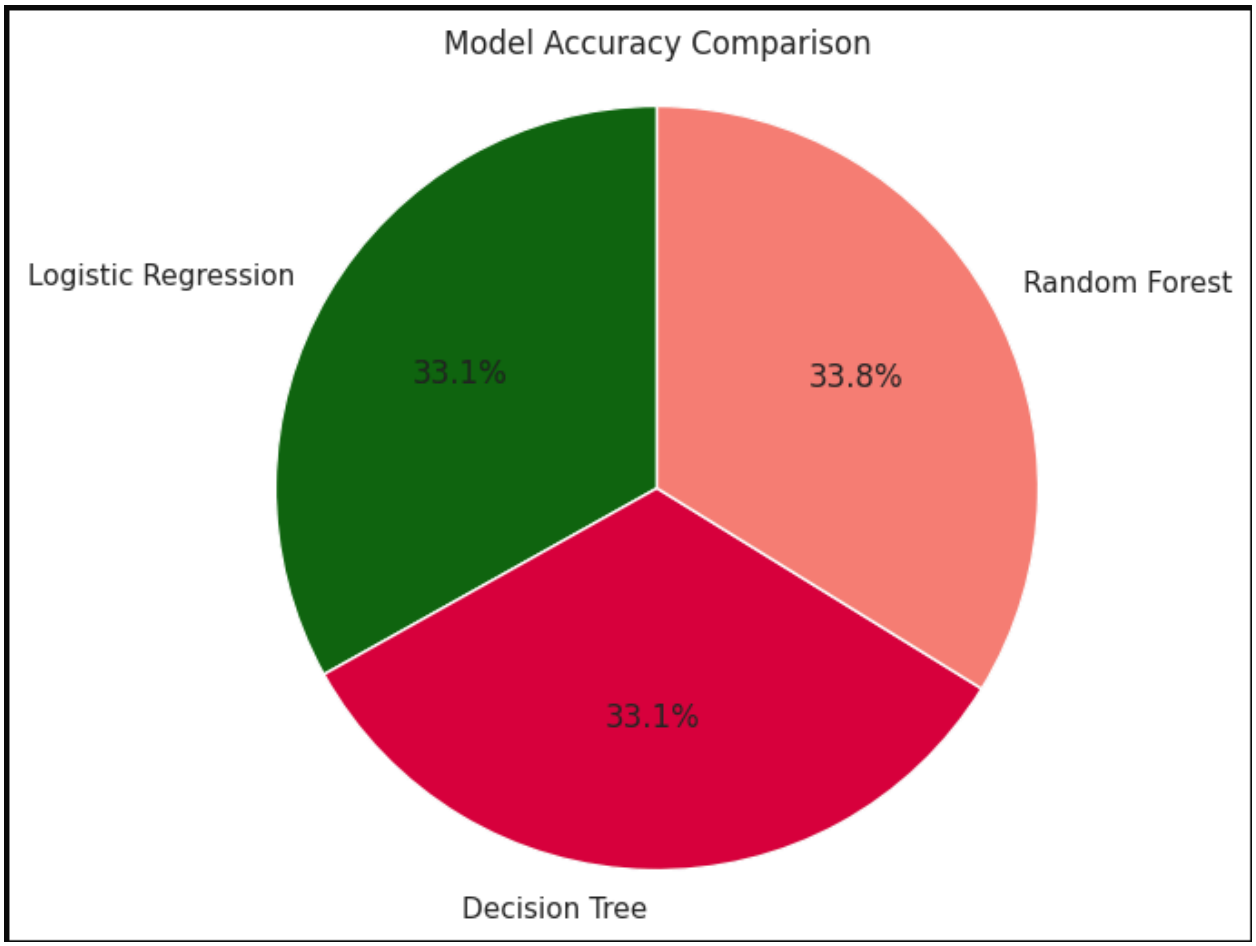


Figure 34: Model Accuracy Comparison Pie Chart

4.14. CUI for CVD CardioVascular Diseases Prediction

Code

```
import os
from pathlib import Path

import joblib
import pandas as pd
import numpy as np

from sklearn.model_selection import train_test_split
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.ensemble import RandomForestClassifier


# -------- CONFIG --------
CSV_PATHS = [
    Path("Cardiovascular_Disease_Dataset.csv"),
    Path("/mnt/data/Cardiovascular_Disease_Dataset.csv"),
]
TARGET = "target"
DROP_COLS = ["patientid"]          # dropped in your notebook:contentReference[oaicite:4]{index=4}
MODEL_PATH = "cvd_best_model.joblib"
RANDOM_STATE = 42


def _parse_value(x: str):
    x = x.strip()
    if x == "":
        return None
    try:
        return int(x)
    except:
        try:
            return float(x)
        except:
            return x
```

,

```
def _find_csv():
    for p in CSV_PATHS:
        if p.exists():
            return p
    raise FileNotFoundError("Cardiovascular_Disease_Dataset.csv not found.")


def _drop_unnamed(df: pd.DataFrame) -> pd.DataFrame:
    return df.loc[:, ~df.columns.astype(str).str.lower().str.startswith("unnamed")]


def _train_and_save(csv_path: Path):
    df = pd.read_csv(csv_path)
    df = _drop_unnamed(df)

    # drop patientid if exists (same as notebook)
    for c in DROP_COLS:
        if c in df.columns:
            df = df.drop(columns=[c])

    if TARGET not in df.columns:
        raise ValueError(f"Target column '{TARGET}' not found in CSV. Available: {list(df.columns)}")

    X = df.drop(columns=[TARGET])
    y = df[TARGET].astype(int)

    # detect numeric/categorical
    num_cols = X.select_dtypes(include=[np.number]).columns.tolist()
    cat_cols = [c for c in X.columns if c not in num_cols]

    preprocess = ColumnTransformer(
        transformers=[
            ("num", Pipeline([
                ("imputer", SimpleImputer(strategy="median")),
                ("scaler", StandardScaler()),
            ]), num_cols),
```

```
                ("cat", Pipeline([
                    ("imputer", SimpleImputer(strategy="most_frequent")),
                    ("onehot", OneHotEncoder(handle_unknown="ignore")),
                ]), cat_cols),
            ],
            remainder="drop"
        )

        model = Pipeline([
            ("prep", preprocess),
            ("model", RandomForestClassifier(
                n_estimators=300,
                random_state=RANDOM_STATE,
                n_jobs=-1
            ))
        ])

        X_train, X_test, y_train, y_test = train_test_split(
            X, y, test_size=0.2, random_state=RANDOM_STATE, stratify=y
        )

        model.fit(X_train, y_train)
        joblib.dump(model, MODEL_PATH)
        return model


    def _load_or_train():
        if os.path.exists(MODEL_PATH):
            return joblib.load(MODEL_PATH)
        return _train_and_save(_find_csv())


    # -------------- RUN CUI --------------
    model = _load_or_train()

    feature_cols = list(model.feature_names_in_)
```

```
answers = {}
for col in feature_cols:
    answers[col] = _parse_value(input(f"{col}: "))

X_new = pd.DataFrame([answers])
pred = int(model.predict(X_new)[0])

# ONLY FINAL OUTPUT
print("CVD Detected" if pred == 1 else "No CVD")
5
```

Prediction output

```
...  age: 55
     gender: 1
     chestpain: 45
     restingBP: 34
     serumcholestrol: 56
     fastingbloodsugar: 76
     restingrelectro: 55
     maxheartrate: 3
     exerciseangia: 87
     oldpeak: 65
     slope: 45
     noofmajorvessels: 44
     CVD Detected
```

Figure 35: CVD CardioVascular Diseases Prediction

5. Conclusion

This study proposes possibilities regarding the use of machine learning technologies for the diagnosis and management of CVD, particularly congenital heart disease. A model for predicting heart diseases has been developed, working on some of the key challenges that include handling an imbalanced dataset, improving model performance, and increasing accuracy at a practical healthcare level. Advanced machine learning classifiers, such as (Random Forest, Decision tree,

and Logistic Regression), have been applied to develop models that can be reliable and accurate by applying feature engineering with proper data preprocessing to provide early and accurate diagnosis of cardiovascular diseases to medical professionals.

5.1. Evaluation of Previous Work

- **Presented Research:** The literature mostly talks about the role of machine learning in the prediction of cardiovascular diseases, and different algorithms, like Logistic Regression, Random Forest, and Decision Tree, promise well for early detection. Yet, the problems of dataset imbalance, overfitting, and optimization issues are still very frequent.
- **Proposed Enhancements:** The area of data preprocessing, missing values treatment, and feature selection has received a lot of attention in order to improve the model performance and to reduce the impact of the problems identified in the former studies..
- **Closing Gaps:** The strategy that has been suggested aims to enhance the reliability and the generalizability of predictions by spotlighting more the issues of data imbalance and classifier tuning that have been discussed in the literature related to these problems.

5.2. The Solution to the Real World Problems:

The present study aims at developing predictive models to attain early detection, which will, in turn, enable timely intervention and lead to reduced mortality.

- **Dataset Imbalance Issues:** The datasets used in healthcare are imbalanced. This may lead to biased predictions. In this scenario, the present work focuses on methodologies to handle class imbalance so that correct predictions could be attained between diseased and healthy subjects.
- **Real-Time Decision Supporting:** The developed models can be integrated into health care to support real-time decision making by the physician community to enhance the efficiency and effectiveness of patient care.
- **High Prediction Accuracy:** By using feature selection methods and a tuned model, the study tends to enhance the accuracy of predictions to reduce the errors occurring in conventional diagnosis.

5.3. Future Works and Horizons

Nevertheless, the suggested system for cardiovascular disease prediction has great potential and it can still be improved in various ways. One of the major avenues for his work is adding more clinical and lifestyle-related factors such as eating habits, exercising, family history of diseases, stress levels, and different types of cholesterol. The model capturing of health patterns through complexity is what this issue is about and thus more accurate risk assessments will be the result of the model's development.

Apart from that, more powerful machine learning and lesser learning like Gradient Boosting, Support Vector Machines, Artificial Neural Networks, and ensemble-based hybrid models could be applied to not only lifting predictive performance but also making the model robust. In addition to this, hyperparameter optimization and feature selection methods may be applied to enhance the model productivity even more and address overfitting.

The system can provide real-time cardiovascular risk prediction by integrating patient-generated data from wearable devices or electronic health records and thus it can be extended to that. It can then categorize patients into different risk levels such as low, moderate, and high which would remarkably add to the clinical value of the system and its role in decision-making.

In the end, the entire system's effectiveness can be increased through the creation of a web or mobile-based dashboard that enables professionals as well as the patients to see predictions, risk variations and health insights in a simple way. This extension would lead to greater availability, higher user participation and it would also contribute to the prevention of health care issues.

References

Bhanu Prakash Doppala, D. B. (2021, April 16). Cardiovascular_Disease_Dataset.
Retrieved from Mendeley Data: https://data.mendeley.com/datasets/dzz48mvjht/1
Ch Anwar ul Hassan, J. I. (2022). Effectively Predicting the Presence of Coronary Heart Disease Using Machine Learning Classifiers.

Jaime Lynn Speiser, M. E. (2019, November 15). A comparison of random forest variable selection methods for classification prediction modeling. Expert Systems with Applications, 93-101.

Jarett D. Berry, M. A.-J. (2012, January 26). Lifetime Risks of Cardiovascular Disease. Lifetime Risks of Cardiovascular Disease.

Ogunpola, A. (2024, January 8). Machine Learning-Based Predictive Models for Detection of Cardiovascular Diseases. Machine Learning-Based Predictive Models for Detection of Cardiovascular Diseases.

Saidi, A. (2021). Logistic Regression. FPGA-based implementation of classification techniques: A survey.

SONG, Y.-y. (2015, April 25). Decision tree methods: applications for classification and prediction. Decision tree methods: applications for classification and prediction.

WHO. (2021, June 11). World Health Organization. Retrieved from Cardiovascular diseases (CVDs): https://www.who.int/news-room/fact- sheets/detail/cardiovascular-diseases-(cvds)