Northeastern University
College of Professional Studies

ALY6040: DATA MINING APPLICATIONS

MODULE 6 Final Project

GROUP MEMBERS:

Ansh Vipul Aya - aya.a@northeastern.ed,

Deep Anish Gada - gada.de@northeastern.edu,

Shreyas Mallesh - mallesh.s@northeastern.edu,

Hanirvesh Reddy Chillakuru - chillakuru.h@northeastern.edu

GROUP NUMBER: 6

PROFESSOR NAME: Kasun Samarasinghe

DATE: 05/16/2022

Abstract

Chronic illnesses and ailments, such as heart disease, stroke, cancer, type 2 diabetes, obesity, and arthritis, are among the most frequent, expensive, and avoidable health issues. The Centers for Disease Control and Prevention (CDC) estimates that chronic diseases cause 7 out of every 10 deaths each year, and as of 2012, about half of all adults in the United States — 117 million individuals — had one or more chronic health conditions. Heart disease is causing an increase in mortality in today's health trend. By detecting heart illness on samples of health patient sources received from medical clinics, lives can be saved and fatalities reduced. As a follow-up, the relevant therapies must be instructed and recommended. One of the major elements anticipated for forecasting cardiac problems in advance is accuracy. Many approaches were studied and compared with few factors based on this aspect. We discovered and forecasted human heart illness using a range of machine learning algorithms and used the heart disease dataset to evaluate its performance using various metrics, according to the suggested study. We used Linear Regression, Logistic Regression, Neural Networks and Random forest. The approaches outlined in terms of their working theme and their adequacy are acknowledged.

Introduction

The Cleveland Heart Disease dataset from the UCI repository was utilized in this work, and it contains 303 persons and a range of health-related factors. Columns(14) in the dataset are named age, sex, cp, trestbps, chol, fbs, restecg, thalach, exang, oldpeak, slope, ca, thal, and target. This study's charts provide essential information in the form of visualizations.

exercise-induced angina (1 = yes; 0 = no)

ca: the number of large boats (0-3)

CP stands for Chest Pain.

Value 1: standard angina

Atypical angina (value 2)

3rd value: non-anginal pain

trtbps 4: asymptomatic resting blood pressure (in mmHg)

chol: cholesterol in milligrams per deciliter obtained by BMI sensor fbs: (fasting blood sugar > 120 milligrams per deciliter) (1 = true; 0 = false)

rest ecg: electrocardiographic findings at rest

0 is the default value.

Value 1: having an aberrant ST-T wave (T wave inversions and/or ST elevation or depression of more than 0.05 mV).

According to Estes' criterion, value 2 indicates probable or definite left ventricular hypertrophy.

thalach's maximum heart rate attained goal: 0 = less likely to have a heart attack 1 = increased risk of heart attack n

Business Objective

The major goal is to accurately forecast a patient's cardiac status so that additional therapy can be beneficial. As a result, predictive analysis will be used to estimate the chance of people developing cardiac disease. It enables considerable knowledge to be established, such as correlations between medical variables connected to heart disease and patterns.

EDA

Prior to undertaking analysis, we ensured that the data is accurate, comprehensive, and consistent. This implied that all required data was supplied in a precise and consistent format.

Installing all the required packages and loading the respective libraries.

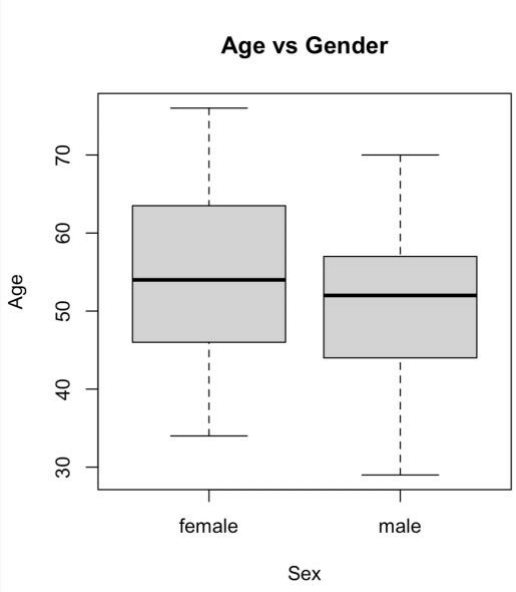
```
#Installing packages and importing libraries
install.packages("caret")
install.packages("neuralnet")
library(dplyr)
library(neuralnet)
library(caret)
library(ggplot2)
```

The summary of the heart dataset

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

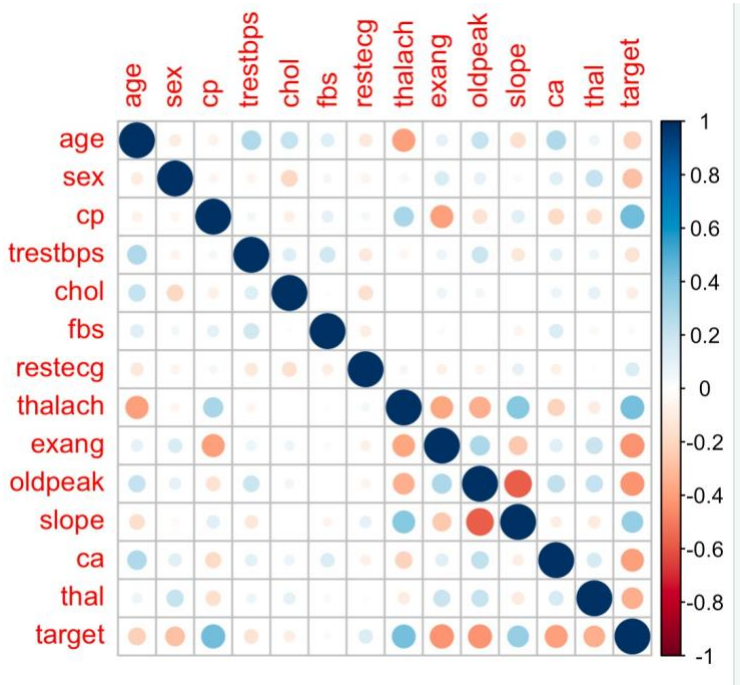
It shows that the age range of the dataset is 29-77 similarly it can be determined for all variables.

The Boxplot for Age vs gender



The age range for males and females with cardiac disease is depicted in the box plot above. Males are found to be more likely than females to develop cardiac disease at a younger age.

Next, the correlation plot was constructed to help understand the patterns and correlations between the individual factors and the target variable



When compared to other stated characteristics, parameters such as chest pain, maximum heart rate, angina exercises, oldpeak, and ca had a substantial influence on the goal variable.

Data Analysis

The following Machine learning techniques were implemented to perform Data analysis:

Linear Regression:

A simple linear regression model examines the connection between a response variable (commonly referred to as y) and a predictor variable (x). Multiple linear regression is an extension of simple linear regression that uses multiple distinct predictor variables to predict an outcome variable. For this assignment we have implemented Multiple linear regression.

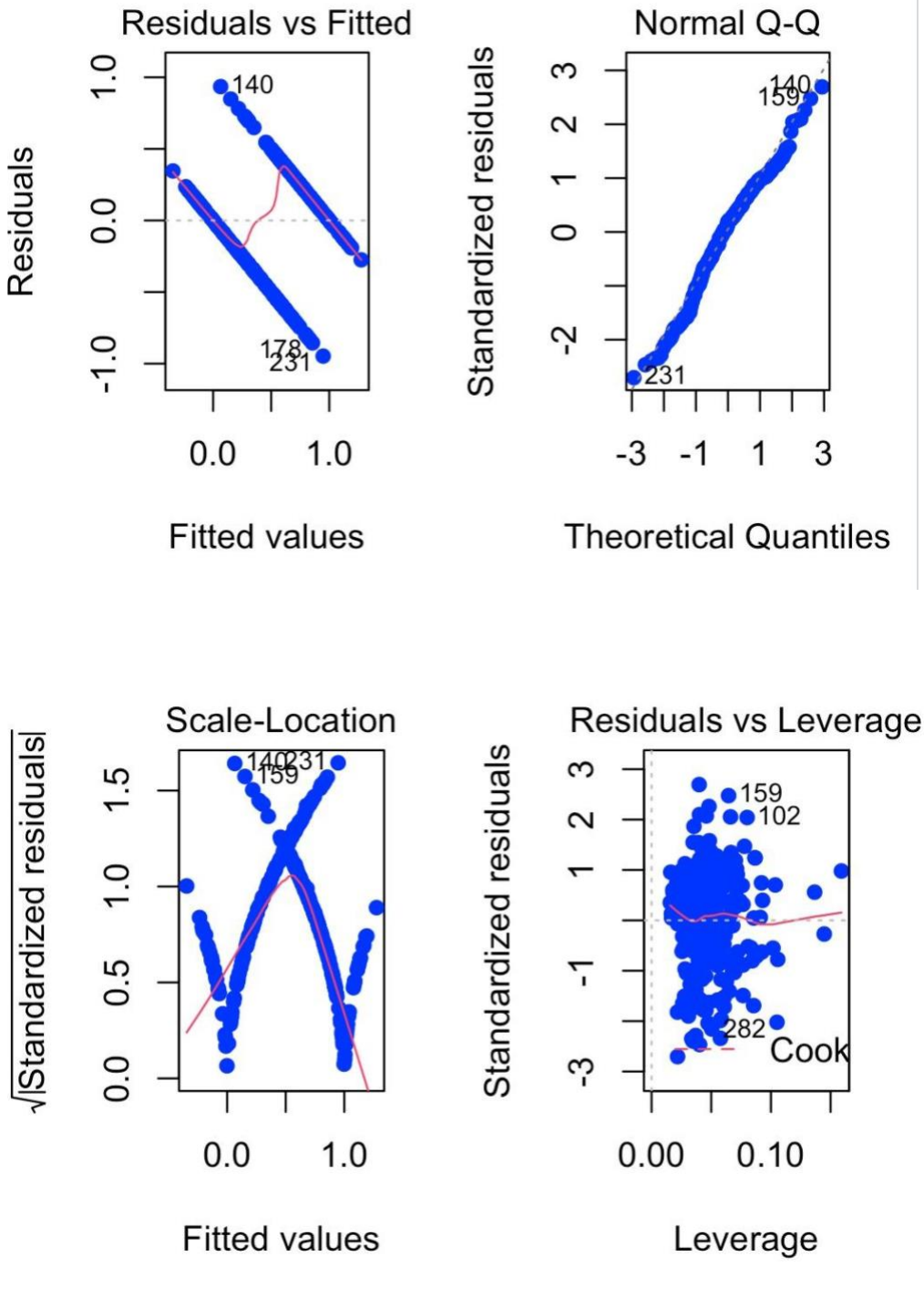
We want to build a model that can predict if a person will have a heart attack based on parameters like cholesterol, blood pressure, blood sugar, etc. So, we developed a model with all of the possible parameters and then we got the following results.

```
> lmoutput = lm(target~., data = data)

> summary(lmoutput)

Call:
lm(formula = target ~ ., data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.94748 -0.21270  0.06608  0.25022  0.93509

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.8115744  0.2133689   3.804 0.000174 ***
age         -0.0008204  0.0026962  -0.304 0.761129
sex         -0.1959956  0.0471429  -4.157 4.24e-05 ***
cp           0.3381102  0.0671449   5.036 8.40e-07 ***
trestbps    -0.2110423  0.1332781  -1.583 0.114407
chol        -0.1548500  0.1847151  -0.838 0.402545
fbs          0.0173736  0.0596669   0.291 0.771125
restecg      0.0996959  0.0798457   1.249 0.212819
thalach      0.3955321  0.1480767   2.671 0.007988 **
exang       -0.1440459  0.0513689  -2.804 0.005387 **
oldpeak     -0.3644899  0.1421467  -2.564 0.010847 *
slope        0.1579576  0.0847791   1.863 0.063453 .
ca          -0.4024086  0.0874260  -4.603 6.25e-06 ***
thal        -0.3571175  0.1069651  -3.339 0.000952 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3542 on 289 degrees of freedom
Multiple R-squared:  0.5175,    Adjusted R-squared:  0.4958
F-statistic: 23.85 on 13 and 289 DF,  p-value: < 2.2e-16
```

R-squared value was found to be 49%. It means that the independent variable was able to explain 49% of the variance in the dependent variable. Hence, it was considered as an average model.

We also tried to optimize the model by using only a few significant parameters, but the accuracy obtained in the first model was better as compared to the other models.

We also implemented regression diagnostics for this model. We got the following visualizations.



The points are spread in a somewhat random manner in the first plot. As a result, the linearity can be classified as normal.

The points in the second plot follow a pattern similar to that of a regression line, in that they align with it. In addition, as you can see, the plot contains a few outliers. However, the model is

considered normal because the outliers have no significant impact on the model. As a result, the outliers do not need to be removed.

Homoscedasticity, or constant variance, is defined in the third plot. There is no discernible pattern. All the points are shown in a random order.

Now, since we couldn't meet the desired result from this model which was to determine the impact of significant variables on the response variable, target. So, we implemented logistic regression to perform a better predictive analysis.

Logistic Regression:

Logistic regression is frequently used in predictive analytics and modeling, as well as machine learning applications. The dependent variable is finite or categorical in this approach: 1 or 2. (binary regression). By estimating probabilities using a logistic regression equation, it is employed in statistical software to comprehend the relationship between the dependent variable and one or more independent variables.

This form of analysis can assist you in predicting the chances of an occurrence or a decision occurring. For example, you might be curious about your chances of having a heart attack — or not (dependent variable). We have used a heart dataset to implement a Logistic regression model here which can help you figure out how likely you are to have a heart attack. Consequently, you will be able to make smarter judgments and gain a greater understanding of what triggers a heart attack.

Before creating models, we did data Wrangling where we adjusted the variable type of data by creating various levels and labels to check if there are any missing values in the dataset. We have then Pre-processed the data using prop.table to find the actual number of the variable target class. We have also done Cross validation where we split the data into Training and Testing to ensure that the proportion is enough to balance the data so that there is no risk involved if our model is overfit.

First, we have created a model without predictor as Model_1

Secondly, we have created a model with predictor as Model_2

```
> model_1

Call:  glm(formula = target ~ 1, family = "binomial", data = training)

Coefficients:
(Intercept)
     0.2932

Degrees of Freedom: 212 Total (i.e. Null);  212 Residual
Null Deviance:      290.8
Residual Deviance: 290.8        AIC: 292.8
> model_2

Call:  glm(formula = target ~ ., family = "binomial", data = training)

Coefficients:
(Intercept)          age          sex           cp     trestbps         chol
   3.421021     0.012978    -1.616742     1.190031    -0.009878    -0.004557
        fbs      restecg      thalach        exang      oldpeak        slope
  -0.856221     0.667052     0.008611    -0.843245    -0.519216     0.511259
         ca         thal
  -0.936730    -0.952283

Degrees of Freedom: 212 Total (i.e. Null);  199 Residual
Null Deviance:      290.8
Residual Deviance: 152.4        AIC: 180.4
```

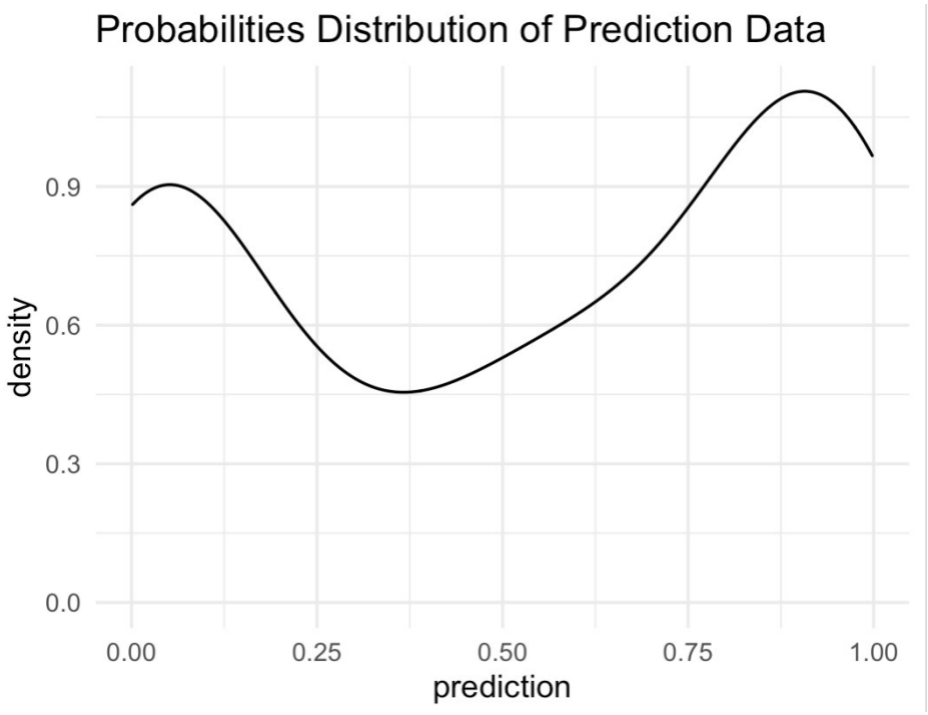
We have used the Stepwise method to create a better model Model_3 wherein we mixed both the model_1 and model_2 as there are only a few significant variables for our model. We found Sex, Cp, fbs, restecg, exang, oldpeak, cal, thal as our main significant effect towards our target variable.

```
> model_3

Call:  glm(formula = target ~ sex + cp + fbs + restecg + exang + oldpeak +
    ca + thal, family = "binomial", data = training)

Coefficients:
(Intercept)          sex           cp          fbs      restecg        exang
     3.8409      -1.3348       1.2121      -0.9646       0.7970      -0.9824
    oldpeak           ca         thal
    -0.7805      -0.8753      -1.0110

Degrees of Freedom: 212 Total (i.e. Null);  204 Residual
Null Deviance:      290.8
Residual Deviance: 155.7        AIC: 173.7
```

We can clearly see that the Residual Deviance has declined a lot compared to model_1 and model_2, and AIC has gradually declined which indicates a better fit model.

Prediction

We can visualize the probability distribution of our forecast data using ggplot().



We will try to make a prediction with our test data by utilizing a model from the stepwise method. We may get the probability value of our forecast by using the parameter type = "response." From our prediction we can clearly depict that the model is more inclined to the 1(not health).

Below is an extensive view of our prediction variable compared with the target variable for testing dataset

```
       target prediction
3  Not Health Not Health
6  Not Health     Health
8  Not Health Not Health
16 Not Health Not Health
21 Not Health     Health
```

Here is the overview comparison between our prediction data and the target variable of our test data

We then created a confusion matrix to identify the performance of our classification algorithm and check how good the model fits

```
Confusion Matrix and Statistics

            Reference
Prediction   Health Not Health
  Health         31          4
  Not Health     10         45

               Accuracy : 0.8444
                 95% CI : (0.7528, 0.9123)
    No Information Rate : 0.5444
    P-Value [Acc > NIR] : 1.629e-09

                  Kappa : 0.6826

 Mcnemar's Test P-Value : 0.1814

            Sensitivity : 0.9184
            Specificity : 0.7561
         Pos Pred Value : 0.8182
         Neg Pred Value : 0.8857
             Prevalence : 0.5444
         Detection Rate : 0.5000
   Detection Prevalence : 0.6111
      Balanced Accuracy : 0.8372

       'Positive' Class : Not Health
```

The results obtained are shown above which defines specificity, sensitivity and accuracy of the model. It is seen that the model was 83% accurate and successful in predicting our target variable. Wherein the ability to correctly predict for Not Health individuals is found to be 92% and to correctly predict the health individual is 75%.

The advantage of a neural network is that it is adaptive in nature. It learns from the information provided, i.e. trains itself from the data, which has a known outcome and optimizes its weights for a better prediction in situations with unknown outcome.

Neural Network:

Artificial neural networks (ANNs) and simulated neural networks (SNNs) are a subset of machine learning that are at the heart of deep learning methods.
Training data is used by neural networks to learn and increase their accuracy over time. However, once these learning algorithms have been fine-tuned for accuracy, they become formidable tools in computer science and artificial intelligence, allowing us to quickly classify and cluster data.
The data is divided for analysis when using the ANN (Artificial Neural Networks) approach for classification analysis. The first is the training data distribution, in which the researcher determines a chunk of 70 percent of the data for training and another 30 percent for testing. Modeling neural networks with training data and 5 hidden layers is used to yield the following model.

Initially the required packages for neural networks were installed such as “neuralnet”, “dplyr”, “caret” and the respective libraries were imported for the same:

Next, the min-max method was used to standardize the data in order to allow the scale of the data units for each variable be the same

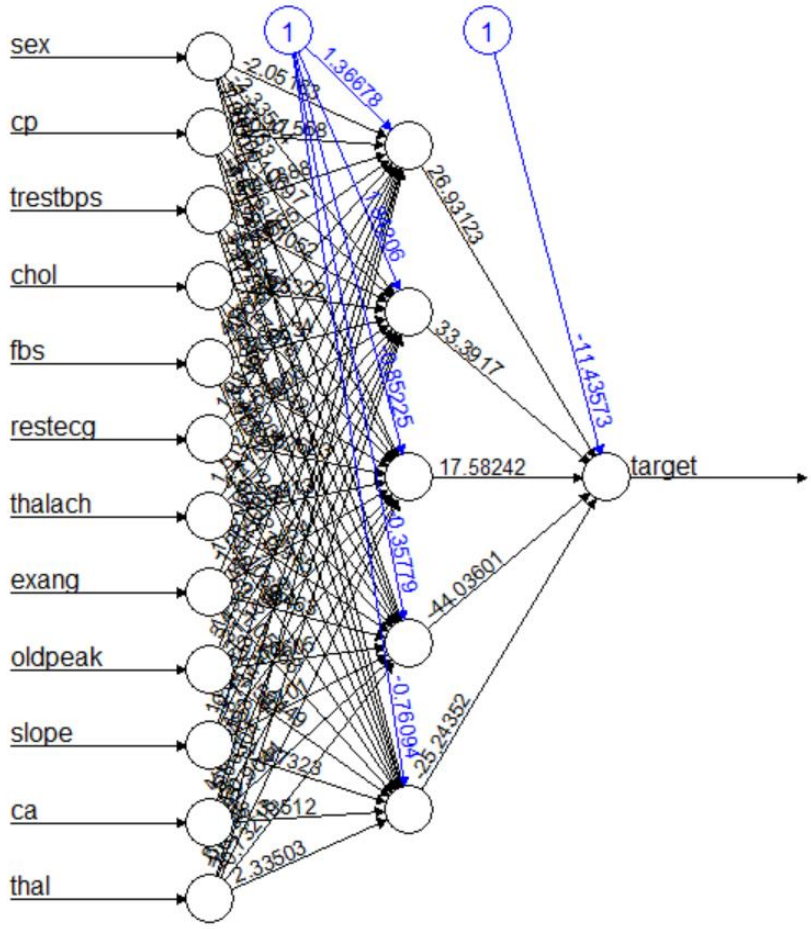
```
> for (i in names(data[,-1]))
+   data[i] <- (data[i] - min(data[i]))/(max(data[i]) - min(data[i]))
> data
   age sex        cp   trestbps       chol fbs restecg   thalach exang    oldpeak
1   63   1 1.0000000 0.48113208 0.24429224   1     0.0 0.6030534     0 0.37096774
2   37   1 0.6666667 0.33962264 0.28310502   0     0.5 0.8854962     0 0.56451613
3   41   0 0.3333333 0.33962264 0.17808219   0     0.0 0.7709924     0 0.22580645
4   56   1 0.3333333 0.24528302 0.25114155   0     0.5 0.8167939     0 0.12903226
5   57   0 0.0000000 0.24528302 0.52054795   0     0.5 0.7022901     1 0.09677419
6   57   1 0.0000000 0.43396226 0.15068493   0     0.5 0.5877863     0 0.06451613
7   56   0 0.3333333 0.43396226 0.38356164   0     0.0 0.6259542     0 0.20967742
8   44   1 0.3333333 0.24528302 0.31278539   0     0.5 0.7786260     0 0.00000000
9   52   1 0.6666667 0.73584906 0.16666667   1     0.5 0.6946565     0 0.08064516
10  57   1 0.6666667 0.52830189 0.09589041   0     0.5 0.7862595     0 0.25806452
11  54   1 0.0000000 0.43396226 0.25799087   0     0.5 0.6793893     0 0.19354839
12  48   0 0.6666667 0.33962264 0.34018265   0     0.5 0.5190840     0 0.03225806
13  49   1 0.3333333 0.33962264 0.31963470   0     0.5 0.7633588     0 0.09677419
14  64   1 1.0000000 0.15094340 0.19406393   0     0.0 0.5572519     1 0.29032258
15  58   0 1.0000000 0.52830189 0.35844749   1     0.0 0.6946565     0 0.16129032
```

Now, we divided the data into 2 sets, out of which 70% of the data was considered as training set which was used to train the model and 30% was the testing set which was used for testing the model.

```
> set.seed(123)
> ind <- sample(2, nrow(data), replace = TRUE, prob = c(0.7, 0.3))
> training <- data[ind==1,]
> testing <- data[ind==2,]
```

Next we worked on the training set for which we had to eliminate the target variable from the training set. We have set the activation function as logistic as our target variable was in the form of 1's and 0's. Also, the plot for the neural network is displayed below.

```
> set.seed(321)
> n <- neuralnet(target~.,
+                data = training[,-1],
+                hidden = 5,
+                act.fct = "logistic",
+                linear.output = FALSE)
> plot(n)
```



Now, we worked on the testing data for predicting the output.

```
> output <- compute(n, testing[,-1])
> head(output$net.result)
        [,1]
2  0.9996645
4  1.0000000
5  0.9999038
8  0.9254268
11 0.9942693
16 1.0000000
> head(training[1,])
  age sex cp  trestbps      chol fbs restecg   thalach exang   oldpeak slope ca
1  63   1  1 0.4811321 0.2442922   1       0 0.6030534     0 0.3709677     0  0
       thal target
1 0.3333333      1
```

The rounded results for observed and predicted output were generated.

```
> results <- data.frame(Data1=testing$target, Predicted=output$net.result)
> roundedresults <- sapply(results, round, digits=0)
> roundedresults
      Data1 Predicted
 [1,]     1         1
 [2,]     1         1
 [3,]     1         1
 [4,]     1         1
 [5,]     1         1
 [6,]     1         1
 [7,]     1         1
 [8,]     1         0
 [9,]     1         0
[10,]     1         1
[11,]     1         1
[12,]     1         0
[13,]     1         0
[14,]     1         1
[15,]     1         1
```

The model was then created by using the confusionMatrix to determine the accuracy and amount of quality of the analysis results.

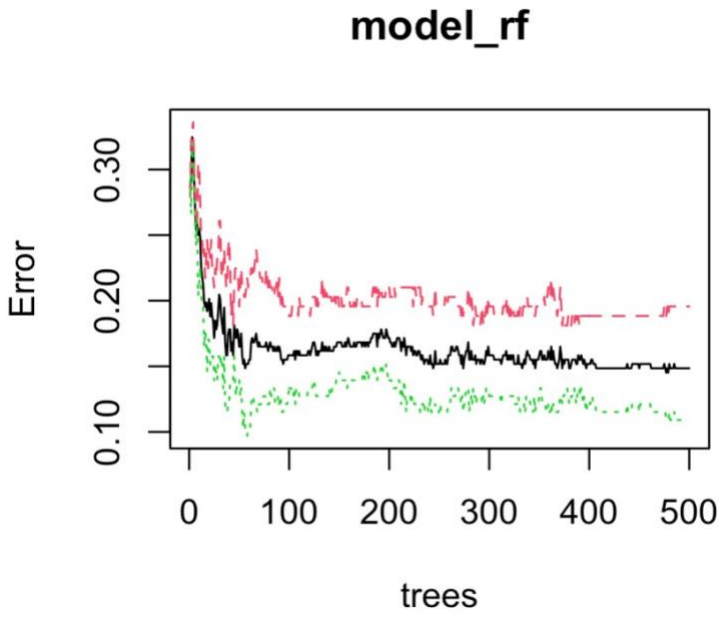
```
> actual <- testing$target
> prediction <- round(output$net.result, digits = 0)
> mtab <- table(actual, prediction)
> mtab
```

```
> confusionMatrix(mtab)
Confusion Matrix and Statistics

      prediction
actual  0  1
     0 29 12
     1  9 36

               Accuracy : 0.7558
                 95% CI : (0.6513, 0.842)
    No Information Rate : 0.5581
    P-Value [Acc > NIR] : 0.0001152

                  Kappa : 0.509

 Mcnemar's Test P-Value : 0.6625206

            Sensitivity : 0.7632
            Specificity : 0.7500
         Pos Pred Value : 0.7073
         Neg Pred Value : 0.8000
             Prevalence : 0.4419
         Detection Rate : 0.3372
   Detection Prevalence : 0.4767
      Balanced Accuracy : 0.7566

       'Positive' Class : 0
```

The model's specificity, sensitivity, and accuracy are defined by the findings displayed above. The model was 75 percent accurate and successful in forecasting our target variable, as can be seen.

Random Forest:

A Random Forest, as the name indicates, operates similarly to a Decision Tree in that it has a Root Node which in our case is considered as a 'target' variable that needs to be forecasted and then a collection of 'if-else' statements is used to analyze how different factors affect our prediction. It constructs and combines multiple decision trees in order to make more accurate predictions. It can be used in a regression model (with a continuous target variable), but it excels at classification models (i.e. categorical target variable).



The graphic above depicts the Error and Number of Trees obtained from our random forest study. We can plainly observe that when additional trees are added and averaged, the Error decreases drastically. In our training set, the red line reflects the MCR of the class identified as not having heart disease, while the green line represents the MCR of the class designated as having heart disease. Furthermore, the black line reflects the total MCR (Method comparison Regression) or OOB (out of bag) error. We're interested in the total mistake rate, which appears to be rather low. Interestingly, we notice a somewhat high Error for our class identified as not having heart disease over the first few trees, but this soon drops when additional trees are added and averaged.

```
Confusion Matrix and Statistics

            rf_pred
             Health Not Health
  Health        111         27
  Not Health     18        147

               Accuracy : 0.8515
                 95% CI : (0.8064, 0.8896)
    No Information Rate : 0.5743
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.699

 Mcnemar's Test P-Value : 0.233

            Sensitivity : 0.8605
            Specificity : 0.8448
         Pos Pred Value : 0.8043
         Neg Pred Value : 0.8909
             Prevalence : 0.4257
         Detection Rate : 0.3663
   Detection Prevalence : 0.4554
      Balanced Accuracy : 0.8526

       'Positive' Class : Health
```

We can observe from our resulting confusion matrix and evaluation that our accuracy is around 0.85. This is a fairly accurate figure. Although this isn't a poor accuracy rate, there have been 18 patients in our hold test set who did not yet have heart disease but were misclassified having heart disease by our model. Similarly, our model misclassified 27 individuals who had heart disease as not having heart disease. Since sensitivity in this situation is the percentage of patients who have heart disease and were properly forecasted to have heart disease, we can see how this result occurred 85 percent of the time based on our model outcome.

Conclusion

From the observations and calculations, we found that heart disease is one of society's key worries nowadays. Manually calculating the chances of developing heart disease based on risk factors is tough. Machine learning techniques like Linear Regression, Logistic Regression, Neural Networks and Random Forest were used to anticipate the outcome of existing data. We utilized various machine learning classifiers to predict cardiac disease, wherein Random Forest achieved an highest accuracy of 85 percent. As a result, a review of various existing therapies to heart disease might be beneficial in recommending the next treatment to individuals in order to preserve their lives.

In the future, we hope to utilize XGBoost to predict heart disease in youngsters and see if we can improve the accuracy. If characteristics are appropriately handled, considerable performance in the categorization of heart disease prediction will be achieved. The outputs of our proposed methodologies will serve as the benchmark performance results on heart disease in future investigations.

References:

[1] Rahman, R. (2021, March 22). Heart attack analysis & prediction dataset. Retrieved April 25, 2022, from https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack- analysis-prediction-dataset?resource=download

[2] Data622_FinalProject. (n.d.). Retrieved April 25, 2022, from https://rpubs.com/DaisyCai/Data622_FinalProject

[3] Ihl, P. (2020, December 15). Supervised ML - regression (I). Retrieved April 25, 2022, from https://www.startupengineer.io/_repos/dat_sci_2/07_ml_sup_i/

[4] Sign in. RPubs. (n.d.). Retrieved May 2, 2022, from https://rpubs.com/Arifyunan360/Heart_Disease_Prediction