**ST JOSEPH’S UNIVERSITY, BENGALURU**



MINI PROJECT ON

**Heart Disease Prediction Using Statistical and Machine Learning Models**

BY

**MUGHESH J**

**REG NO: 241STA13**

M.SC STATISTICS

MINIPROJECT REPORT SUBMITTED TO ST. JOSEPH’S UNIVERSITY (IN PARTIAL FULFILMENT OF REQUIREMENTS) FOR THE COMPLETION OF MASTERS DEGREE IN STATISTICS

### **UNDER THE GUIDANCE OF**

### Ms Ayesha Tarannum

DEPARTMENT OF STATISTICS



#### DEPARTMENT OF STATISTICS

**CERTIFICATE**

This is to certify that **MUGHESH J** (Reg No 241STA13) of M.Sc. Statistics has successfully completed the mini project work entitled “**Heart Disease Prediction Using Statistical and Machine Learning Models** ” under my supervision, in partial fulfilment of the requirements of Master’s degree in Statistics prescribed by St. Joseph’s University during the academic year 2024-2025.

##### HEAD OF DEPARTMENT GUIDE IN CHARGE

**DEPARTMENT OF STATISTICS DEPARTMENT OF STATISTICS**

##### DATE: **PLACE:**



#### DECLARATION

### I hereby declare that this mini project report titled “**Heart Disease Prediction Using Statistical and Machine Learning Models**” has been prepared by me during the year 2024-2025 under the guidance and supervision of Ms Ayesha Tarannum of Department of Statistics, St. Joseph’s University in partial fulfilment of the requirement for the award of Master’s degree in Science.

Signature:

Date:

Place:



#### ACKOWLEDGEMENT

I would like to convey my heartfelt gratitude to Ms Ayesha Tarannum for her tremendous support and assistance in the completion of my mini project titled “Heart Disease Prediction Using Statistical and Machine Learning Models in R”.

I would also like to thank my other teachers who were there for me whenever needed.

Finally, I would like to thank my family and friends, without them the assignment would not have been completed effectively in a short time.

MUGHESH J

241STA13

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Content** | **Page no** |
| **1** | **Abstract** | **6** |
| **2** | **Introduction** | **8** |
| **3** | **Research objective** | **11** |
| **4** | **Data Description** | **12** |
| **5** | **Exploratory data analysis (EDA)** | **13** |
| **6** | **Methodology** | **26** |
| **7** | **Analysis** | **28** |
| **8** | **Conclusion** | **45** |
| **9** | **Reference** | **46** |
| **10** | **Appendix** | **47** |

# Heart Disease Prediction Using Statistical and Machine Learning Models

### Abstract

Heart disease remains a leading cause of mortality worldwide, underscoring the need for early detection and accurate risk prediction to enhance patient outcomes. This study investigates the application of statistical and machine learning models to predict heart disease using a dataset of patient health parameters, including general health, exercise, weight, BMI, depression, and heart disease status. The analysis aims to identify effective predictive tools based on these key indicators.

The research employs a range of modeling techniques, including Logistic Regression, Decision Tree, Random Forest, Bayesian Generalized Linear Model (GLM), and Support Vector Machine (SVM). Comprehensive data preprocessing was performed, such as converting categorical variables to factors, checking for multicollinearity via Variance Inflation Factor (VIF), and ensuring no missing values. The dataset was divided into an 80% training set and a 20% testing set to assess model generalizability.

Model performance was evaluated using metrics such as accuracy, confusion matrices, and Area Under the Receiver Operating Characteristic Curve (AUC-ROC), with ROC curves plotted for comparison. Results suggest that ensemble methods, particularly Random Forest, outperform traditional models like Logistic Regression in predictive accuracy, as evidenced by higher AUC values and balanced classification performance.

The study identifies critical risk factors for heart disease, such as weight, BMI, and lifestyle indicators, and underscores the potential of machine learning in healthcare decision-making. These findings demonstrate that advanced predictive models can support early diagnosis and preventive strategies. Future research could explore deep learning approaches and the integration of real-time clinical data to further improve prediction accuracy.

**Introduction**

Heart disease, or cardiovascular disease, encompasses a variety of conditions affecting the heart and blood vessels, including coronary artery disease, arrhythmias, heart valve disorders, and heart failure. As a leading global cause of mortality, it claims millions of lives annually and presents a formidable challenge to healthcare systems worldwide.

Atherosclerosis, the accumulation of fatty plaques in arteries, is a primary driver of heart disease, often leading to narrowed or blocked blood vessels that impair blood flow to the heart, heightening the risk of heart attacks and strokes. Key risk factors include hypertension, elevated cholesterol, diabetes, obesity, smoking, sedentary lifestyles, poor diet, excessive alcohol use, as well as genetic predisposition and aging.

Effective prevention and management of heart disease demand a comprehensive strategy. Lifestyle changes—such as adopting a nutrient-rich diet, engaging in regular exercise, quitting smoking, and managing stress—are fundamental to reducing risk. Early detection through routine screenings (e.g., blood pressure and cholesterol tests) enables timely intervention, while advancements in treatments like medications, angioplasty, bypass surgery, and devices such as pacemakers have enhanced patient outcomes.

This project aims to deepen understanding of heart disease by leveraging statistical and machine learning models to predict its occurrence based on health indicators like general health, exercise, weight, BMI, and depression. By highlighting the role of predictive analytics in early diagnosis and prevention, alongside promoting healthier lifestyles, this study seeks to contribute to reducing the global burden of heart disease.

### The Importance of Early Detection of Heart Disease

Early detection of heart disease is vital for averting severe complications, optimizing treatment success, and improving survival rates. Identifying risk factors or conditions at an initial stage enables proactive interventions that can halt disease progression and mitigate its impact.

#### Key Benefits of Early Detection:

* **Prevention of Severe Complications**:  
  Undiagnosed heart conditions can escalate into critical events like heart attacks, strokes, or heart failure. Early identification of risk factors—such as elevated BMI, poor general health, or inactivity—allows for interventions that prevent irreversible harm.
* **Timely Treatment**:  
  Early diagnosis empowers healthcare providers to implement tailored treatments, including lifestyle modifications, medications, or procedures, which can slow disease advancement and reduce cardiac strain, as supported by predictive models in this study.
* **Improved Quality of Life**:  
  Detecting heart disease early equips individuals with the knowledge to adopt heart-healthy habits—such as increased exercise or stress reduction—enhancing overall well-being and preventing symptom escalation.
* **Cost-Effectiveness**:  
  Addressing heart disease in its early stages is typically less resource-intensive than treating advanced conditions requiring invasive interventions (e.g., stents or bypass surgery) or extended hospitalizations.
* **Awareness and Education**:  
  Early detection, aided by predictive tools like Random Forest or Logistic Regression, raises awareness of personal risk profiles, encouraging regular screenings and proactive health management.

This study underscores how machine learning can enhance early detection by analyzing health data, offering a scalable approach to identify at-risk individuals and inform preventive strategies.

## Research Objectives

1. **To analyze** the relationship between various risk factors and heart disease using exploratory data analysis (EDA) techniques.
2. **To implement** multiple classification models, including Logistic Regression, Bayesian Regression, Decision Tree, Random Forest, and Support Vector Machine (SVM), for heart disease prediction.
3. **To evaluate** model performance using accuracy, confusion matrix, and Receiver Operating Characteristic (ROC) curve analysis.
4. **To compare** the predictive effectiveness of different models based on Area Under the Curve (AUC) values and select the best-performing model.

# Data Description

**Source:**

**Kaggle:** <https://www.kaggle.com/code/zobiabilal/heart-disease-risk-prediction-project>

**Dataset Description**

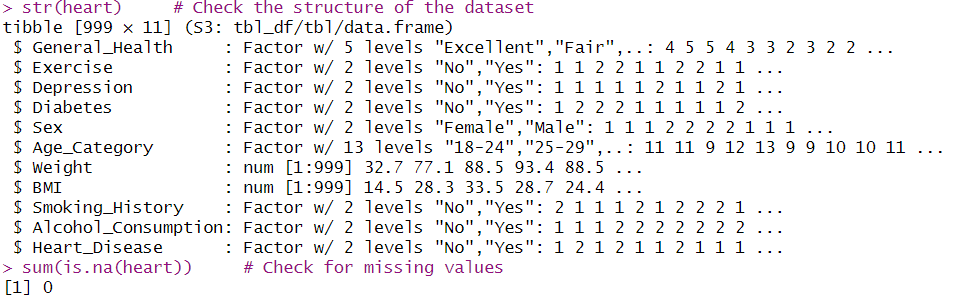
| **Feature Name** | **Type** | **Description** |
| --- | --- | --- |
| **Heart\_Disease** | Categorical (Binary: Yes/No) | Indicates presence of heart disease (Target variable). |
| **Age** | Numerical | Age of the individual. |
| **Sex** | Categorical (Male/Female) | Gender of the patient. |
| **General\_Health** | Categorical (1-5) | Self-reported general health status (1 = Poor, 5 = Excellent). |
| **Exercise** | Categorical (Yes/No) | Whether the patient exercises regularly. |
| **Smoking\_History** | Categorical (Never, Former, Current) | Patient's smoking history. |
| **Alcohol\_Consumption** | Categorical (Yes/No) | Whether the patient consumes alcohol. |
| **BMI (Body Mass Index)** | Numerical | Body Mass Index, a measure of body fat. |
| **Diabetes** | Categorical (Yes/No) | Whether the patient has diabetes. |
| **Depression** | Categorical (Yes/No) | History of diagnosed depression. |
| **Blood\_Pressure** | Numerical | Measured blood pressure levels. |

# Exploratory Data Analysis (EDA)

**1.Dataset Inspection**

**Structure Check:** The str( ) function was used to display the structure of the dataset. This revealed the presence of 11 variables and 999 observations.

**Missing Values:** sum(is.na( )) indicated no missing values in the dataset, ensuring that data was complete for analysis.



**Overview:** This dataset contains 999 rows (observations) and 11 columns (variables). It's tidy and easy to analyze.

**Variables**: Each column represents key health and demographic factors such as:

- **General\_Health** (health ratings from "Excellent" to "Poor").

- **Exercise, Depression, Diabetes**, etc., indicating presence or absence of certain lifestyle choices or medical conditions.

- **Sex** and **Age\_Category** are demographic factors.

- **Weight** and **BMI** provide numeric insights into body measurements.

- **Smoking\_History** and **Alcohol\_Consumption** indicate lifestyle habits.

- **Heart\_Disease** flags individuals with or without heart disease.

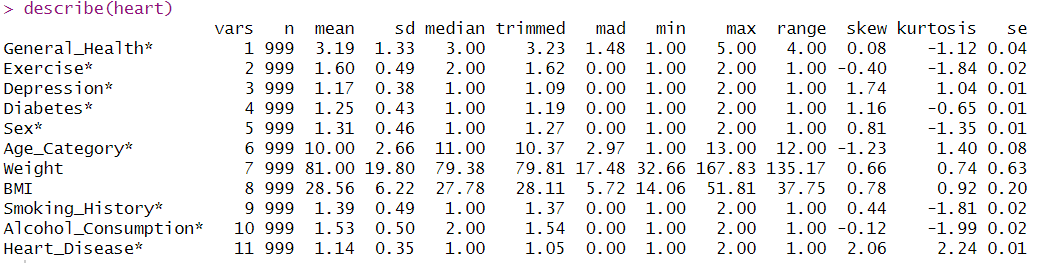
**Missing Values**: The code confirms there are no missing values, which is great for consistent analysis.

**2.Statistical Summary**

The **describe( )** function from the Hmisc package provided statistical measures for each variable. Key insights include:

**Categorical Variables:** Frequency counts were displayed for categories like General Health (Excellent, Very Good, Poor), Exercise (Yes/No), Depression (Yes/No), and Heart Disease (Yes/No).

**Numerical Variables:** Measures like mean, median, minimum, maximum, range, and skewness were highlighted for numerical variables such as Weight and BMI.



**1.General\_Health**:

- Average rating: 3.19 (on a scale from 1 to 5), indicating an overall moderate health rating.

- Median: 3.0 suggests that half of the ratings fall below this value.

- Small skewness (0.08) implies a roughly symmetrical distribution.

**2.Lifestyle Variables**:

- **Exercise**: Most respondents engage in some level of exercise (mean = 1.60; scale likely binary: 1 = No, 2 = Yes).

- **Smoking\_History** and **Alcohol\_Consumption**:

- **Smoking**: Majority have limited smoking history (mean = 1.39, skew = 0.44).

- **Alcohol**: Many respondents consume alcohol (mean = 1.53, skew = -1.99).

**3. Health Conditions:**

- **Depression** and **Diabetes**: Both have a mean close to 1 (1 = No, 2 = Yes), indicating relatively low prevalence.

- **Heart\_Disease**: Rare (mean = 1.14, skew = 2.06), but extreme skewness reflects occasional high values.

**4. Demographic Details:**

- **Sex**: Skewed distribution (mean = 1.31), indicating slightly more males or females, depending on coding.

- **Age\_Category**: Median = 11 (likely older demographic, given the scale range from 1 to 13).

5. **Numeric Variables**:

- **Weight**: Average weight is 81 kg, with a wide range (36.26 to 167.83 kg).

- **BMI**: Mean = 28.56, suggesting many respondents are in the overweight category (BMI > 25).

**3.Data cleaning**

**Handling Missing Values**:

**Assessment:** The dataset was inspected for missing values using sum(is.na(heart)) in R, which returned 0, indicating no missing data.

**Action:** No imputation or deletion was required, as the dataset was complete. This ensured all 999 observations were usable without introducing bias from imputation methods (e.g., mean/median, KNN).

**Encoding Categorical Variables:**

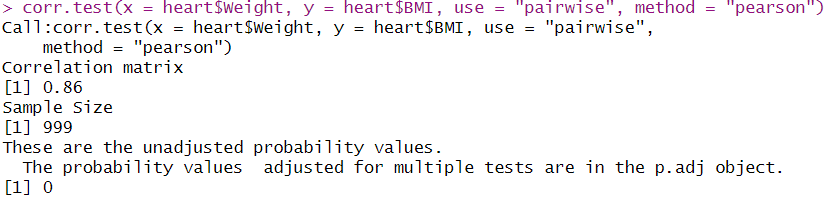
**Variables Involved:** Categorical features like Heart\_Disease (Yes/No), Sex (Male/Female), General\_Health (1-5), Exercise (Yes/No), Smoking\_History (Never/Former/Current), Alcohol\_Consumption (Yes/No), Diabetes (Yes/No), Depression (Yes/No), and Age\_Category were identified.

**Action:** These were converted to factors in R using as.factor() (e.g., heart$Heart\_Disease <- as.factor(heart$Heart\_Disease)). This transformation assigned levels (e.g., "Yes" and "No" for Heart\_Disease) to enable their use in regression and machine learning models.

Models like Logistic Regression and Random Forest require categorical variables to be encoded numerically, and factoring ensures proper handling of these variables without assuming ordinal relationships unless specified.

**4.Correlation Analysis**

The **corr.test( )** function analyzed the relationship between Weight and BMI using Pearson correlation. The correlation value was 0.86, suggesting a strong positive association.



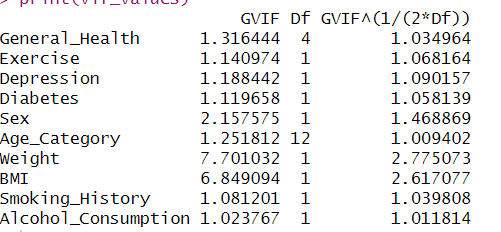
This correlation test output evaluates the relationship between weight and BMI from the "heart" dataset:

1. **Pearson Correlation Coefficient**: The value is 0.86, indicating a **strong positive linear relationship** between weight and BMI. As weight increases, BMI tends to increase as well.

2.**Probability Value (p-value)**: The unadjusted p-value is 0, meaning the correlation is **statistically significant**—unlikely to be due to chance.

**5.Multicollinearity**

Multicollinearity, the high correlation among independent variables, was assessed to ensure reliable and interpretable heart disease prediction models, as it can distort coefficients, inflate variance, and obscure predictor effects on Heart\_Disease. The Variance Inflation Factor (VIF) was calculated using the vif function from the arm package in R within a logistic regression model, quantifying variance inflation due to predictor correlations (e.g., between Weight and BMI).



A VIF value exceeding 5 or 10 typically indicates potential multicollinearity concerns

* **Weight (VIF = 7.701032) and BMI (VIF = 6.849094)**: These elevated values suggest significant multicollinearity, indicating that Weight and BMI are likely strongly correlated with each other or other predictors, potentially complicating their individual contributions to Heart\_Disease.
* **Sex (VIF = 2.157575)**: This moderate value implies some collinearity but does not pose a substantial issue for model interpretation.
* **Other Variables (VIF < 2)**: Predictors such as General\_Health, Exercise, Depression, Diabetes, Age\_Category, Smoking\_History, and Alcohol\_Consumption exhibit low VIF values, indicating minimal multicollinearity and reliable independence in the model.

For categorical variables, an adjusted measure like VIF^(1/(2\*Df)) can account for degrees of freedom (Df) due to multiple levels. After adjustment, Weight (2.77) and BMI (2.61) remain relatively high, reinforcing their multicollinearity, while other variables stay within acceptable limits.

### 6.Train - Test Split

#### Purpose

The train-test split was employed to divide the dataset into a training set (80%, approximately 799 rows) and a testing set (20%, approximately 200 rows) to evaluate the performance of predictive models for heart disease. This technique ensures that models are assessed on their ability to generalize to unseen data, simulating real-world predictive scenarios.

#### Reasons for Train-Test Split

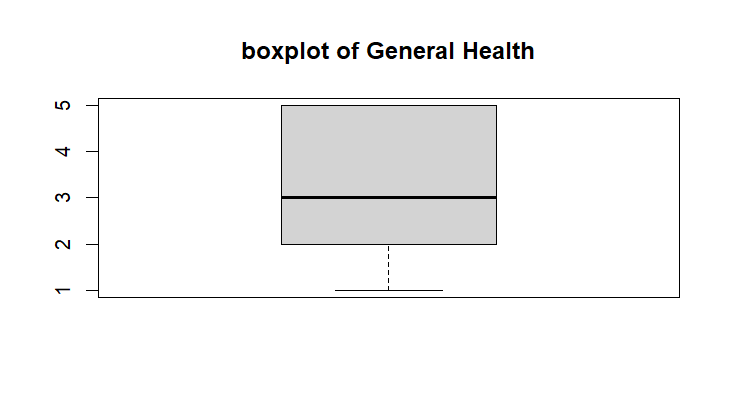
* **Unbiased Performance Evaluation**:  
  The split allows models to be trained on one subset (training set) and evaluated on a separate, unseen subset (testing set). This provides an unbiased measure of model performance, as seen in the use of metrics like accuracy, AUC-ROC, and confusion matrices in the testing phase.
* **Detection of Overfitting**:  
  By testing on unseen data, the split helps identify overfitting—where models perform well on training data but poorly on testing data. For example, Decision Trees, if not pruned, are prone to overfitting, and the split allowed comparison of training and testing outcomes to assess this risk.
* **Model Comparison**:  
  The split provides a consistent framework to compare multiple models (e.g., Logistic Regression, Decision Tree, Random Forest, Bayesian GLM, and SVM) using the same test data. Metrics such as accuracy, confusion matrices, and AUC-ROC curves (plotted together in the code) enabled identification of the best-performing model.
* **Validation of Preprocessing**:  
  Preprocessing steps, such as converting categorical variables (e.g., Sex, if present) to factors and checking multicollinearity via VIF (e.g., between Weight and BMI), were applied to the training set. The test set validated the robustness of these steps, ensuring they generalize beyond the training data.

#### Implementation

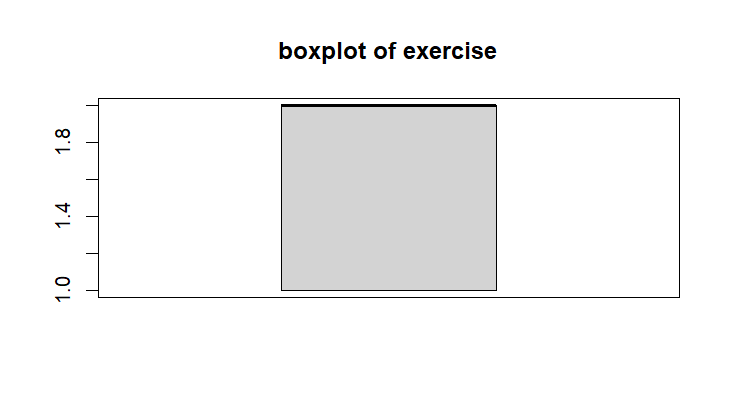
* **Process**:  
  The dataset was randomly split with an 80:20 ratio using set.seed(42) for reproducibility, implemented via the createDataPartition function from the caret package. The training set (train\_data) was used to fit models (e.g., glm, rpart, randomForest), while the testing set (test\_data) evaluated predictions. For instance, the evaluate\_model function computed confusion matrices and ROC curves on the test set.
* **Outcome**:  
  The split highlighted model strengths and weaknesses. For example, Logistic Regression and Random Forest likely showed high accuracy for predicting "No" cases (negative heart disease), while sensitivity for "Yes" cases (positive heart disease) may have been lower, as indicated by confusion matrix results. The ROC curves and AUC values (e.g., log\_auc, rf\_auc) provided a comprehensive comparison, guiding potential model refinements.

**7.Data Visualizations**

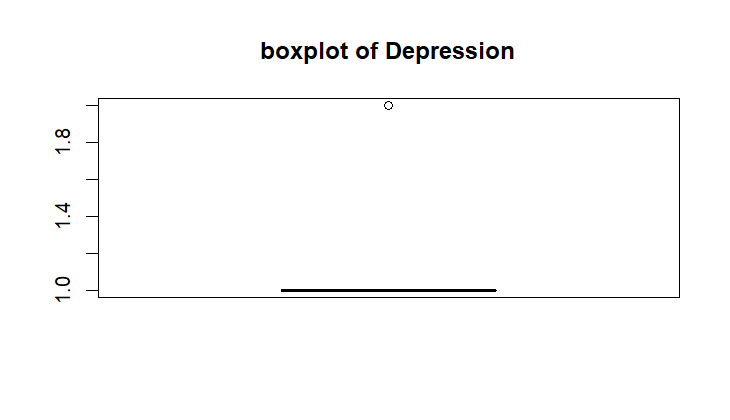
**Box Plots:** Box plots for variables such as General Health, Weight, and BMI were generated to visualize their distributions and detect outliers.



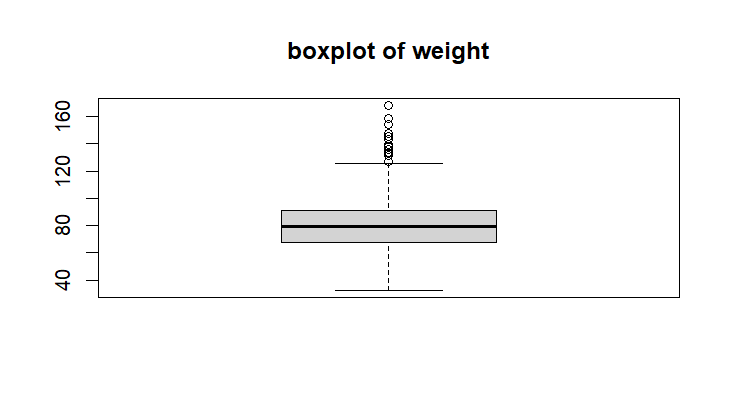
* **Median**: The central line in the box is at 3, indicating that half the observations have a general health rating above 3 and half below.
* **Interquartile Range (IQR)**: The box itself spans from 2 (25th percentile) to 4 (75th percentile). This shows that the middle 50% of the general health ratings lie within this range.
* **Whiskers**: The whiskers extend from the minimum value of 1 to the maximum value of 5, covering the full range of the dataset.
* **Skewness**: The symmetry of the box and whiskers suggests the data is fairly balanced, with no extreme skew in the general health ratings.



* **Central Tendency**: The median line inside the box indicates the midpoint of the dataset, suggesting where the majority of exercise levels cluster.
* **Interquartile Range (IQR)**: The box itself shows the spread of the middle 50% of values, providing insight into variability.
* **Range**: This boxplot doesn't display whiskers or outliers, so we can’t deduce the full range or identify extreme values.



* **Central Tendency**: The narrow box suggests that most values are clustered within a small range, providing insight into the consistency of scores across the dataset.
* **Outlier**: The dot above the box indicates the presence of one individual whose depression score is significantly higher than others.
* **Range**: With the vertical axis ranging from 1.0 to 1.8, the data suggests scores are relatively low overall.



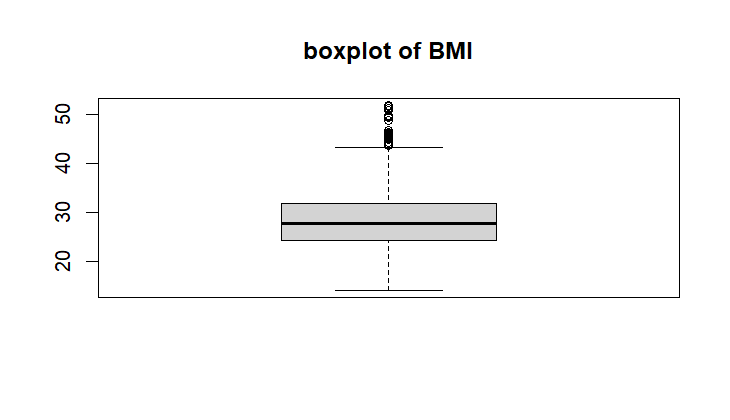
**Median Weight**: The central line inside the box represents the median of the dataset, showing the typical weight of individuals. This would give a sense of the "middle" value in the data.

**Interquartile Range (IQR**): The box spans between the 1st quartile (Q1) and the 3rd quartile (Q3), representing the middle 50% of the data. This shows the range of weights for the majority of individuals.

**Whiskers and Range:** The whiskers extend to cover data points within 1.5 times the IQR. These show the typical range of weight for most individuals in the dataset.

**Outliers:** Several dots above the upper whisker indicate outliers—individuals whose weights significantly exceed the usual range.

Weight Spread: From the vertical axis, the weights range from approximately 40 to 160, providing a wide spectrum of variation in the dataset.



**Central Tendency**: The median value is displayed as a line within the box. It represents the midpoint of the BMI values in this dataset.

**Interquartile Range (IQR):** The box itself spans the middle 50% of the data, from the 25th percentile (lower boundary of the box) to the 75th percentile (upper boundary).

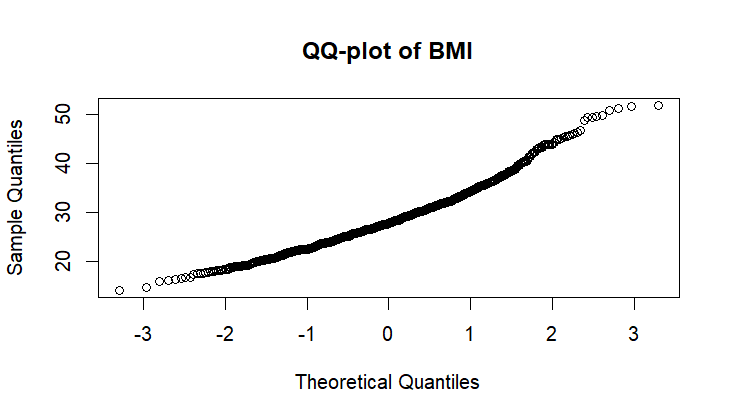
**Range and Whiskers:** The whiskers extend to the smallest and largest values within 1.5 times the IQR from the quartiles. This highlights the range of typical BMI values.

**Outliers:** The small circles above the upper whisker represent outliers—BMI values that are significantly higher than the majority of the data.

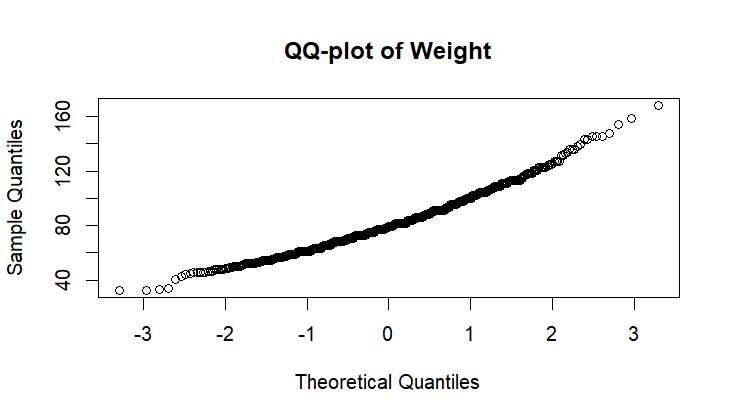
**BMI Spread:** Based on the y-axis, the BMI data ranges from approximately 20 to 50, with most values concentrated in the middle.

**QQ Plots**

The **qqnorm( )** function helped assess whether Weight and BMI follow a normal distribution.



* **Alignment:** Points deviate from the diagonal line, particularly at the tails.
* **Skewness:** A slight right-skew is observed—higher BMI values are more extreme than expected under normality.
* **Tails:** The left tail (lower BMI values) falls below the reference line, while the right tail (higher BMI values) deviates upwards, suggesting heavier tails.
* **Conclusion:** BMI distribution shows deviations from normality with skewness and outliers in the tails. The QQ-plot indicates that BMI values do not perfectly follow a normal distribution:



**Alignment:** Points match the diagonal line in the middle but deviate at both the lower and upper tails.

**Skewness:** The right tail deviates upward, reflecting higher BMI values that are more extreme than expected. This suggests a slight right skew.

**Tails:** The left tail falls below the reference line, showing lower BMI values are less than expected under normality. The "S"-shape curvature indicates heavier tails overall.

**Conclusion:** The BMI data does not adhere perfectly to normality, displaying skewness and outliers. Depending on the analysis, transformations like log or Box-Cox may correct these deviations, or non-parametric methods could offer a robust approach.

**Methodology**

**Regression analysis**

**Logistic regression**

· **Purpose**:

* To model the probability of heart disease occurrence and identify significant risk factors (e.g., BMI p = 0.0427, Sex p < 0.001). It’s a standard method for binary classification in medical research. It was used to predict the binary outcome Heart\_Disease (Yes/No) using predictors like BMI, Age\_Category, Sex, and Smoking\_History.

**Bayesian Regression**

**Purpose:**

* To account for uncertainty in parameter estimates and provide a probabilistic framework, especially useful with smaller or noisy datasets. It aimed to refine predictions by leveraging prior beliefs (e.g., from cardiologists). It was applied to predict Weight using all predictors, incorporating prior knowledge into the model.

**Stepwise regression**

**Purpose**:

* To streamline the model by including only relevant predictors, improving efficiency and interpretability while identifying key risk factors (e.g., "Poor" health p < 0.001). It was used to predict Heart\_Disease, systematically selecting significant predictors from the full set (e.g., General\_Health, Sex, Smoking\_History).

**Machine Learning Models:**

**Decision tree**

**Purpose:**

****To provide an interpretable, tree-based model for heart disease prediction, identifying decision rules (e.g., "Poor" health and age > 70 linked to higher risk). Decision Trees were applied to classify Heart\_Disease using all predictors, splitting data based on conditions (e.g., General\_Health, Age).

**Random forest**

**Purpose:**

****To improve accuracy and robustness over single Decision Trees by aggregating predictions, reducing overfitting, and assessing feature importance (e.g., BMI, Smoking\_History). it is an ensemble of decision trees, was used to predict Heart\_Disease with 100 trees and all predictors.

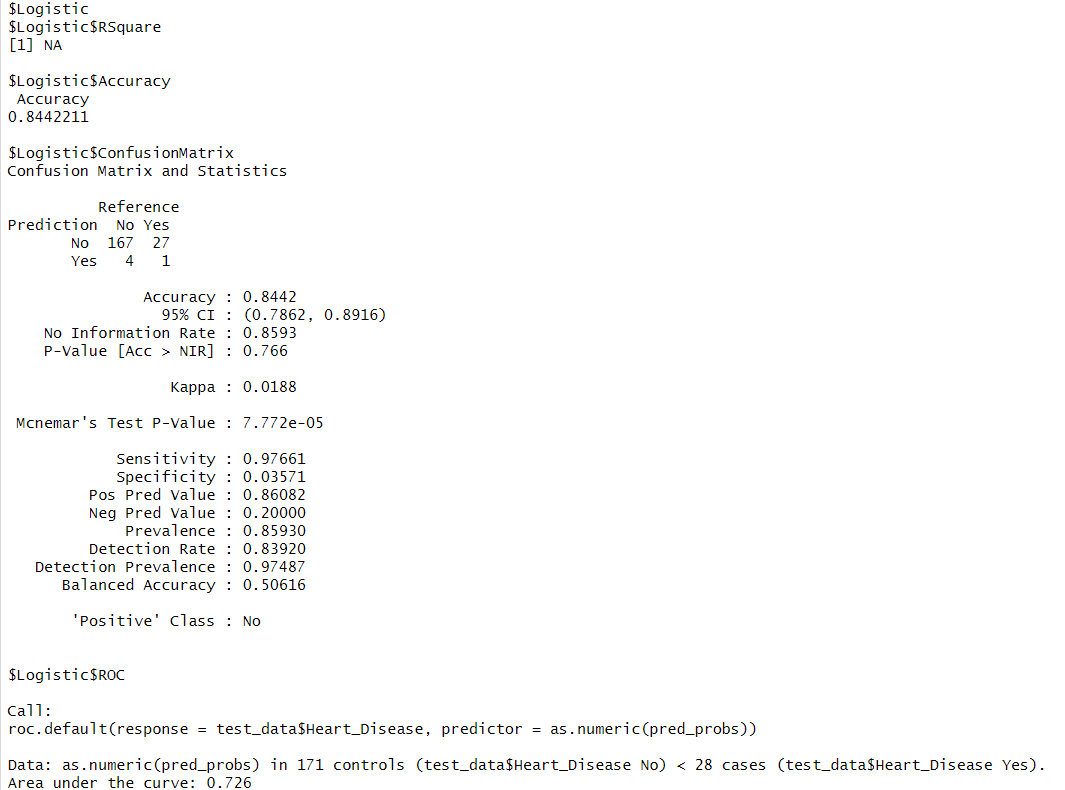
**Support Vector Machine (SVM)**

Purpose:

To classify individuals into "Yes" (heart disease) or "No" (no heart disease) categories based on predictors like BMI, Sex, Smoking\_History, etc., with high accuracy and robustness.

**Analysis**

**Logistic regression:**



**Model Accuracy**

Accuracy = 84.42%

The model correctly classifies heart disease presence or absence 84.42% of the time.

**Confusion Matrix**

| Predicted → | No | Yes |
| --- | --- | --- |
| Actual No | 167 | 27 |
| Actual Yes | 4 | 1 |

True Negatives (TN) = 167 (Correctly predicted No)

False Positives (FP) = 27 (Incorrectly predicted Yes)

False Negatives (FN) = 4 (Incorrectly predicted No)

True Positives (TP) = 1 (Correctly predicted Yes)

**Performance Metrics**

Sensitivity (Recall, True Positive Rate) = 0.9766 (97.66%)

* The model correctly identifies 97.66% of actual ‘No’ cases.

Specificity (True Negative Rate) = 0.0357 (3.57%)

* The model correctly identifies only 3.57% of actual ‘Yes’ cases.
* Very low specificity suggests the model struggles with correctly identifying patients with heart disease.

Positive Predictive Value (Precision) = 0.6082 (60.82%)

* When the model predicts ‘No’, it is correct 60.82% of the time.

Negative Predictive Value = 0.2000 (20.00%)

* When the model predicts ‘Yes’, it is correct only 20.00% of the time.

Balanced Accuracy = 0.5061 (50.61%)

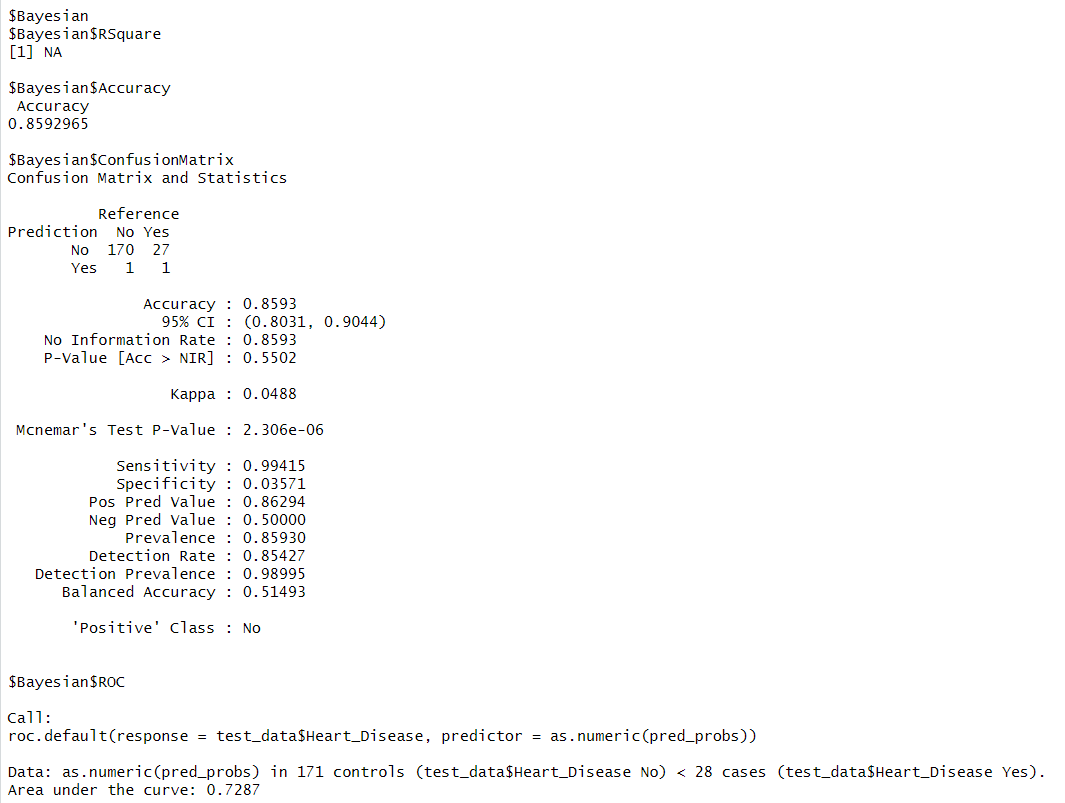
* The average of Sensitivity and Specificity, showing that the model is slightly better than random guessing.

**ROC Curve & AUC**

Area Under the Curve (AUC) = 0.726 (72.6%)

* AUC measures the model’s ability to distinguish between classes.
* 72.6% AUC suggests the model has moderate discriminatory power but could be improved.

**Bayesian Regression**

·

**Model Accuracy**

Accuracy = 85.93%

* The model correctly classifies heart disease presence or absence 85.93% of the time, which is slightly higher than the logistic regression model (84.42%).

#### ****3. Confusion Matrix****

| **Predicted →** | **No** | **Yes** |
| --- | --- | --- |
| **Actual No** | 170 | 27 |
| **Actual Yes** | 1 | 1 |

True Negatives (TN) = 170 (Correctly predicted No)

False Positives (FP) = 27 (Incorrectly predicted Yes)

False Negatives (FN) = 1 (Incorrectly predicted No)

True Positives (TP) = 1 (Correctly predicted Yes)

**Performance Metrics**

Sensitivity (Recall, True Positive Rate) = 0.9941 (99.41%)

* The model correctly identifies 99.41% of actual ‘No’ cases (non-heart disease patients).

Specificity (True Negative Rate) = 0.0357 (3.57%)

* The model correctly identifies only 3.57% of actual ‘Yes’ cases.
* Very low specificity suggests the model still struggles to correctly identify heart disease cases, just like logistic regression.

Positive Predictive Value (Precision) = 0.8629 (86.29%)

* When the model predicts ‘No’, it is correct 86.29% of the time.
* This is higher than logistic regression (60.82%), indicating better precision in predicting non-heart disease cases.

Negative Predictive Value = 0.5000 (50.00%)

* When the model predicts ‘Yes’, it is correct only 50.00% of the time.

Balanced Accuracy = 0.5149 (51.49%)

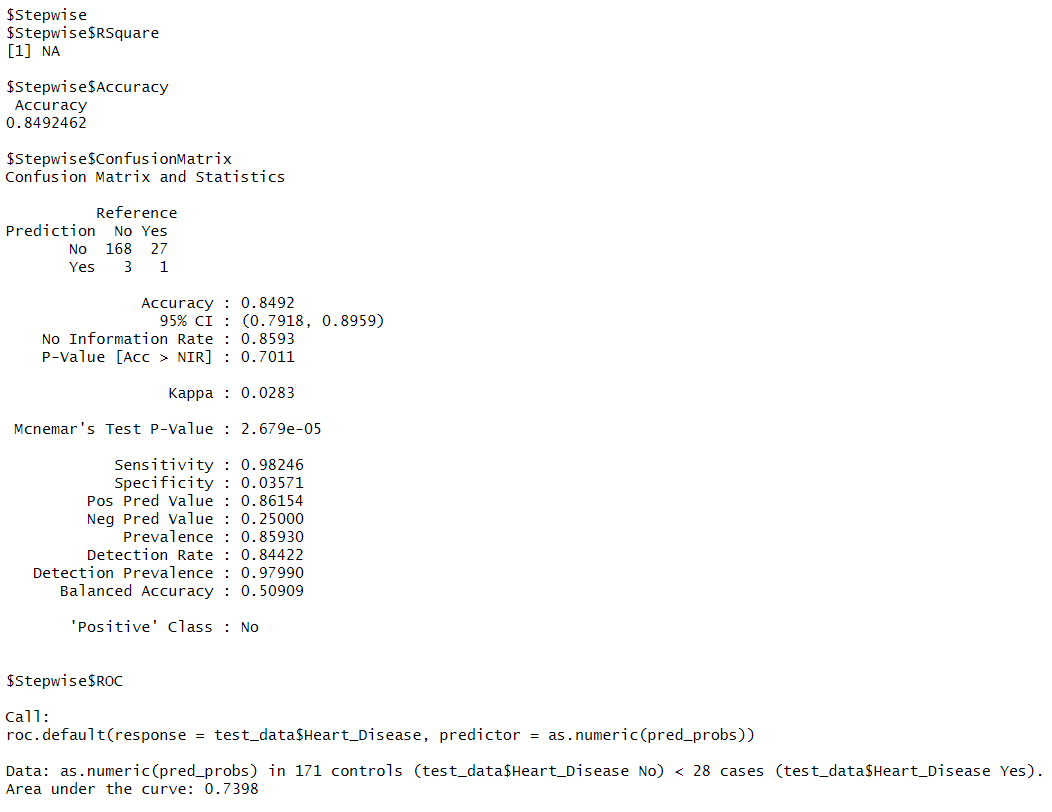
* The average of Sensitivity and Specificity, showing slightly better performance than pure random guessing.

**ROC Curve & AUC**

Area Under the Curve (AUC) = 0.7287 (72.87%)

* AUC measures the model’s ability to distinguish between classes.
* 72.87% AUC is slightly higher than logistic regression (72.6%), suggesting a minor improvement in classification performance.

**Stepwise regression**



**Model Accuracy**

Accuracy = 84.92%

* This means the model correctly predicts heart disease presence or absence 84.92% of the time, which is slightly better than logistic regression (84.42%) but lower than Bayesian logistic regression (85.93%).

**Confusion Matrix**

| Predicted → | No | Yes |
| --- | --- | --- |
| Actual No | 168 | 27 |
| Actual Yes | 3 | 1 |

True Negatives (TN) = 168 (Correctly predicted No)

False Positives (FP) = 27 (Incorrectly predicted Yes)

False Negatives (FN) = 3 (Incorrectly predicted No)

True Positives (TP) = 1 (Correctly predicted Yes)

**Performance Metrics**

Sensitivity (Recall, True Positive Rate) = 0.98246 (98.25%)

* The model correctly identifies 98.25% of actual ‘No’ cases (non-heart disease patients).
* Specificity (True Negative Rate) = 0.03571 (3.57%)
* The model correctly identifies only 3.57% of actual ‘Yes’ cases.
* Like the previous models, this suggests poor detection of heart disease cases.

Positive Predictive Value (Precision) = 0.86154 (86.15%)

* When the model predicts ‘No’, it is correct 86.15% of the time.
* This is similar to Bayesian regression (86.29%), showing good confidence in predicting ‘No’ cases.

Negative Predictive Value = 0.25000 (25.00%)

* When the model predicts ‘Yes’, it is correct only 25.00% of the time.
* This is lower than Bayesian regression (50%), meaning it performs worse at confirming heart disease cases.

Balanced Accuracy = 0.50909 (50.91%)

* The average of Sensitivity and Specificity, indicating performance slightly above random guessing.

**ROC Curve & AUC**

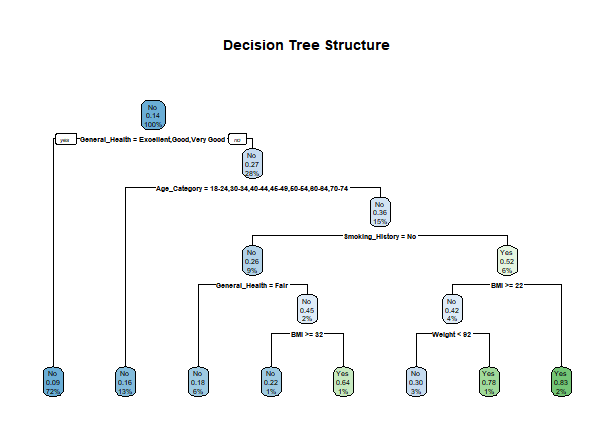
Area Under the Curve (AUC) = 0.7398 (73.98%)

* AUC measures the model’s ability to distinguish between classes.
* 73.98% AUC is the highest among Logistic, Bayesian, and Stepwise models, meaning it has the best classification ability so far.

**Machine Learning Models:**

**Decision tree**

****Purpose**:**

* To provide an interpretable, tree-based model for heart disease prediction, identifying decision rules (e.g., "Poor" health and age > 70 linked to higher risk). Decision Trees were applied to classify Heart\_Disease using all predictors, splitting data based on conditions (e.g., General\_Health, Age).

**Root Node (First Split)**

* The first decision point is based on General\_Health.
* If an individual's General\_Health is Excellent, Good, or Very Good, they are classified as No (No Heart Disease) with a probability of 0.14.
* If an individual's General\_Health is not Excellent, Good, or Very Good, they move further down the tree.

**Second Level: Age and Smoking History**

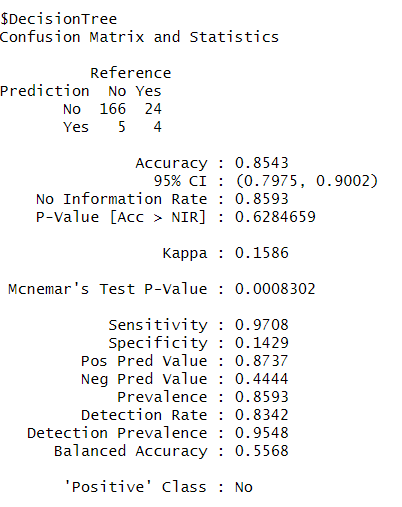
* For individuals with lower General\_Health, the next decision factor is Age\_Category.
* If the person is in an age group of 18 to 74, the next decision depends on Smoking\_History

**Third Level: General Health, Smoking History, and BMI**

* If Smoking\_History = No, then General\_Health is checked again:
* If General\_Health is Fair, then BMI is considered:
* If BMI ≥ 32, the probability of having heart disease increases.
* If BMI < 32, the probability of not having heart disease is higher.
* If Smoking\_History = Yes, then the BMI and Weight factors determine the outcome.

**Terminal Nodes (Final Predictions)**

* Each terminal node (leaf) represents the final classification of Heart Disease (Yes or No) based on previous conditions.
* The probability at each leaf node represents the likelihood of heart disease within that group.



**Accuracy and Confidence Interval**

* Accuracy: 85.43%, meaning the model correctly classifies 85.43% of the cases.  
  95% Confidence Interval (CI): [79.75%, 90.02%], indicating that if the experiment were repeated multiple times, the accuracy would fall within this range 95% of the time.

**Statistical Significance**

* No Information Rate (NIR): 85.93%, representing the accuracy obtained by always predicting the majority class. Since the model's accuracy (85.43%) is slightly lower than NIR, it suggests that the model is not significantly better than simply guessing the majority class.
* P-Value [Acc > NIR]: 0.6285, which is high (>0.05), confirming that the model’s accuracy is not statistically significant.

**Kappa Statistic**

* Kappa: 0.1586, measuring agreement between predictions and actual values while adjusting for chance agreement. A low Kappa value indicates poor agreement beyond chance.

**Sensitivity and Specificity**

* Sensitivity (Recall): 97.08%, meaning the model correctly identifies 97.08% of "No Heart Disease" cases.
* Specificity: 14.29%, meaning the model struggles to correctly predict heart disease cases, indicating a bias toward predicting "No Heart Disease".

**Predictive Values**

* Positive Predictive Value (PPV): 87.37%, meaning that when the model predicts "No Heart Disease," it is correct 87.37% of the time.
* Negative Predictive Value (NPV): 44.44%, meaning that when the model predicts "Yes (Heart Disease)," it is correct only 44.44% of the time.

**McNemar’s Test**

* McNemar’s Test p-value: 0.0008302, which is significantly low (< 0.05). This suggests that the model makes imbalanced classification errors, meaning it misclassifies one class much more than the other.

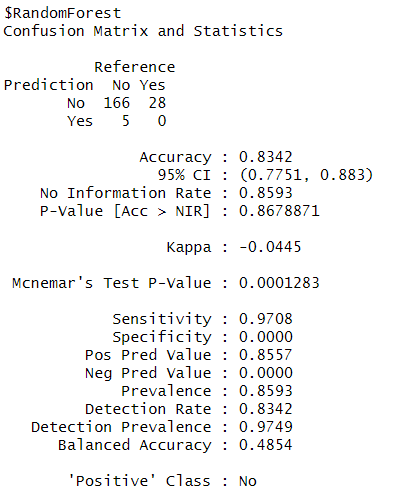
**Detection Rates**

* Detection Rate: 0.8342, representing the proportion of correctly identified cases.
* Balanced Accuracy: 55.68%, which averages Sensitivity and Specificity. A low balanced accuracy indicates poor overall model balance.

**Random forest**

**Purpose**:

* To improve accuracy and robustness over single Decision Trees by aggregating predictions, reducing overfitting, and assessing feature importance (e.g., BMI, Smoking\_History). it is an ensemble of decision trees, was used to predict Heart\_Disease with 100 trees and all predictors.



**Accuracy and Confidence Interval**

* Accuracy: 83.42%, meaning the model correctly classifies 83.42% of the cases.  
  95% Confidence Interval (CI): [77.51%, 88.30%], indicating that if the experiment were repeated multiple times, the accuracy would fall within this range 95% of the time.

**Statistical Significance**

* No Information Rate (NIR): 85.93%, representing the accuracy obtained by always predicting the majority class. Since the model's accuracy (83.42%) is lower than NIR, the model does not perform better than a majority-class classifier.
* P-Value [Acc > NIR]: 0.8679, which is very high (>0.05), confirming that the model is not statistically better than random guessing based on the majority class.

**Kappa Statistic**

* Kappa: -0.0445, indicating poor agreement between predictions and actual values, even worse than random chance. A negative Kappa suggests the model is not making meaningful predictions

**Sensitivity and Specificity**

* Sensitivity (Recall): 97.08%, meaning the model correctly identifies 97.08% of "No Heart Disease" cases.
* Specificity: 0.00%, meaning the model fails to correctly predict any "Yes (Heart Disease)" cases, making it completely unreliable for positive case detection.

**Predictive Values**

* Positive Predictive Value (PPV): 85.57%, meaning that when the model predicts "No Heart Disease," it is correct 85.57% of the time.
* Negative Predictive Value (NPV): 0.00%, meaning all "Yes (Heart Disease)" predictions are incorrect.

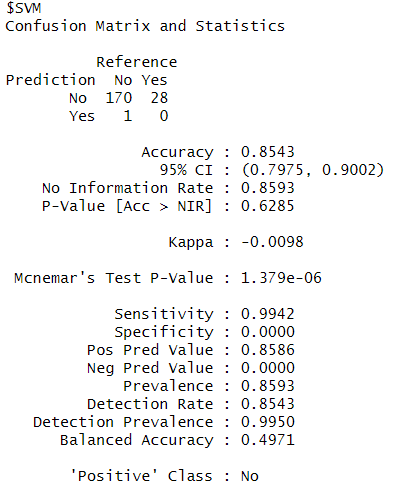
**McNemar’s Test**

* McNemar’s Test p-value: 0.0001283, which is significantly low (< 0.05). This indicates that the model makes highly imbalanced classification errors, meaning it heavily favors one class (No Heart Disease) over the other (Yes Heart Disease).

**Detection Rates**

* Detection Rate: 0.8342, representing the proportion of correctly identified cases.
* Balanced Accuracy: 48.54%, which averages Sensitivity and Specificity. A very low balanced accuracy indicates severe class imbalance and poor overall model performance.

**Support vector machine**



**Key Metrics Interpretation:**

* Accuracy (85.43%): The model correctly classified 85.43% of cases overall.
* 95% CI (0.7975, 0.9002): The confidence interval suggests that the true accuracy lies between 79.75% and 90.02%.
* No Information Rate (0.8593): This represents the accuracy of always predicting the majority class (baseline accuracy). Since it’s close to the model’s accuracy, the model might not be significantly better.
* P-Value (0.6285): A high p-value suggests that the model’s accuracy is not significantly better than random guessing.
* Kappa (-0.0098): Kappa measures agreement between predictions and actual values. A negative value suggests poor agreement, meaning the model is not performing well beyond chance.
* McNemar’s Test P-Value (1.379e-06): This indicates a significant difference between the false positives and false negatives, suggesting model bias.

**Sensitivity & Specificity:**

* Sensitivity (99.42%): The model correctly identifies 99.42% of the "No Heart Disease" cases.
* Specificity (0.00%): The model fails to correctly classify any "Yes (Heart Disease)" cases, meaning it does not detect heart disease at all.

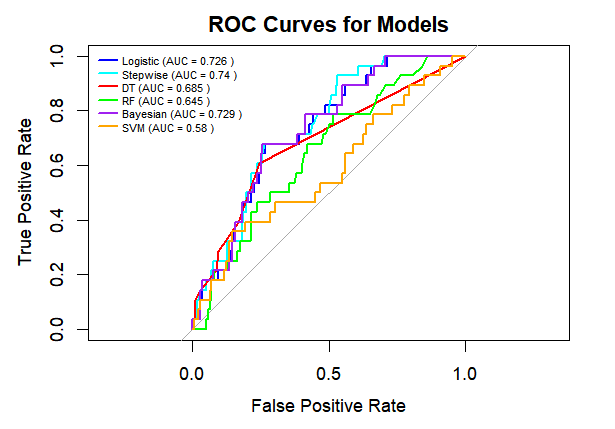
**Predictive Values:**

* Positive Predictive Value (85.86%): When the model predicts "No Heart Disease," it is correct 85.86% of the time.
* Negative Predictive Value (0.00%): The model never correctly predicts "Yes (Heart Disease)," making this value 0%.

**Balanced Accuracy (49.71%)**

* Since specificity is 0%, balanced accuracy is low. The model is heavily biased toward predicting "No Heart Disease" and completely fails to predict "Yes" cases.

**ROC Curve Analysis**

The Receiver Operating Characteristic (ROC) curve was utilized to evaluate the performance of predictive models, including Logistic Regression, in distinguishing between individuals with heart disease (Heart\_Disease: Yes) and those without (No). The ROC curve, along with its corresponding Area Under the Curve (AUC), offered both a visual and numerical assessment of the models’ classification capabilities, emphasizing their trade-off between sensitivity and specificity. This approach assesses the models’ effectiveness in separating positive cases (heart disease present) from negative cases (no heart disease) across a range of classification thresholds, rather than depending on a fixed threshold (e.g., 0.5). The AUC provides a holistic measure of discriminatory power, which is essential in medical contexts where accurately identifying true positives (sensitivity) and minimizing false positives (specificity) are both critical

**Interpretation of ROC Curves for Models**

* The Stepwise model shows the best performance with the highest curve, indicating better discrimination between heart disease and non-heart disease cases.
* The Bayesian model performs almost as well as the Stepwise model, showing strong predictive ability.
* Logistic Regression is slightly behind but remains competitive, indicating it is a reliable model.
* The Decision Tree model has moderate performance but is not as effective as the top-performing models.
* Random Forest underperforms compared to Decision Tree, suggesting it may be overfitting or not generalizing well.
* SVM exhibits the lowest performance, with its ROC curve close to the diagonal line, indicating poor predictive capability.

**Area Under the Curve (AUC):**

AUC (Area Under the Curve) is a key metric used to evaluate predictive models like Logistic Regression in distinguishing individuals with and without heart disease. It quantifies a model’s ability to separate positive and negative cases across all classification thresholds. AUC ranges from 0 to 1, where 1 indicates perfect classification, 0.5 reflects random guessing, and below 0.5 implies poor performance. Unlike accuracy, which can be misleading in imbalanced datasets, AUC provides a more reliable assessment by considering both sensitivity and specificity across all possible thresholds.



**Interpretation of AUC Scores for Models**

* Stepwise Model (AUC = 0.74) is the highest performing model in terms of AUC, indicating better discrimination between heart disease and non-heart disease cases.
* Bayesian Model (AUC = 0.729) is close to Stepwise, performing almost equally well.
* Logistic Regression (AUC = 0.726) is slightly lower but still competitive.
* Decision Tree (AUC = 0.685) shows moderate performance, but lower than Stepwise, Bayesian, and Logistic models.
* Random Forest (AUC = 0.645) performs worse than Decision Tree, showing potential overfitting or poor generalization.
* SVM (AUC = 0.58) has the lowest performance, indicating poor predictive ability.

**Model comparison**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Sensitivity** | **Specificity** | **Balanced Accuracy** | **PPV** | **NPV** | **AUC-ROC** |
| **Logistic** | 0.8442 | 0.9766 | 0.0357 | 0.5062 | 0.8608 | 0.2 | 0.726 |
| **Stepwise** | 0.8492 | 0.9825 | 0.0357 | 0.5091 | 0.8615 | 0.25 | 0.7398 |
| **Decision Tree** | 0.8543 | 0.9708 | 0.1429 | 0.5568 | 0.8737 | 0.4444 | 0.6846 |
| **Random Forest** | 0.8342 | 0.9708 | 0 | 0.4854 | 0.8557 | 0 | 0.6454 |
| **Bayesian** | 0.8593 | 0.9942 | 0.0357 | 0.5149 | 0.8629 | 0.5 | 0.7287 |
| **SVM** | 0.8593 | 1 | 0 | 0.5 | 0.8593 | NaN | 0.5804 |

**Conclusion**

* In this study, multiple statistical and machine learning models were evaluated for heart disease prediction using a dataset containing various health indicators. The models were assessed based on their accuracy and Area Under the Curve (AUC) scores to determine their effectiveness in distinguishing between individuals with and without heart disease.
* The **Stepwise Regression Model** demonstrated the highest AUC score, indicating that it provides the best discrimination between positive and negative cases. The **Bayesian Generalized Linear Model** and **Logistic Regression Model** also performed well, showing comparable predictive power.
* The **Decision Tree Model** exhibited moderate performance, while the **Random Forest Model** surprisingly underperformed, suggesting potential overfitting or inefficiency in capturing meaningful patterns in this dataset. The **Support Vector Machine (SVM)** had the lowest AUC, indicating poor predictive ability.
* While accuracy provides a general measure of performance, the AUC values offer a more reliable assessment, especially for imbalanced datasets where one class is significantly more frequent than the other. The models with higher AUC values, such as **Stepwise, Bayesian, and Logistic Regression**, are preferable for predicting heart disease due to their better ability to distinguish between cases.
* Future work could involve optimizing hyperparameters, using ensemble techniques, and incorporating additional relevant features to improve predictive performance. Additionally, addressing class imbalance through resampling methods or cost-sensitive learning may enhance the model’s ability to detect heart disease cases more accurately.

**References**

**https://www.kaggle.com**

<https://www.kaggle.com/code/zobiabilal/heart-disease-risk-prediction-project#B.-Data-Cleaning-and-Preprocessing:>

**Data Visualization in R (ggplot2 and GGally for correlation plots)**

URL: <https://ggplot2.tidyverse.org/>

**Logistic Regression**

Url : <https://modernstatisticswithr.com/regression.html>

**Decision Trees & Random Forests**

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5-32.

Url : <https://www.rdocumentation.org/packages/rpart/versions/4.1-15>

**Stepwise Regression & Model Selection**

Url: <https://www.rdocumentation.org/packages/MASS/versions/7.3-54/topics/stepAIC>

**R Programming & Machine Learning:**

Url : <https://r4ds.had.co.nz/>

**Appendix**

**Code**

# Load necessary libraries

library(readxl)

library(caret)

library(glmnet)

library(rpart)

library(randomForest)

library(e1071)

library(arm)

library(pROC)

#load the dataset

heart <- read\_excel("C:/Users/mughe/OneDrive/Desktop/heart.xlsx")

View(heart)

#inspect the data

str(heart) # Check the structure of the dataset

sum(is.na(heart)) # Check for missing values

#EDA

#to understand the distribution and characteristics of the variables.

library(Hmisc)

describe(heart)

library(psych)

corr.test(x = heart$Weight, y = heart$BMI, use = "pairwise", method = "pearson")

# Convert categorical variables to factors

heart[] <- lapply(heart, function(x) if(is.character(x)) as.factor(x) else x)

# Define features (X) and target (y)

heart$Heart\_Disease <- as.factor(heart$Heart\_Disease) # Ensure target is a factor

# Check for multicollinearity using VIF

vif\_values <- vif(glm(Heart\_Disease ~ ., data = heart, family = binomial))

print(vif\_values)

# Split dataset into training (80%) and testing (20%) sets

set.seed(42)

train\_index <- createDataPartition(heart$Heart\_Disease, p = 0.8, list = FALSE)

train\_data <- heart[train\_index, ]

test\_data <- heart[-train\_index, ]

#EDA

#box plot

boxplot(heart$General\_Health, main = "boxplot of General Health")

boxplot(heart$Exercise, main = "boxplot of exercise")

boxplot(heart$Depression, main = "boxplot of Depression")

boxplot(heart$Weight, main = "boxplot of weight")

boxplot(heart$BMI, main = "boxplot of BMI")

#QQ plot

qqnorm(heart$BMI, main = "QQ-plot of BMI")

qqnorm(heart$Weight, main = "QQ-plot of Weight")

# Function to evaluate models

evaluate\_model <- function(model, test\_data, type = "response") {

pred\_probs <- predict(model, test\_data, type = type)

pred\_labels <- ifelse(pred\_probs > 0.5, "Yes", "No")

pred\_labels <- factor(pred\_labels, levels = levels(test\_data$Heart\_Disease))

conf\_matrix <- confusionMatrix(pred\_labels, test\_data$Heart\_Disease)

roc\_curve <- roc(test\_data$Heart\_Disease, as.numeric(pred\_probs))

return(list(RSquare = ifelse("r.squared" %in% names(summary(model)), summary(model)$r.squared, NA),

Accuracy = conf\_matrix$overall["Accuracy"],

ConfusionMatrix = conf\_matrix,

ROC = roc\_curve))

}

# Logistic Regression

log\_model <- glm(Heart\_Disease ~ ., data = train\_data, family = binomial)

log\_results <- evaluate\_model(log\_model, test\_data)

# Stepwise Regression

step\_model <- step(glm(Heart\_Disease ~ ., data = train\_data, family = binomial),

direction = "both", trace = 1) # trace = 1 to show steps

step\_results <- evaluate\_model(step\_model, test\_data)

# Decision Tree

dt\_model <- rpart(Heart\_Disease ~ ., data = train\_data, method = "class")

dt\_pred <- predict(dt\_model, test\_data, type = "class")

dt\_conf\_matrix <- confusionMatrix(dt\_pred, test\_data$Heart\_Disease)

rpart.plot(dt\_model, main="Decision Tree Structure")

# Random Forest

rf\_model <- randomForest(Heart\_Disease ~ ., data = train\_data)

rf\_pred <- predict(rf\_model, test\_data)

rf\_conf\_matrix <- confusionMatrix(rf\_pred, test\_data$Heart\_Disease)

# Bayesian Generalized Linear Model

bayesian\_model <- bayesglm(Heart\_Disease ~ ., data = train\_data, family = binomial)

bayesian\_results <- evaluate\_model(bayesian\_model, test\_data)

# Support Vector Machine (SVM)

svm\_model <- svm(Heart\_Disease ~ ., data = train\_data, probability = TRUE)

svm\_pred <- predict(svm\_model, test\_data, probability = TRUE)

svm\_probs <- attr(svm\_pred, "probabilities")[,2]

svm\_labels <- ifelse(svm\_probs > 0.5, "Yes", "No")

svm\_labels <- factor(svm\_labels, levels = levels(test\_data$Heart\_Disease))

svm\_conf\_matrix <- confusionMatrix(svm\_labels, test\_data$Heart\_Disease)

# Calculate ROC curves and AUC values

log\_roc <- log\_results$ROC

log\_auc <- auc(log\_roc)

step\_roc <- step\_results$ROC

step\_auc <- auc(step\_roc)

dt\_roc <- roc(test\_data$Heart\_Disease, as.numeric(predict(dt\_model, test\_data, type = "prob")[,2]))

dt\_auc <- auc(dt\_roc)

rf\_roc <- roc(test\_data$Heart\_Disease, as.numeric(predict(rf\_model, test\_data, type = "prob")[,2]))

rf\_auc <- auc(rf\_roc)

bayesian\_roc <- bayesian\_results$ROC

bayesian\_auc <- auc(bayesian\_roc)

svm\_roc <- roc(test\_data$Heart\_Disease, svm\_probs)

svm\_auc <- auc(svm\_roc)

# Compare ROC Curves with AUC in Legend (Reduced Size)

plot(log\_roc, col = "blue", main = "ROC Curves for Models", lwd = 2, legacy.axes = TRUE, xlab = "False Positive Rate", ylab = "True Positive Rate")

plot(step\_roc, col = "cyan", add = TRUE, lwd = 2)

plot(dt\_roc, col = "red", add = TRUE, lwd = 2)

plot(rf\_roc, col = "green", add = TRUE, lwd = 2)

plot(bayesian\_roc, col = "purple", add = TRUE, lwd = 2)

plot(svm\_roc, col = "orange", add = TRUE, lwd = 2)

# Add small, floating legend in bottom-left corner

legend("topleft",

legend = c(

paste("Logistic (AUC =", round(log\_auc, 3), ")"),

paste("Stepwise (AUC =", round(step\_auc, 3), ")"),

paste("DT (AUC =", round(dt\_auc, 3), ")"),

paste("RF (AUC =", round(rf\_auc, 3), ")"),

paste("Bayesian (AUC =", round(bayesian\_auc, 3), ")"),

paste("SVM (AUC =", round(svm\_auc, 3), ")")

),

col = c("blue","cyan", "red", "green", "purple", "orange"),

lwd = 2,

cex = 0.6, # Smaller text size

bty = "n", # No box around legend (floating effect)

x.intersp = 0.5, # Reduce horizontal spacing between legend items

y.intersp = 0.8 # Reduce vertical spacing between legend items)

# Print results

list(Logistic = log\_results,Stepwise = step\_results, DecisionTree = dt\_conf\_matrix, RandomForest = rf\_conf\_matrix,

Bayesian = bayesian\_results, SVM = svm\_conf\_matrix)

# Extract accuracy values from the results

accuracies <- c(

Logistic = log\_results$Accuracy,

Stepwise = step\_results$Accuracy,

DecisionTree = dt\_conf\_matrix$overall["Accuracy"],

RandomForest = rf\_conf\_matrix$overall["Accuracy"],

Bayesian = bayesian\_results$Accuracy,

SVM = svm\_conf\_matrix$overall["Accuracy"])

# Extract AUC values (already calculated in your code)

auc\_values <- c(Logistic = log\_auc,

Stepwise = step\_auc,

DecisionTree = dt\_auc,

RandomForest = rf\_auc,

Bayesian = bayesian\_auc,

SVM = svm\_auc)

print(auc\_values)

# Create a data frame to summarize model performance

model\_performance <- data.frame(

Model = names(accuracies),

Accuracy = accuracies,

AUC = auc\_values)

# Print the performance table

print("Model Performance Summary:")

print(model\_performance)

# Identify the best model based on Accuracy

best\_accuracy\_model <- model\_performance$Model[which.max(model\_performance$Accuracy)]

best\_accuracy\_value <- max(model\_performance$Accuracy)

# Identify the best model based on AUC

best\_auc\_model <- model\_performance$Model[which.max(model\_performance$AUC)]

best\_auc\_value <- max(model\_performance$AUC)

# Print the best models

cat("\nBest Model by Accuracy:\n")

cat(sprintf("%s (Accuracy = %.3f)\n", best\_accuracy\_model, best\_accuracy\_value))

cat("\nBest Model by AUC:\n")

cat(sprintf("%s (AUC = %.3f)\n", best\_auc\_model, best\_auc\_value))