Project: BUAN 6356.006

Business Analytics With R

HEART ATTACK PREDICTION

Team Members (GROUP – 14)

Yamini Nathani (yxn230000)

Srikanth Venkateshwara Subramanian(sxv230015)

Kajal Singhal (kxs220094)

Chakravarthy Pappu(cxp230005)

Rahul Sadineni(rxs230051)

Table of Contents

Objective:

The escalating prevalence of heart disease in recent years is a multifaceted issue, with factors such as environmental influences, dietary patterns, and diverse lifestyle choices playing pivotal roles. Our primary focus remains on deciphering the key elements that significantly elevate the risk of a heart attack. In our ongoing efforts, we are devising strategies aimed at effectively identifying potential risk factors that contribute to the increased likelihood of individuals experiencing a heart attack. These strategies are integral to our broader objective of not only identification but also the mitigation of these causes. By addressing these risk factors, we aspire to make significant strides in improving overall health. Through a comprehensive analysis of this data, we seek to glean insights that not only aid in understanding how to leverage this information for health improvement but also enable us to identify all conceivable causes contributing to heart-related problems.

Summary:

To formulate our models, our approach involved utilizing R to create models to predict the likelihood of individuals developing heart disease based on specific health indicators. Initially, we conducted a thorough analysis of a diverse health dataset to gain a foundational understanding. Notably, we identified correlations among the variables under examination. Subsequently, we crafted two types of classification models—the decision tree model and logistic regression model—utilizing the dataset. These models were instrumental in discerning whether individuals with particular health indicators were at an elevated risk of developing heart disease. For the decision tree model, we divided the dataset into two portions: one for validation to assess model performance by allocation 20% of records and the remaining 80% as other for training to enhance the model's understanding. We visually represented the decision tree and used charts to evaluate its effectiveness. Similarly, for the logistic regression model, we partitioned the dataset for training and validation, calculated the odds ratio, and assessed the model using charts in ROC analysis. The selection of the most effective model resulted from a meticulous evaluation of each classification model.

Introduction:

Cardiovascular disease poses a major health challenge, as the foremost cause of mortality and disability in the United States that we must address. As the top cause of death and disability in the US, about 695,000 people die of heart disease in the United States every year–that's 1 in every 5 deaths. Every year about 805,000 Americans have a heart attack. Of these, 605,000 are a first heart attack and 200,000 happen in people who have already had a heart attack. It’s estimated to affect 44% of adults by 2030 thereby severely lowering people's quality of life. Lifestyle factors like smoking, lack of exercise, and obesity that strain the heart have slowly raised disease rates, especially in young adults. Understanding current cardiovascular health status and availing resources facilitating lifestyle changes can significantly reduce the likelihood of developing illness. We aim to motivate prevention by analyzing disease prevalence, quantifying death rates leveraging supplied information, and assessing environmental alterations over time. We want to understand what's causing this rise in heart conditions to help prevent it. Figuring out people's current heart health and offering resources to make diet and activity changes can greatly lower their risk. Remembering the heart's vital role in our body, our research aims to provide information to develop better health programs, warn those in danger sooner, and make care more equitable.

Specifically, in our study, we will be exploring various potential business factors that could positively impact the health sector:

1. What are the risk factors leading to the main reasons in people causing heart attacks?
2. If older individuals in comparison to young adults are more prone to attacks or not.
3. Whether high cholesterol further worsens the odds of getting an attack?

Data Description:

The core dataset leveraged in this research was obtained from the Kaggle repository focused on heart disease analysis. Having determined sufficient predictive variables present, no supplemental data collection was pursued. Specifically, the dataset encompasses 14 total attributes – 1 binary target variable indicating disease presence plus 13 categorical and numeric input variables covering relevant risk factors. Descriptions of each feature and associated measurement level (numerical/categorical) are enclosed in the following table:

Given table provides the description of independent variables:

Independent Variable	**Type**	**Description**
age	Numerical	Indicates the age of the patient in years.
sex	Categorical	Indicates the sex of the patient in a binary format ~ 1= male 0= female
cp	Categorical	Chest pain type ~ 0 = Typical Angina 1 = Atypical Angina 2 = Non-anginal Pain 3 = Asymptomatic
trtbps	Numerical	Resting blood pressure (in mm Hg on admission to the hospital)
chol	Numerical	Cholesterol in mg/dl fetched via BMI sensor
fbs	Categorical	(Fasting blood sugar > 120 mg/dl) ~ 1 = True 0 = False
restecg	Categorical	Resting electrocardiographic results ~ 0 = Normal 1 = ST-T wave normality 2 = Left ventricular hypertrophy
thalachh	Numerical	Maximum heart rate achieved in the scale of (71 to 202)
oldpeak	Numerical	ST depression induced by exercise relative to rest.
slp	Categorical	The slope of the peak exercise ST segment 0 = downsloping. 1=flat.

		2=upsloping
caa	Categorical	Number of major blood vessels (0-4)
*-thall	Categorical	Thallium Stress Test result ~ 1= fixed defect 2=reversible defect 3=normal
exng	Categorical	Exercise-induced angina ~ 1 = Yes 0 = No

Given table provides the description of dependent variables:

Dependent Variable	**Type**	**Description**
output	Categorical	0=lesser chance of heart attack 1= higher chance of heart attack

Dataset:

https://www.kaggle.com/code/namanmanchanda/heart-attack-eda-prediction-90-accuracy/data

Data Pre-processing:

Summary of Exploratory Data Analysis and Interpretation:

a) Here, we read the dataset from Kaggle into R as a data frame named heart.df and display the first 6 rows of the dataset.

```
> heart.df <- read.csv("heart.csv")
> head(heart.df)
  age sex cp trtbps chol fbs restecg thalachh exng oldpeak slp caa thall output
1  63   1  3    145  233   1       0      150    0     2.3   0   0     1      1
2  37   1  2    130  250   0       1      187    0     3.5   0   0     2      1
3  41   0  1    130  204   0       0      172    0     1.4   2   0     2      1
4  56   1  1    120  236   0       1      178    0     0.8   2   0     2      1
5  57   0  0    120  354   0       1      163    1     0.6   2   0     2      1
6  57   1  0    140  192   0       1      148    0     0.4   1   0     1      1
>
```

b) Here, we check for possible null values and clean the dataset. On checking, we found the dataset to be clean and there are no null values

```
> sum(!complete.cases(heart.df))
[1] 0
> missing_counts <- colSums(is.na(heart.df));missing_counts
     age      sex       cp   trtbps     chol      fbs  restecg thalachh     exng  oldpeak      slp      caa    thall   output
       0        0        0        0        0        0        0        0        0        0        0        0        0        0
> total_missing <- sum(is.na(heart.df));total_missing
[1] 0
>
```

c) The following represents the summary statistics of all various variables (both numerical and categorical) that are present in the given dataset.

```
> summary(heart.df)
      age              sex               cp             trtbps           chol            fbs              restecg          thalachh
 Min.   :29.00   Min.   :0.0000   Min.   :0.000   Min.   : 94.0   Min.   :126.0   Min.   :0.0000   Min.   :0.0000   Min.   : 71.0
 1st Qu.:47.50   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:120.0   1st Qu.:211.0   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:133.5
 Median :55.00   Median :1.0000   Median :1.000   Median :130.0   Median :240.0   Median :0.0000   Median :1.0000   Median :153.0
 Mean   :54.37   Mean   :0.6832   Mean   :0.967   Mean   :131.6   Mean   :246.3   Mean   :0.1485   Mean   :0.5281   Mean   :149.6
 3rd Qu.:61.00   3rd Qu.:1.0000   3rd Qu.:2.000   3rd Qu.:140.0   3rd Qu.:274.5   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:166.0
 Max.   :77.00   Max.   :1.0000   Max.   :3.000   Max.   :200.0   Max.   :564.0   Max.   :1.0000   Max.   :2.0000   Max.   :202.0
      exng           oldpeak          slp             caa             thall           output
 Min.   :0.0000   Min.   :0.00   Min.   :0.000   Min.   :0.0000   Min.   :0.000   Min.   :0.0000
 1st Qu.:0.0000   1st Qu.:0.00   1st Qu.:1.000   1st Qu.:0.0000   1st Qu.:2.000   1st Qu.:0.0000
 Median :0.0000   Median :0.80   Median :1.000   Median :0.0000   Median :2.000   Median :1.0000
 Mean   :0.3267   Mean   :1.04   Mean   :1.399   Mean   :0.7294   Mean   :2.314   Mean   :0.5446
 3rd Qu.:1.0000   3rd Qu.:1.60   3rd Qu.:2.000   3rd Qu.:1.0000   3rd Qu.:3.000   3rd Qu.:1.0000
 Max.   :1.0000   Max.   :6.20   Max.   :2.000   Max.   :4.0000   Max.   :3.000   Max.   :1.0000
>
```

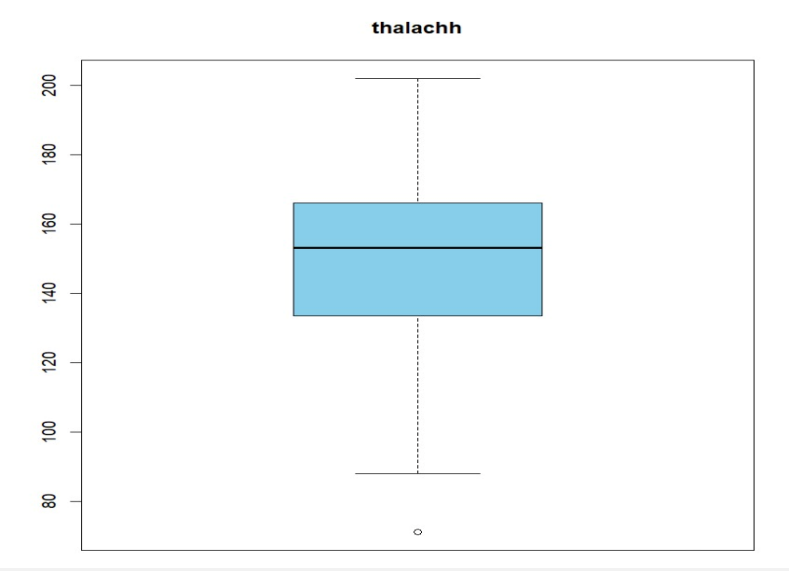
d) We proceed to various plots for visualization of the summary statistics and statistical summary of dataset and find key data-insights into the shape of the data distribution, spread, and presence of outliers.

Following variables are analyzed using boxplot to check for possible outliers:

Thalachh

Maximum Heart Rate Achieved (Scale of 71 – 202)

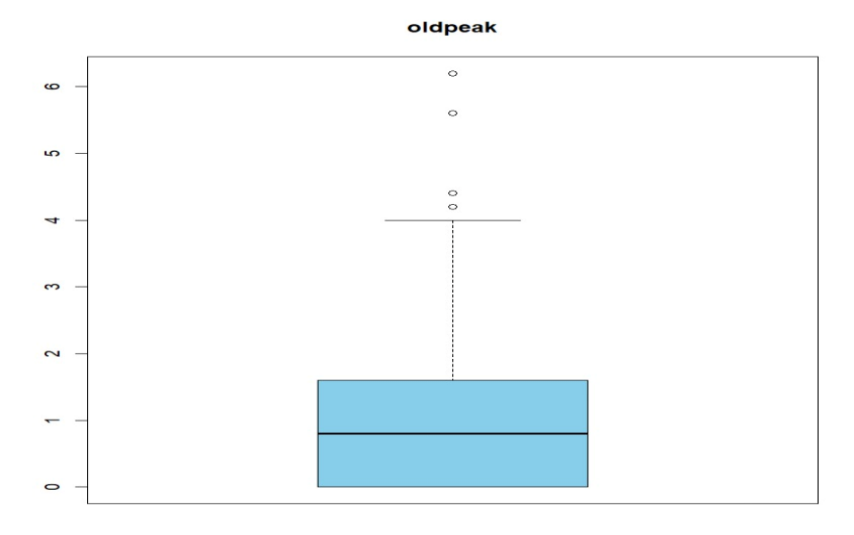
We find one outlier that is below 75, but since it falls within the typical range for resting heart rate, we leave it in place because it shows that the subject is healthy.



Old peak:

ST Depression Induced by Exercise Relative to Rest

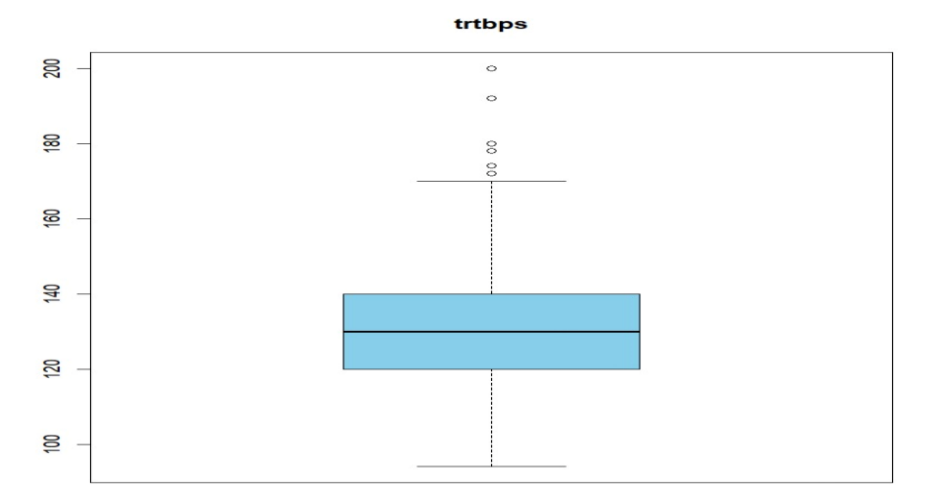
We find four outliers in this variable that are larger than four, but we leave them in place because higher old peak values can be indicative of significant cardiac problems.



Trtbps:

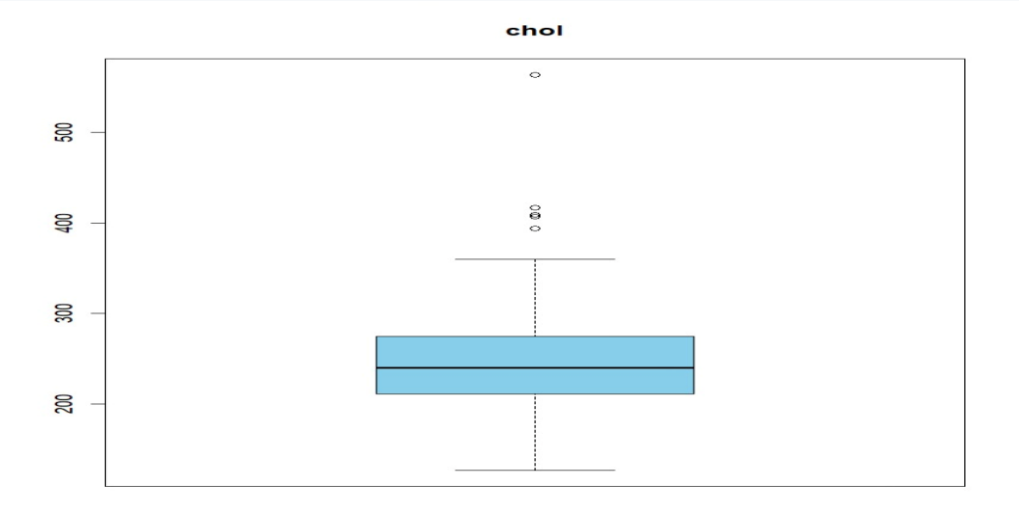
Resting Blood Pressure (in mm Hg)

As we can see, there are six resting blood pressure outliers that we were able to monitor, three of which are 180 or higher. We leave these outliers in place since blood pressure measurements above 180/110 are regarded as hypertensive crises.



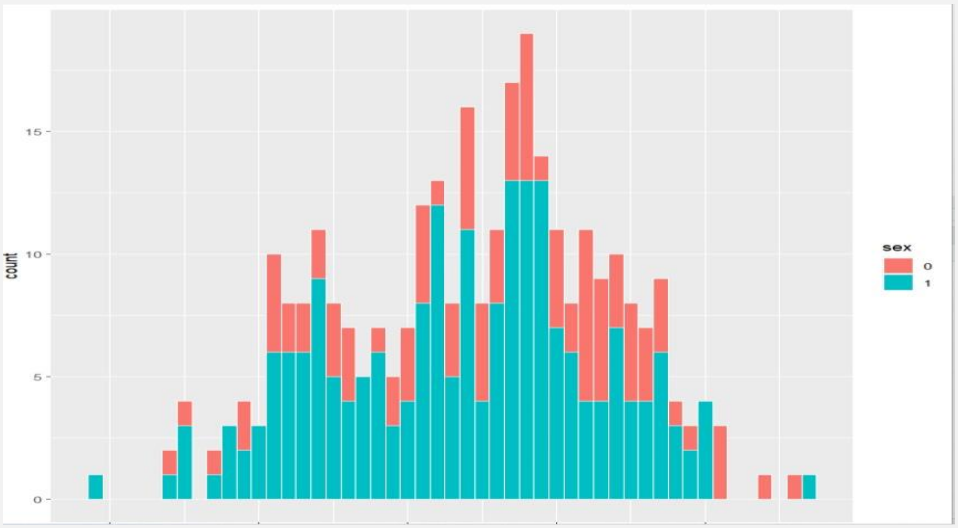
Chol:

Cholesterol (in mg/dl)
Five outliers, or roughly 400 or more, were noted. Since the normal range for cholesterol is between 200 and 240, we leave these outliers in place because they indicate critical levels that need to be addressed right away.



The following variables analysis is done using histogram for single-variable data: Here, we employ Histogram for variable Age and Trtbps as these are numerical (continuous) variables.

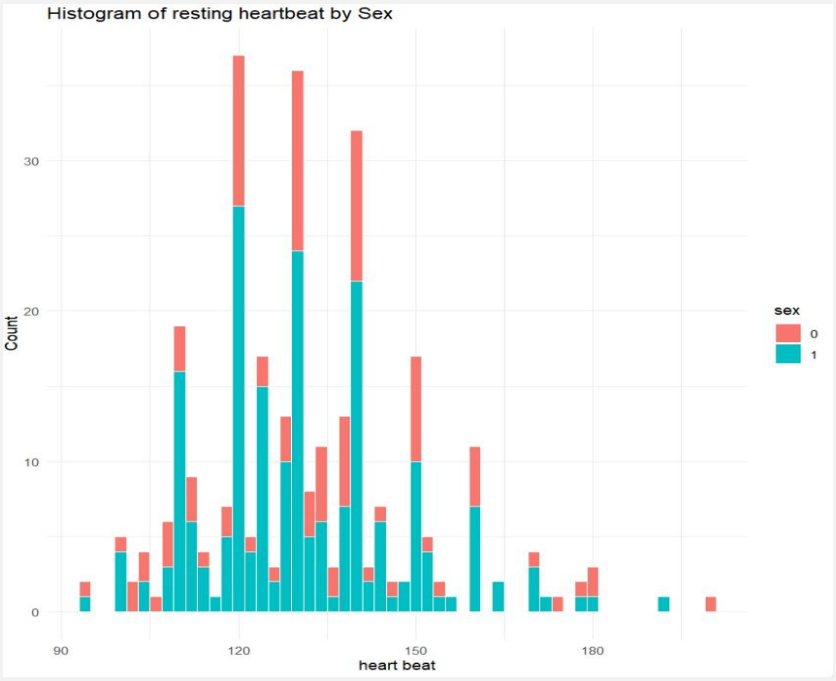
Age:



The graphical depiction of the age-related data is displayed in the histogram below.

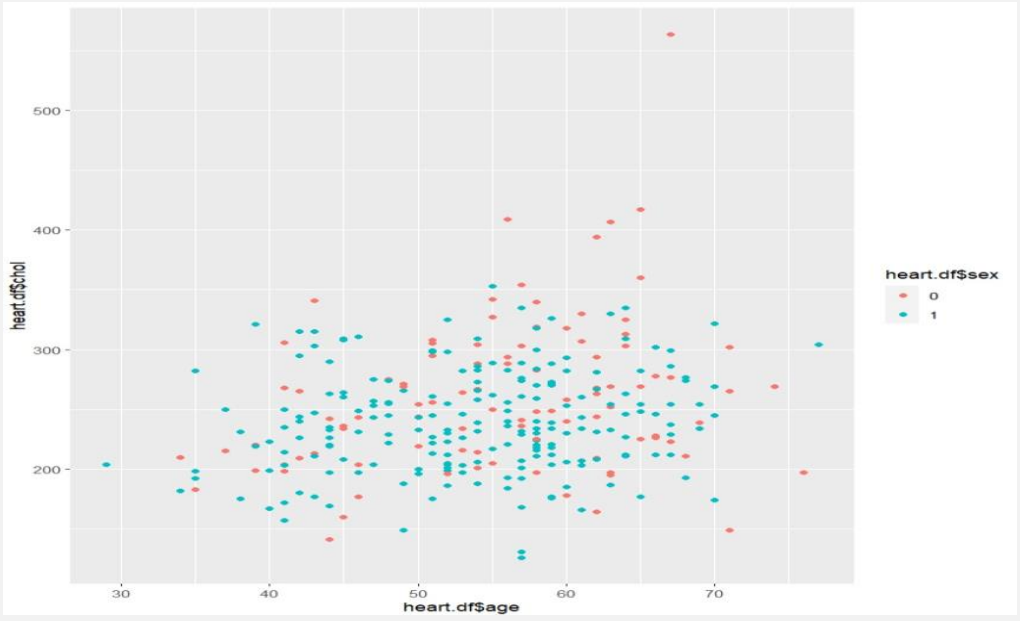
29 years is the minimum, 54.37 years is the mean, and 77 years is the maximum.

Trtbps



The resting blood pressure graphical data is displayed on this graph. It is evident that there is a minimum of 94 units, a mean of 131.6 units, and a maximum of 200 units.

The following variables are analyzed using scatter plots to visualize the relationship between two continuous variables to help identify patterns, trends, correlations, and outliers in bivariate data.



Elderly women between the ages of 60 and 70 are more likely to have cholesterol, as evidenced above. Regarding men, however, it is difficult to determine at what age they are more likely to develop cholesterol.

Data Modeling:

We are employing two classification techniques, namely Decision Tree and Logistic Regression for developing predictive models.

The dataset underwent an initial division into two subsets for the purpose of data modeling in classification: a training dataset and a validation dataset. 80% of the dataset was allocated for training, with the remaining 20% designated for the validation dataset.

Logistic Regression:

Logistic regression is a statistical method used for binary and multiclass classification. Unlike linear regression, which predicts continuous outcomes, logistic regression models the probability that a given instance belongs to a particular class. It's a popular choice for classification problems where the outcome variable is categorical and can take on two or more classes. The logistic model (or logit model) is a statistical model that models the probability of an event taking place by having the log odds for the event be a linear combination of one or more independent variables. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (the coefficients in the linear combination). In the logistic regression equation, the dependent or response variable is represented by logit(pi), while the independent variable is denoted as x. The model's beta parameter, or coefficient, is typically estimated using maximum likelihood estimation (MLE). In this approach, various values of beta are tested through multiple iterations to achieve the optimal fit of log odds. Each iteration contributes to the log-likelihood function, and the objective of logistic regression is to maximize this function to obtain the most accurate parameter estimate.

In this context, the chosen variables— cp, caa, thall followed by old peak, exng—play crucial roles in predicting the likelihood of the outcome. The comparison between evaluation done using null deviation and residual deviation helps us assess how well the chosen variables(predictors) contribute to predicting the response variable compared to a baseline model with no predictor variables. The inclusion of these significant variables aids in constructing a logistic regression model that effectively predicts the likelihood of cardiovascular events based on the given dataset.

Residual deviance represents the unexplained variability in the data and helps us provide the error measurement which quantifies the discrepancy between the observed outcomes and the outcomes predicted by the model. Smaller residual deviance values imply that the model accounts for a larger proportion of the variation in the outcomes, indicating a more accurate representation of the underlying relationships determining the significance of chosen variables, and understanding how well the model explains the observed outcomes.

In logistic regression, the odds ratio signifies the continuous impact of a predictor variable X on the probability of a specific outcome.

We ran logistic regression on the given dataset.
Following provides the summary of Logistic Regression:

```
Call:
glm(formula = output ~ ., family = "binomial", data = train.df)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.125534   2.660386   1.175 0.240058
age         -0.001503   0.025946  -0.058 0.953805
sex         -1.824557   0.522798  -3.490 0.000483 ***
cp           0.986633   0.217172   4.543 5.54e-06 ***
trtbps      -0.020590   0.011217  -1.836 0.066406 .
chol        -0.004876   0.004094  -1.191 0.233680
fbs          0.043432   0.563703   0.077 0.938585
restecg      0.248093   0.373876   0.664 0.506966
thalachh     0.022209   0.011396   1.949 0.051329 .
exng        -0.562949   0.465748  -1.209 0.226778
oldpeak     -0.392256   0.230442  -1.702 0.088720 .
slp          0.706198   0.387930   1.820 0.068694 .
caa         -0.645739   0.212876  -3.033 0.002418 **
thall       -0.883016   0.318091  -2.776 0.005503 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 334.42  on 241  degrees of freedom
Residual deviance: 174.74  on 228  degrees of freedom
AIC: 202.74

Number of Fisher Scoring iterations: 6
```

Following provides the regression coefficients of the Logistic Regression:

```
> heart.train.logit.full$coefficients   ## coefficients of the full logistic regression
 (Intercept)         age         sex          cp       trtbps        chol         fbs      restecg     thalachh        exng
3.125534411 -0.001503010 -1.824557031  0.986633276 -0.020590448 -0.004875909  0.043432102  0.248092616  0.022208501 -0.562948917
    oldpeak         slp         caa       thall
-0.392255714  0.706197583 -0.645739177 -0.883015812
>
```

Further construction of backward and forward models is analyzed using stepwise AIC to choose the best fit.

AIC:

AIC, or Akaike Information Criterion, is a statistical measure used in logistic regression and other model selections to assess the goodness of fit of a model while penalizing for its complexity. AIC is a balance between how well the model explains the data and how simple the model is. The AIC score acts as an informative criterion for model selection that strikes a balance between accuracy and complexity. Lower AIC values indicate a better trade-off between model fit and simplicity. When comparing different models, the one with the lower AIC is generally preferred.

We select the optimal model, considering all significant variables, based on AIC. Although the residual deviation in the model is lower than the null variance in the IC model, it suggests a certain degree of predictive capability for the chosen model.

Below provides the summary of various logistic models constructed using forward selection, backward elimination and Stepwise AIC:

```
> summary(stepwise)

Call:
glm(formula = output ~ oldpeak + cp + caa + thall + exng + sex +
    thalachh + trtbps + restecg + slp, family = "binomial", data = heart.df)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.157907   1.945343   1.109 0.267315
oldpeak     -0.558608   0.211713  -2.639 0.008327 **
cp           0.870475   0.182131   4.779 1.76e-06 ***
caa         -0.763038   0.185060  -4.123 3.74e-05 ***
thall       -0.940441   0.284072  -3.311 0.000931 ***
exng        -0.970656   0.403365  -2.406 0.016111 *
sex1        -1.563590   0.432720  -3.613 0.000302 ***
thalachh     0.022809   0.009395   2.428 0.015185 *
trtbps      -0.020406   0.009988  -2.043 0.041046 *
restecg      0.553402   0.339071   1.632 0.102656
slp          0.564616   0.346525   1.629 0.103236
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 417.64  on 302  degrees of freedom
Residual deviance: 213.08  on 292  degrees of freedom
AIC: 235.08

Number of Fisher Scoring iterations: 6
```

```
> summary(forwards)

Call:
glm(formula = output ~ oldpeak + cp + caa + thall + exng + sex +
    thalachh + trtbps + restecg + slp, family = "binomial", data = heart.df)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.721497   2.020702   1.842 0.065521 .
oldpeak     -0.558608   0.211713  -2.639 0.008327 **
cp           0.870475   0.182131   4.779 1.76e-06 ***
caa         -0.763038   0.185060  -4.123 3.74e-05 ***
thall       -0.940441   0.284072  -3.311 0.000931 ***
exng        -0.970656   0.403365  -2.406 0.016111 *
sex         -1.563590   0.432720  -3.613 0.000302 ***
thalachh     0.022809   0.009395   2.428 0.015185 *
trtbps      -0.020406   0.009988  -2.043 0.041046 *
restecg      0.553402   0.339071   1.632 0.102656
slp          0.564616   0.346525   1.629 0.103236
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 417.64  on 302  degrees of freedom
Residual deviance: 213.08  on 292  degrees of freedom
AIC: 235.08

Number of Fisher Scoring iterations: 6
```

```
> summary(backwards)

Call:
glm(formula = output ~ sex + cp + trtbps + thalachh + oldpeak +
    slp + caa + thall, family = "binomial", data = train.df)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.93478    1.99215   0.971 0.331447
sex         -1.72314    0.48313  -3.567 0.000362 ***
cp           1.07367    0.20125   5.335 9.55e-08 ***
trtbps      -0.02389    0.01047  -2.282 0.022503 *
thalachh     0.02417    0.01020   2.370 0.017792 *
oldpeak     -0.41966    0.22605  -1.857 0.063378 .
slp          0.75322    0.37947   1.985 0.047151 *
caa         -0.64001    0.20420  -3.134 0.001723 **
thall       -0.95692    0.30594  -3.128 0.001761 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 334.42  on 241  degrees of freedom
Residual deviance: 178.40  on 233  degrees of freedom
AIC: 196.4

Number of Fisher Scoring iterations: 6

>
```

Here, we observe that i.e. (mention which one) logistic regression model provides the lowest stepwise AIC among all the models of logistic regression.
However, further analysis is done using accuracy rate using confusion matrix metric in order to make the best selection among the logistic regression models.

Confusion Matrix for logistic model using Full Model selection:

```
    caa + thall
> heart.train.logit.full.predict <- predict(heart.train.logit.full,valid.df,type="response")
> heart.train.logit.full.classes <- ifelse(heart.train.logit.full.predict > 0.5, 1, 0)
> confusionMatrix(as.factor(heart.train.logit.full.classes), as.factor(valid.df$output))
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 20  2
         1  8 31

               Accuracy : 0.8361
                 95% CI : (0.7191, 0.9185)
    No Information Rate : 0.541
    P-Value [Acc > NIR] : 1.184e-06

                  Kappa : 0.6645

 Mcnemar's Test P-Value : 0.1138

            Sensitivity : 0.7143
            Specificity : 0.9394
         Pos Pred Value : 0.9091
         Neg Pred Value : 0.7949
             Prevalence : 0.4590
         Detection Rate : 0.3279
   Detection Prevalence : 0.3607
      Balanced Accuracy : 0.8268

       'Positive' Class : 0

>
```

Confusion Matrix for logistic model using Forward AIC Model selection:

```
> forwards.predict <- predict(forwards,valid.df,type="response")
> forwards.predict.classes <- ifelse(forwards.predict>0.5,1,0)
> confusionMatrix(as.factor(forwards.predict.classes), as.factor(valid.df$output))
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 20  1
         1  8 32

               Accuracy : 0.8525
                 95% CI : (0.7383, 0.9302)
    No Information Rate : 0.541
    P-Value [Acc > NIR] : 2.6e-07

                  Kappa : 0.6972

 Mcnemar's Test P-Value : 0.0455

            Sensitivity : 0.7143
            Specificity : 0.9697
         Pos Pred Value : 0.9524
         Neg Pred Value : 0.8000
             Prevalence : 0.4590
         Detection Rate : 0.3279
   Detection Prevalence : 0.3443
      Balanced Accuracy : 0.8420

       'Positive' Class : 0
```

Confusion Matrix for logistic model using Backward AIC elimination:

```
> backwards.AIC.predict <- predict(backwards.AIC, valid.df, type = "response")
> backwards.AIC.predict.classes <- ifelse(backwards.AIC.predict > 0.5, 1, 0)
> confusionMatrix(as.factor(backwards.AIC.predict.classes), as.factor(valid.df$output))
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 20  2
         1  8 31

               Accuracy : 0.8361
                 95% CI : (0.7191, 0.9185)
    No Information Rate : 0.541
    P-Value [Acc > NIR] : 1.184e-06

                  Kappa : 0.6645

 Mcnemar's Test P-Value : 0.1138

            Sensitivity : 0.7143
            Specificity : 0.9394
         Pos Pred Value : 0.9091
         Neg Pred Value : 0.7949
             Prevalence : 0.4590
         Detection Rate : 0.3279
   Detection Prevalence : 0.3607
      Balanced Accuracy : 0.8268

       'Positive' Class : 0
```

Confusion Matrix for logistic model using Stepwise-AIC (direction =both):

```
> stepwise.predict <- predict(stepwise,valid.df,type="response")
> stepwise.predict.classes <- ifelse(stepwise.predict>0.5,1,0)
> confusionMatrix(as.factor(stepwise.predict.classes), as.factor(valid.df$output))
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 20  1
         1  8 32

               Accuracy : 0.8525
                 95% CI : (0.7383, 0.9302)
    No Information Rate : 0.541
    P-Value [Acc > NIR] : 2.6e-07

                  Kappa : 0.6972

 Mcnemar's Test P-Value : 0.0455

            Sensitivity : 0.7143
            Specificity : 0.9697
         Pos Pred Value : 0.9524
         Neg Pred Value : 0.8000
             Prevalence : 0.4590
         Detection Rate : 0.3279
   Detection Prevalence : 0.3443
      Balanced Accuracy : 0.8420

       'Positive' Class : 0

>
```

Here, we observe that the logistic model constructed using forward selection and Stepwise-AIC in which 10 variables are considered gives us the best accuracy in comparison to the logistic model constructed using backward elimination.

Decision Tree:

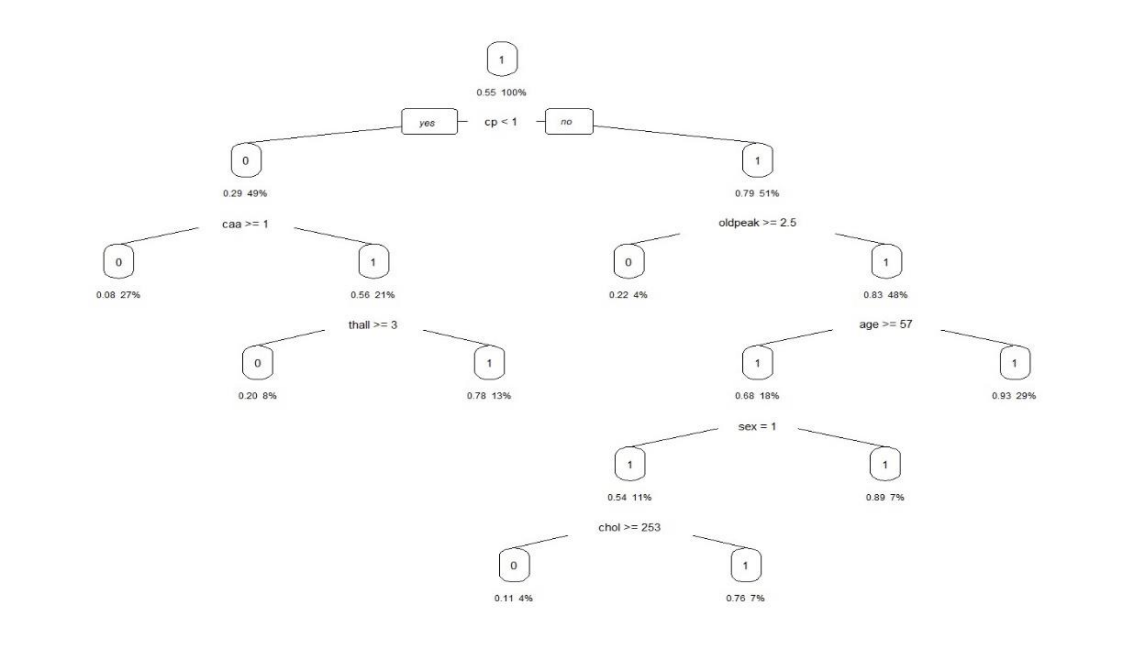
Decision trees are the most prominent and efficient classification techniques due to their transparency, interpretability, and ability to provide rule-based predictions. Decision trees can reveal the importance of different features in the classification process, aiding in feature selection. One of the biggest advantages of decision trees for which they are largely employed for predictive modeling is their interpretability and that they are easily visualized and understood, making them valuable for explaining complex decision-making processes to non-technical stakeholders. The method is efficient in classifying data into distinct categories, and the rules extracted from the tree offer a systematic approach to predicting outcomes for new instances.

Working:

- A decision tree is a tree-like structure that resembles a flow chart.
- Each internal node of the tree represents a test on a specific attribute or feature.
- Branches emanating from internal nodes depict the outcome of the attribute test.
- Each leaf node in the tree represents a class label or an outcome.
- Decision trees inherently generate rules based on the structure of the tree.
- These rules are derived by traversing the tree from the root to the leaf nodes.
- The rules provide a clear understanding of the conditions that lead to a particular classification.

Now we perform the decision-tree classification on the given dataset where the independent/explanatory variables act as the internal nodes on which tests are performed to classify the dependent/outcome variables by top-down recursively partitioning resulting in a decision tree with 5 levels.

Complexity = level of decision tree:



Now, we perform pruning preventing overfitting and improving the generalization ability of the model. Pruning helps address this issue by simplifying the tree, making it more robust and applicable to a broader range of instances.

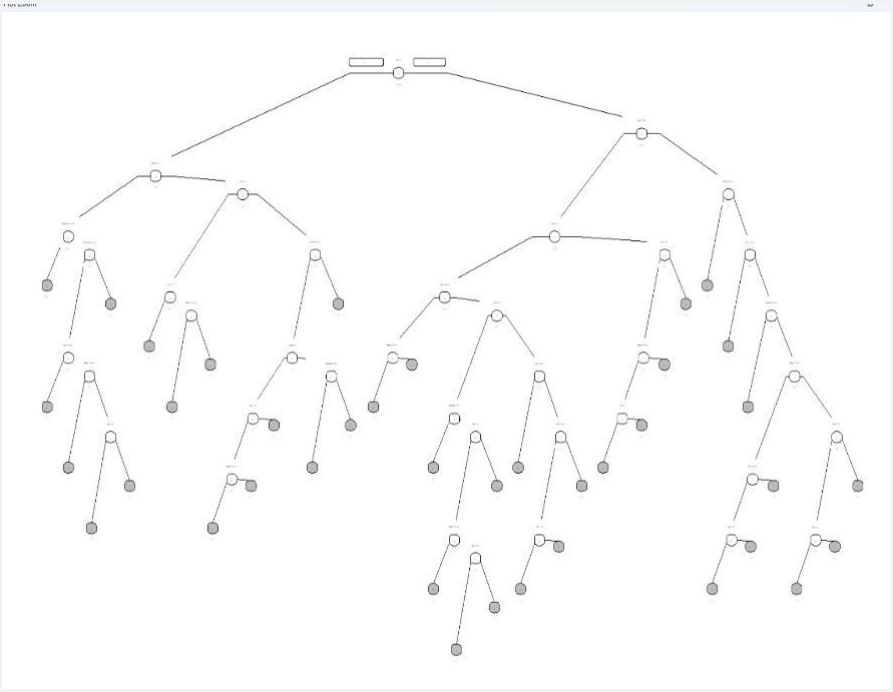
Pre-Pruning:

Pre-pruning is a technique employed in decision tree algorithms to manage the tree's growth by setting conditions to limit node splitting before reaching the maximum tree depth. This approach aims to prevent the tree from becoming overly intricate and overfitting the training data. At each stage of tree splitting, we monitor cross-validation errors. If the error value ceases to decrease, we terminate the tree's growth. The implementation of pre-pruning in the decision tree aims to enhance accuracy.

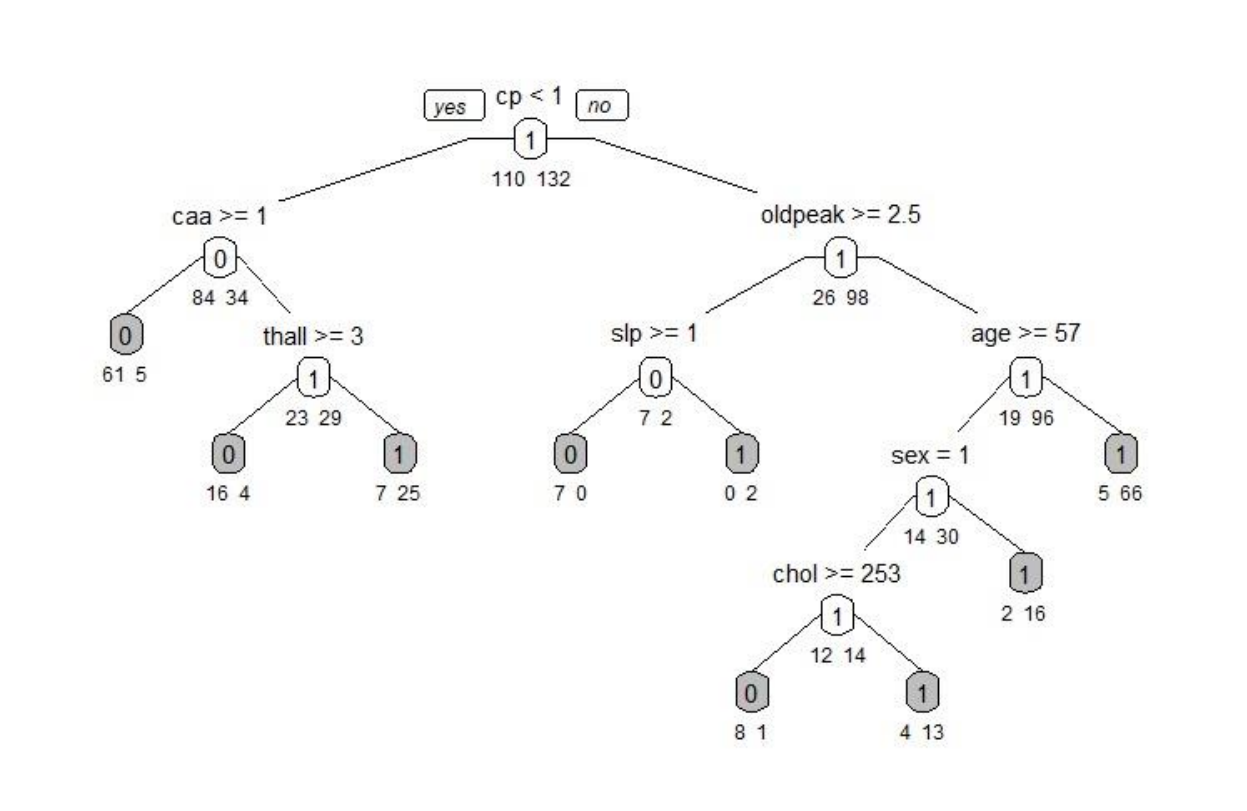
Post -Pruning:

Post-pruning is a crucial technique employed in decision tree algorithms to enhance generalization and combat overfitting. Overfitting occurs when a decision tree captures noise from the training data, leading to poor performance on new data. Post-pruning involves removing branches or nodes after the tree has been fully grown, preventing it from becoming overly complex. This process improves the tree's ability to generalize to unseen data, simplifies the model for better interpretability, and enhances computational efficiency by reducing the tree's size. The decision to prune a specific branch is guided by evaluating the model's performance on a validation dataset, ensuring that the removal of branches positively impacts the tree's predictive accuracy.

Following fig provides the overfitted decision tree:



Below figure provides the pruned Decision Tree:



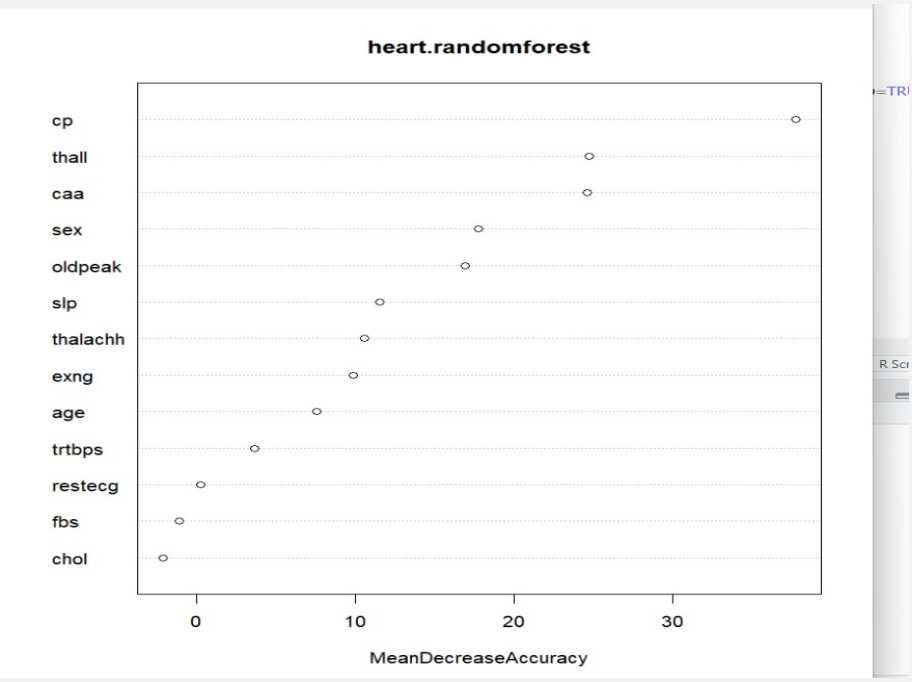
Further, on analysis using cross-validation errors after pre-pruning and post-pruning, we observe that error- values cease to decrease after post-pruning and hence, we decided to halt the tree's growth by removing branches and nodes from the fully grown-tree using the post-pruning method.

Random Forest:

Random Forest is an ensemble learning technique that combines the predictions of multiple individual decision trees to create a more robust and accurate model. Each decision tree in the forest is trained independently on a random subset of the training data and makes its own prediction. The final prediction of the Random Forest is determined by a majority vote or averaging of the predictions made by individual trees. This ensemble approach helps mitigate overfitting and improves the model's generalization performance. Random Forest is known for its versatility, scalability, and ability to handle complex datasets with high-dimensional features. It is widely used for both classification and regression tasks in machine learning, providing a powerful tool for building reliable and accurate predictive models.

The Dataset is given as an input to the Random Forest model to construct 1000 trees. Since all the variables are given. We could construct the Variable Importance Plot showing the Decreasing importance of variables. The plot shows caa, thall, caa to be the most important amongst all in predicting heart attack.

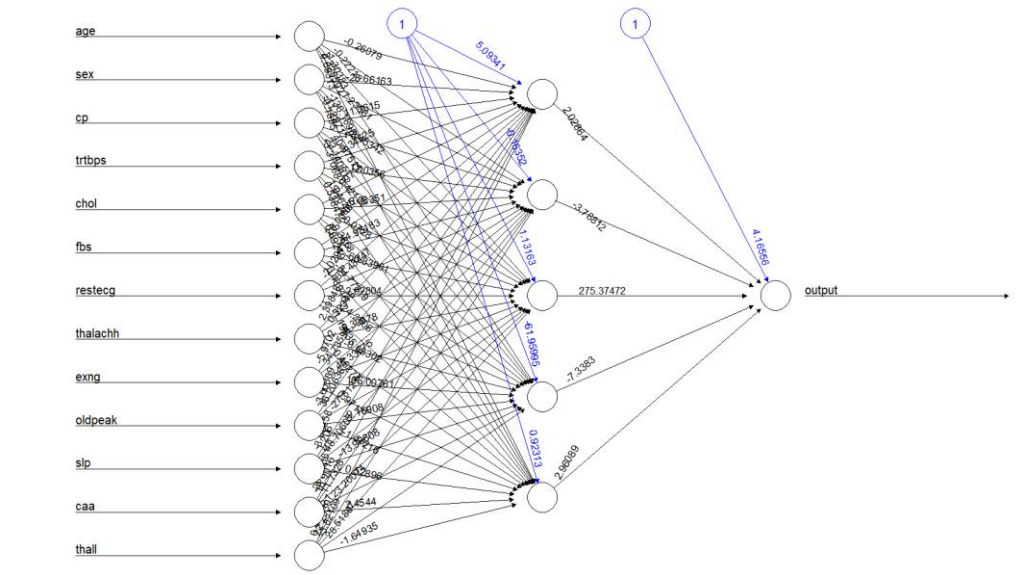
The following Figure provides the Variable importance plot i.e. significance of the variables



Neural Network:

Neural networks are computational models inspired by the human brain, composed of interconnected nodes organized into layers—input, hidden, and output. Each node applies an activation function to introduce non-linearity and facilitate complex pattern recognition. During training, weights and biases are adjusted through backpropagation, minimizing the difference between predicted and actual outcomes. The process involves data preparation, model definition specifying layers and activation functions, training with backpropagation, evaluation using metrics like accuracy, and application to new data for predictions.

The dataset is given as an input to the Neural Network, where augments were provided to create 5 hidden layers. The following is the output.



Performance Evaluation:

In the context of decision tree analysis in terms of model performance, we employed a confusion matrix, cross-validation dataset, and the Area Under the Curve (AUC) derived from the Receiver Operating Characteristic (ROC) curve to comprehensively evaluate the performance of the decision tree model. The confusion matrix allows for a detailed examination of the model's predictions, breaking down the instances of true positives, true negatives, false positives, and false negatives. This detailed assessment helps me gauge the precision, recall, and accuracy of the decision tree in classifying instances. Additionally, the use of a cross-validation dataset enhances the robustness of the evaluation, as it involves systematically partitioning the dataset into multiple subsets for training and testing. By iteratively assessing the model's performance across various folds, thereby can obtain a more reliable measure of its generalization capability and overall effectiveness in making accurate predictions on new, unseen data.

In the context of logistic regression analysis in terms of model performance, we are leveraging key evaluation metrics such as the confusion matrix employing stepwise variable selection with AIC, and the Area Under the Curve (AUC) derived from the Receiver Operating Characteristic (ROC) curve. The confusion matrix provides a detailed breakdown of classification performance, distinguishing between true positives, true negatives, false positives, and false negatives. Simultaneously, we employ AIC-guided stepwise variable selection, systematically refining the model for improved interpretability while guarding

against overfitting. Enhancing our evaluation toolkit, the AUC of the ROC curve offers a comprehensive measure of the model's discriminatory ability across varying classification thresholds. This combination of methodologies ensures a comprehensive evaluation of our logistic regression model, considering accuracy, model simplicity, and the nuanced balance between sensitivity and specificity.

Similarly for context of Random Forest analysis in terms of model performance, we are leveraging key evaluation metrics such as the confusion matrix, and the Area Under the Curve (AUC) derived from the Receiver Operating Characteristic (ROC) curve.

Similarly for context of Neural network analysis in terms of model performance, we are leveraging key evaluation metrics such as the confusion matrix, and the Area Under the Curve (AUC) derived from the Receiver Operating Characteristic (ROC) curve.

Confusion Matrix:

A confusion matrix visualizes the performance of a classification model by tallying correct and incorrect predictions to quantify accuracy alongside error types. We apply this for our cardiac disease classifier to assess predictive capacity on the training data. The matrix compares actual to predicted labels, splitting predictions into true positives, true negatives, false positives and false negatives. This supplies an intuitive snapshot of overall correctness, while also tracking key errors like false alarms vs missed cases to help refine classification thresholds based on diagnostic priorities.

Below provides the confusion matrix for pruned decision tree:

```
> confusionMatrix(pruned.tree, as.factor(valid.df$output))
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 20  1
         1  8 32

               Accuracy : 0.8525
                 95% CI : (0.7383, 0.9302)
    No Information Rate : 0.541
    P-Value [Acc > NIR] : 2.6e-07

                  Kappa : 0.6972

 Mcnemar's Test P-Value : 0.0455

            Sensitivity : 0.7143
            Specificity : 0.9697
         Pos Pred Value : 0.9524
         Neg Pred Value : 0.8000
             Prevalence : 0.4590
         Detection Rate : 0.3279
   Detection Prevalence : 0.3443
      Balanced Accuracy : 0.8420

       'Positive' Class : 0
```

Below is the confusion matrix for Random Forest:

```
> heart.randomforest.predict <- predict(heart.randomforest,valid.df)
> confusionMatrix(heart.randomforest.predict,as.factor(valid.df$output))
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 18  0
         1 10 33

               Accuracy : 0.8361
                 95% CI : (0.7191, 0.9185)
    No Information Rate : 0.541
    P-Value [Acc > NIR] : 1.184e-06

                  Kappa : 0.6607

 Mcnemar's Test P-Value : 0.004427

            Sensitivity : 0.6429
            Specificity : 1.0000
         Pos Pred Value : 1.0000
         Neg Pred Value : 0.7674
             Prevalence : 0.4590
         Detection Rate : 0.2951
   Detection Prevalence : 0.2951
      Balanced Accuracy : 0.8214

       'Positive' Class : 0
```

Below is the confusion matrix for logistic regression model constructed using stepwise AIC

```
> stepwise.predict <- predict(stepwise,valid.df,type="response")
> stepwise.predict.classes <- ifelse(stepwise.predict>0.5,1,0)
> confusionMatrix(as.factor(stepwise.predict.classes), as.factor(valid.df$output))
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 20  1
         1  8 32

               Accuracy : 0.8525
                 95% CI : (0.7383, 0.9302)
    No Information Rate : 0.541
    P-Value [Acc > NIR] : 2.6e-07

                  Kappa : 0.6972

 Mcnemar's Test P-Value : 0.0455

            Sensitivity : 0.7143
            Specificity : 0.9697
         Pos Pred Value : 0.9524
         Neg Pred Value : 0.8000
             Prevalence : 0.4590
         Detection Rate : 0.3279
   Detection Prevalence : 0.3443
      Balanced Accuracy : 0.8420

       'Positive' Class : 0

>
```

Below is the confusion matrix for the Neural Network:

```
> nn.pred <- predict(nn.heart, valid.df, type = "response")
> nn.pred.classes <- ifelse(nn.pred > 0.5, 1, 0)
> confusionMatrix(as.factor(nn.pred.classes), as.factor(valid.df$output))
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 21  7
         1  7 26

               Accuracy : 0.7705
                 95% CI : (0.645, 0.8685)
    No Information Rate : 0.541
    P-Value [Acc > NIR] : 0.0001784

                  Kappa : 0.5379

 Mcnemar's Test P-Value : 1.0000000

            Sensitivity : 0.7500
            Specificity : 0.7879
         Pos Pred Value : 0.7500
         Neg Pred Value : 0.7879
             Prevalence : 0.4590
         Detection Rate : 0.3443
   Detection Prevalence : 0.4590
      Balanced Accuracy : 0.7689

       'Positive' Class : 0
```

Below provides the complexity -parameter table of cross- validation errors:

```
> printcp(cv.ct)

Classification tree:
rpart(formula = output ~ ., data = train.df, method = "class",
    cp = 1e-06, minsplit = -1, xval = 5)

Variables actually used in tree construction:
 [1] age      caa      chol     cp       exng     fbs      oldpeak  restecg  sex      thalachh thall    trtbps

Root node error: 113/242 = 0.46694

n= 242

         CP nsplit rel error  xerror     xstd
1 0.5132743      0  1.000000 1.00000 0.068683
2 0.0619469      1  0.486726 0.48673 0.057692
3 0.0176991      3  0.362832 0.40708 0.054016
4 0.0132743      9  0.247788 0.56637 0.060717
5 0.0088496     11  0.221239 0.62832 0.062682
6 0.0044248     34  0.017699 0.64602 0.063186
7 0.0000010     38  0.000000 0.65487 0.063428
>
```

Integrating a cross-validation dataset adds an extra layer of reliability to the assessment process. This approach enhances the robustness of the evaluation, offering a more dependable measure of the decision tree's ability to generalize and make accurate predictions on new, unseen data.

Analysis Interpretation:

We observe that among all, the Logistic Regression constructed using Stepwise-AIC classifier provides the highest accuracy rate of 85.25% compared to rest of the classifier models i.e., Random Forest classifier giving 82%, Pruned Decision Tree giving 85% and Neural Network giving 77.07% accuracy rate giving thereby telling us which model performing the best based on Accuracy rate among all the classifier models in predicting the heart attack.

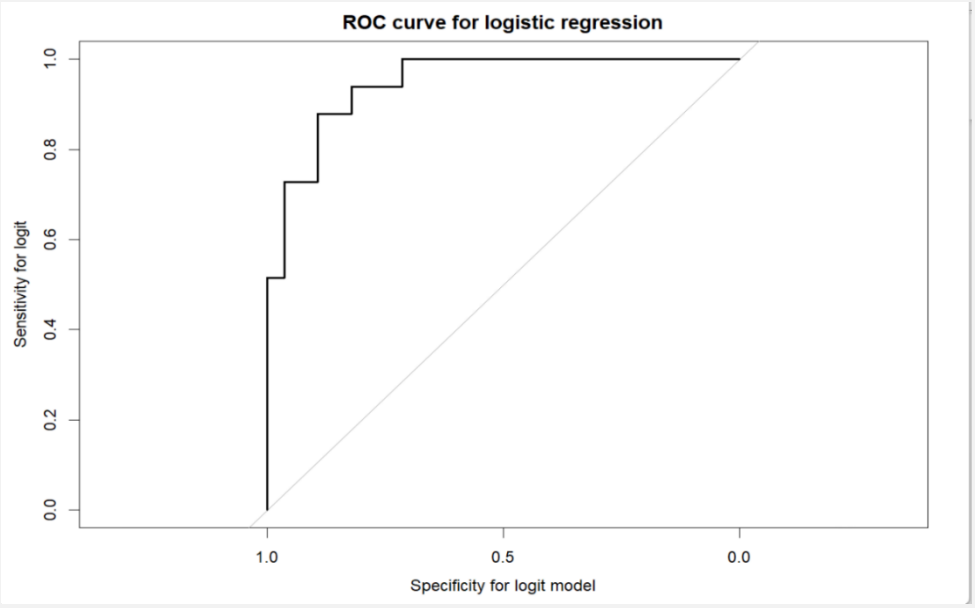
ROC curve:

The ROC curve and AUC metric let us evaluate how well our model can correctly label prone-to-heart-attack cases vs. non-prone-to-heart-attack cases. We integrate these to assess our cardiac disease classification model. ROC curves plot the true positive rate against the false positive rate across output probabilities, compared to a random baseline. AUC measures separability - the model's capacity to distinguish between classes.

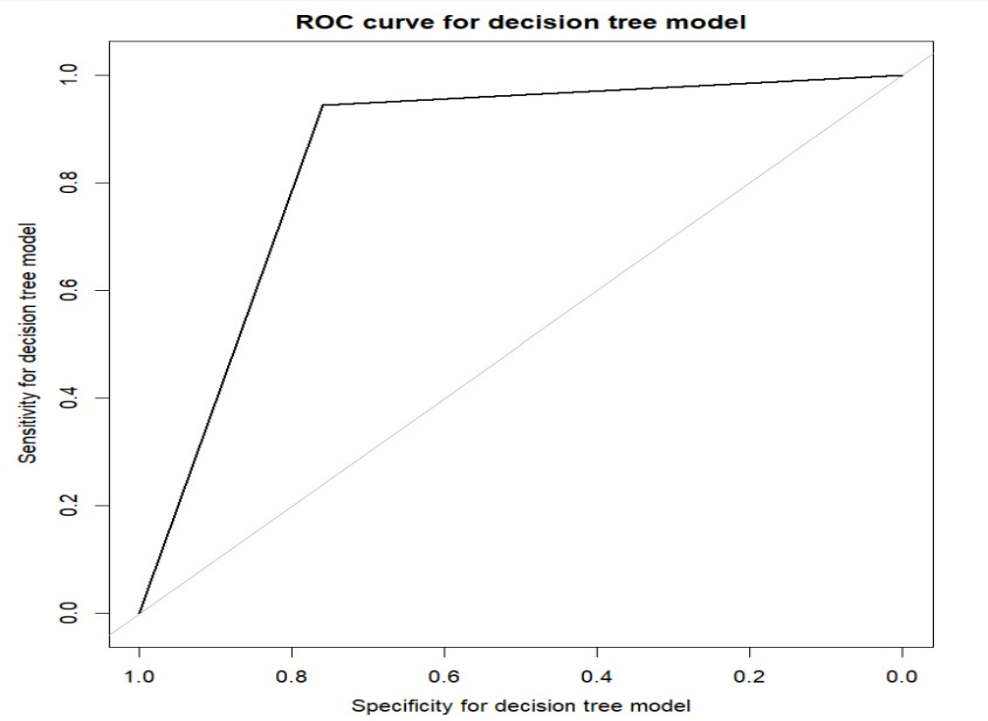
The ROC curve graphs the tradeoff between the true positive rate (correctly predicting those with prone-to-heart-attack) and the false positive rate (incorrectly predicting non-prone-to-heart-attack when absent) at different output thresholds. A good model will maximize true positives while minimizing false alarms.

AUC measures how well the model separates the groups over all possible thresholds. An AUC of 1 means perfect classification. An AUC of 0.5 is as good as random guessing. Values in between show varying levels of discrimination between people prone to and not prone to heart attack. We use these because accuracy alone could be misleading if groups are very unequal.

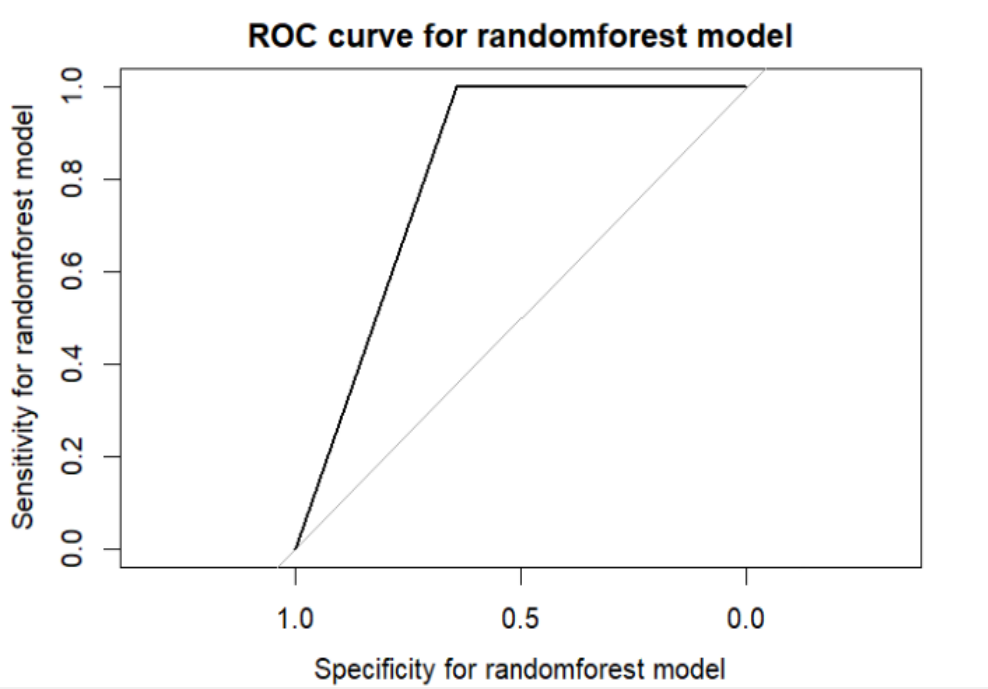
Below if the ROC curve for the Stepwise logistic regression model with an AUC (Area under Curve) of 0.948



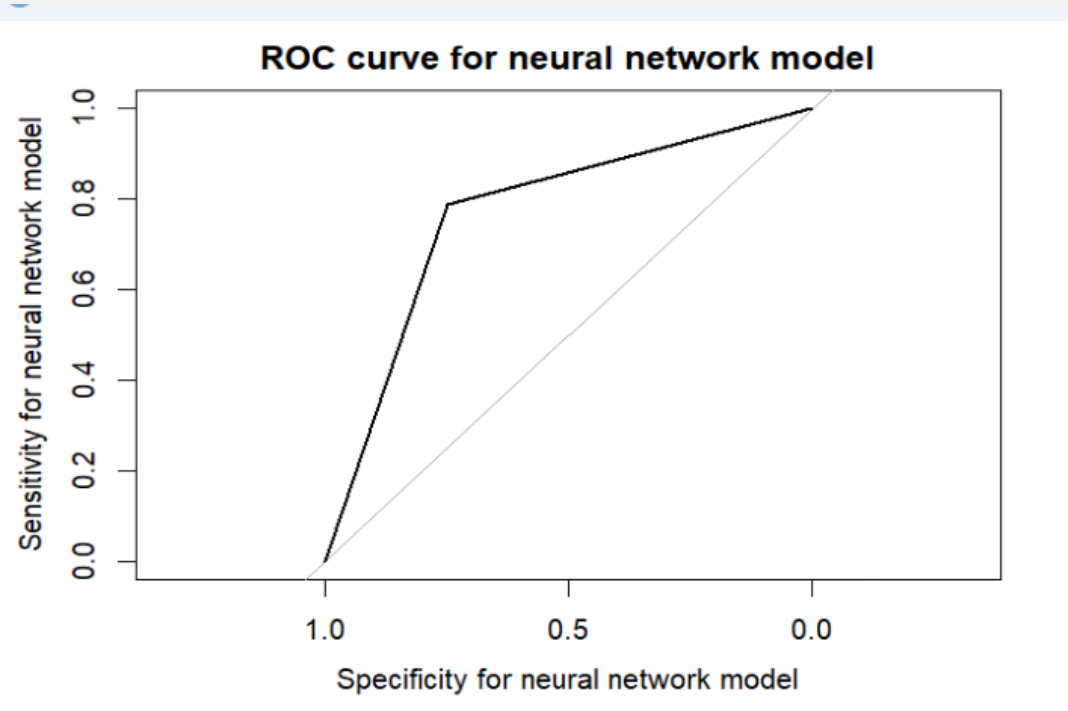
Below provides the ROC curve for the decision tree model with an AUC of 0.848:



Below is the ROC curve for Random Forest Model with AUC of 0.809



Below is the ROC curve for neural network with an AUC of 0.7689:



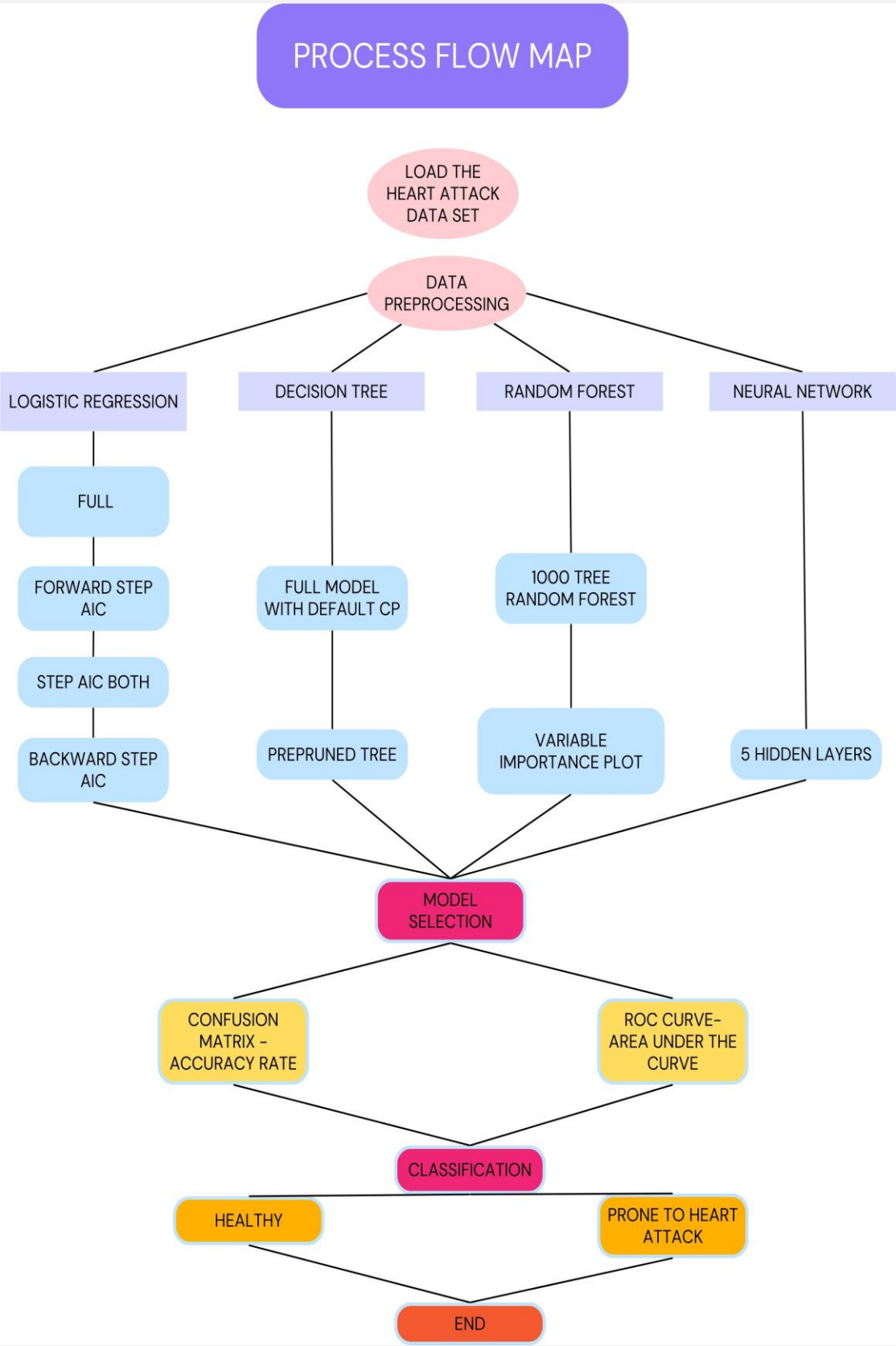
Analysis Interpretation:

The ROC curve visually demonstrates the trade-off between sensitivity and specificity at different classification thresholds. A model with better performance will have a curve that is closer to the top-left corner, indicating higher true positive rates and lower false positive rates across various threshold values.

AUC represents the area under the ROC curve. It is a single value summarizing the model's performance across various threshold settings.

Here, we observe that among all, the Logistic Regression constructed using the Stepwise-AIC classifier provides the highest AUC value of 0.948 compared to rest of the classifier models i.e. Random Forest classifier giving 0.809, Pruned Decision Tree giving 0.848 and Neural Network classifier 0.7689 giving thereby telling us which model performing the best based on Accuracy rate among all the classifier models.

Process Flow Map:



Conclusion:

Model	Accuracy	Recall	AUC
Decision Tree	85%	71.43%	0.848
Random Forest	82%	64.29%	0.809
Neural Network	77.05	75.00%	0.7689
Logistic Regression	85.2%	71.43%	0.948

Evaluation:

Accuracy: Overall records classified correctly
Recall: Of actual positive cases, how many were correctly predicted
AUC: Model discrimination ability

We have chosen Logistic Regression forward Model also for reasons apart from Accuracy rate and better Area under the Curve that

1. It is more robust and doesn't need extensive hyperparameter tuning
2. Has chosen only significant variables or parameters thus clearly elucidating the significant parameter that affects our desired prediction.
3. The Training time is quite quicker
4. It performs well with Datasets of varied sizes, and we can increase our dataset dynamically
5. Can tell very well how a unit change in the parameter causes the changes in the prediction. This is particularly useful for our model in educating people about what factors should be taken care of for reducing heart attack occurrences.

Confusion matrix accuracy only measures overall correct classifications. It does not account for how well the model discriminates between the positive and negative classes. AUC specifically quantifies the discriminative ability and evaluates performance across all probability thresholds, so lower AUC indicates poorer separation between the classes despite overall accuracy.

We developed and evaluated various classification models such as Logistic Regression, Decision Tree, Random Forest and Neural Network models to predict prone to heart attack risk using various patient health parameters from the given dataset. Among all the models, the logistic regression model classifier approach achieved superior accuracy of 85.2% by employing both forward selection and backward elimination approaches, outperforming all other classification models accuracy (Random Forest, Decision Tree and Neural Network). Beyond overall correctness, for our given dataset we found that Logistic regression is well-suited for prediction when the target variable is binary, as is the case for heart disease risk (0 - no disease or 1 - has disease). It models the probability of the 1 outcome directly without imposing overly rigid assumptions. Logistic regression also outputs interpretable coefficient values indicating the influence of each specific patient attribute on heart disease odds. These supports arriving at nuanced medical inferences. Lastly, the ROC curve and AUC for logistic regression quantified its reliable discriminative capacity distinguishing between those with and without disease. Diagnostic predictive systems require both accuracy and differentiation ability to justify clinical usage. Our implementation

pinpointed Thallium Stress Test result- Thall, caa- Number of major blood vessels, cp- Chest Pain Type, exng- Exercise induced angina and oldpeak- ST depression induced by exercise as top predictors - suggesting cholesterol and age are less indicative and carries almost no importance than other vital signs.

The logistic regression model developed in this heart disease analysis demonstrates reliable predictive capacity for stratifying patients by risk level based on an expansive set of demographic, symptom and vital sign predictor variables. Extending the methodology by retraining the algorithms on larger, more representative clinical datasets can further improve personalized probabilistic risk profiling. If productized through mobile platforms, such data-driven systems flagging those likely to decompensate can prompt preventive interventions or urgent treatment earlier. This would enable preemptive care based on analytics-generated foresight into outcomes. Our methodology lays basis for scalable, equitable risk forecasting by learning from population patterns - serving not just cardiovascular conditions but also assisting more vulnerable groups manage other major diseases. Leveraging ever-growing health data along with computational methods is key to guiding clinical decisions through insight into how known risk factors translate statistically into real-world events.