MGSC 661 FINAL PROJECT

Prediction on Heart Diseases

YICHEN WANG

260761601

Introduction

As the pace of the modern life becomes increasingly more intense, heart disease is starting to affect not only the elderly, but the younger generation now as well. Heart disease is now the leading cause of death for people of most racial and ethnic groups around the world. According to statistics provided by CDC, there is one person dies every 36 seconds in the United States from cardiovascular disease, and it accounts for 1 in every 4 deaths in the US every year [1]. However, most of the cases could not be detected at an early stage, as about 1 in 5 heart attacks is silent, where the damage is done without the patient be aware of it. Therefore, how to analyze and predict the possibility of having a heart disease should raise attention to prevent millions of possible deaths globally, and data analysis and modelling could play a crucial role in this process.

This project is aiming to predict the occurrence of the heart disease based on different clinical features such as age, sex, blood pressure and so on based on a dataset provided on Kaggle, which includes over 11 common features, making it the largest heart disease dataset available so far for research purposes [2]. To do such prediction, data cleaning and feature selection will be implemented at first using both PCA analysis and feature importance from random forest to filter out certain less-contributing predictors. After that, both regression modelling and boosted forest modelling will be implemented to compare their accuracies. By constructing and training a reliable model, this project could help the researchers to identify the potential threat from heart disease at an early stage and keep threats from potential patients.

Data Description

In this dataset, 11 clinical features for predicting heart disease have been included and these features are explained in the table 1.1, shown in the Appendix. Most of these features are not the common features such as race, weight, or something general, but are measured with accurate medical apparatus. As a result, the prediction based on professional features could be more reliable and self-explanatory [3].

- Data visualization

To first explore the data, the histograms and bar plots for numerical and categorical variables are plotted using ggplot(), included in the Appendix. For numerical variables such as age, RestingBP, Cholesterol and MaxHR, they all have an approximate normal distributed data. However, the Oldpeak shows a large skewness, peaking at 0. This is understandable as Oldpeak represents numerical value measured for ST segment, which is normally 0 for most people.

For the categorical features, the output HeartDisease has a relatively similar number between observations with heart disease and those without. However, other categorical features such as gender and FastingBS (blood sugar level) has a huge difference in number within their categories, which is interesting as it might be an indication of low collinearity between these predictors and our target variable.

- Feature Engineering & Collinearity check

As the dataset available is rather neat, there is no need for data cleaning and such procedures. For the ease of modelling in the later stage, all the categorical variables needed to be dummified using the "fastDummies" library. For this dataset, "Sex", "ChestPainType", "RestingECG", "ExerciseAngina" and "ST_Slope" are dummified.

The next step is to explore the relationships between each variable. By populating a collinearity matrix, we could easily identify the degree of collinearity between each variable after dummification. The matrix is shown below as Fig. 3.1

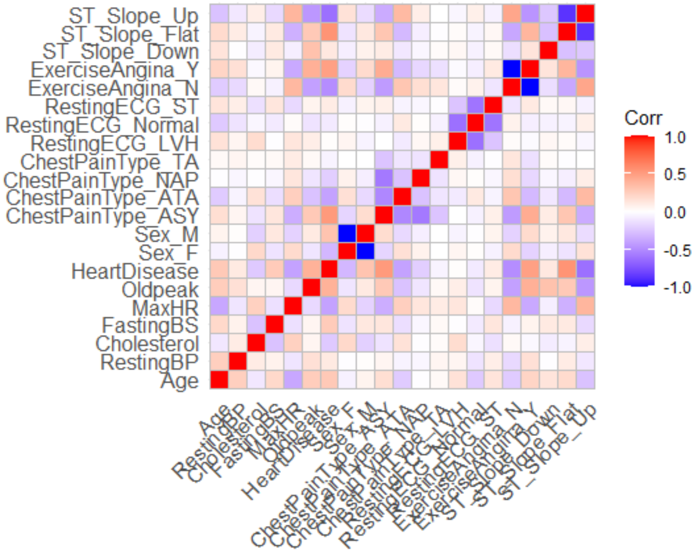


Fig. 3.1. Collinearity Matrix of the dataset

From the correlation matrix it is easy to tell that there are some predictors have a relatively high correlation with others. For certain pairs such as Sex_M /Sex_F and ExerciseAngina_Y/ ExerciseAngina_N, they are supposed to be a boolean type variable, so being negatively correlated to each other is understandable. In this case, Sex_F and ExerciseAngina_N will be dropped to avoid collinearity problem. Another pair showing collinearity problem is ST_Slope_Up/ ST_Slope_Flat, as they may have special relationship in cardiovascular research, so we will only keep ST_Slope_Flat.

Model Selection & Methodology

Before building a model for our dataset, feature selection is required in order to find out the appropriate predictors. To ensure the accuracy of the selection result, two of the techniques will be used to validate each other: Feature importance using random forest model, and the 2-dimensional feature plot of PCA.

For the feature importance method, a random forest model that takes account of all the variables is created and then is analyzed using varImpPlot() function to visualize the ranking of the importance of each variable to the model. The result is shown in Fig. 4.1 below:

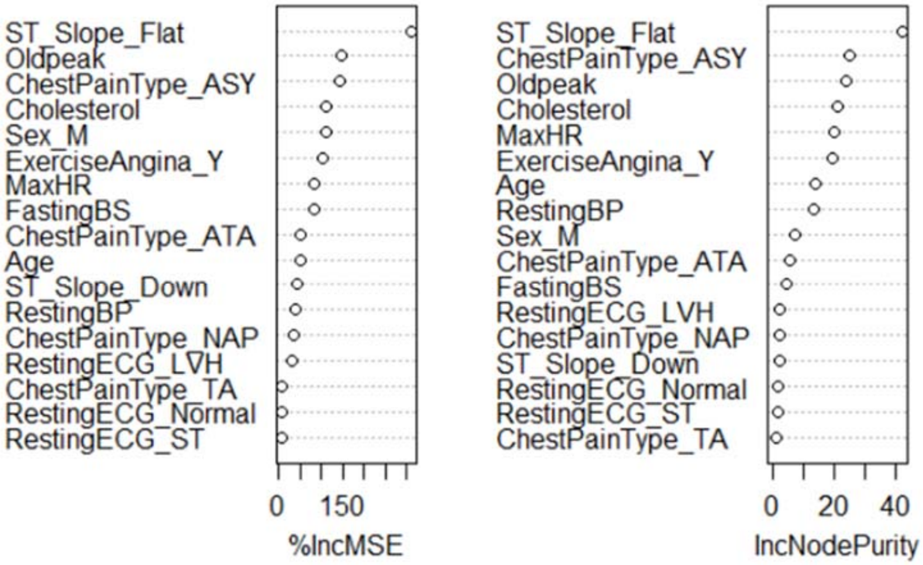


Fig. 4.1. Feature importance plot for the dataset.

For these two plots, the %IncMSE means the potential percentage increase in the MSE if the current chosen feature is not included. The IncNodePurity is a measure of variable importance based on the Gini impurity index, where the higher value corresponds to a higher importance of the variable. Both plots indicate that ST_Slope_Flat is a very important predictor to the model. Without it, a 300% increase in MSE would result in the model. Other predictors such as Oldpeak, ChestPainType_ASY, Cholesterol and ExerciseAngina_Y are in the top of the importance for both plots. These features should be considered in our future model.

The other method for validating our choice of predictors is using the 2-dimensional plot of PCA. With labels set to be HeartDesease and variables set to be our entire dataset, we could use autoplot() function to plot a 2-dimentional plot by taking consider of only 2 principal components of the dataset.The result is shown in the following plot Fig. 4.2.

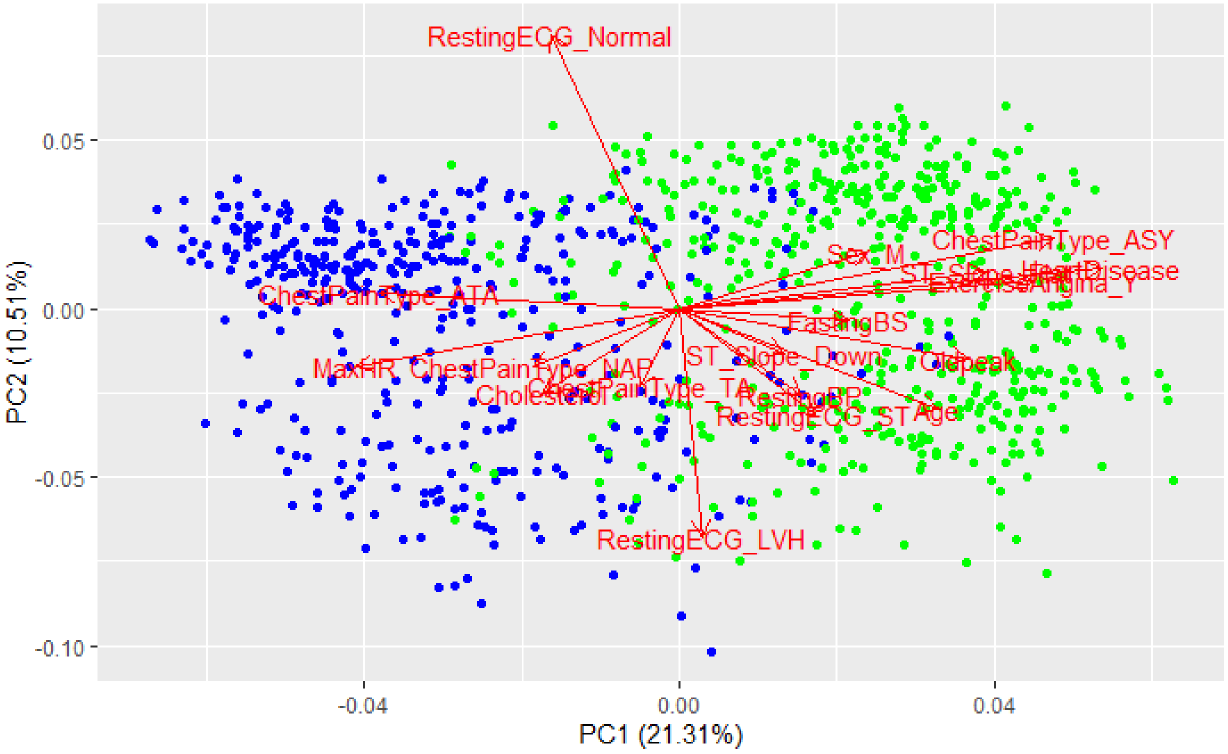
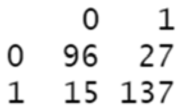


Fig.4.2 2-dimentional plot of PCA analysis

The green dots are those with HeartDisease and the blue dots are those without. When arrows are closer together and in the same direction, the variables will be highly correlated. From the plot we could tell that ChestPainType_ASY, ExerciseAngina_Y, ST_Slope_Flat, FastingBS and Oldpeak are highly correlated to our target variable HeartDisease, where those features being perpendicular to HeartDisease is nearly not correlated at all. This result is in line with the choice provided by the feature importance method, showing the reliability of both methods. Based on both methods, the final choice of features is [ChestPainType_ASY, FastingBS , ST_Slope_Flat, Oldpeak, Cholesterol, ExerciseAngina_Y]. For model selection, there are still two different models chosen: logistic regression and boosted tree, and their accuracy and mse will be compared to see which one excels.

Results

For the regular logistic regression model, we have the data being split into training and testing data at a SplitRatio of 0.7. With the features selected previously, an accuracy of 0.86118 and a mse of 0.138 is achieved, which are fairly acceptable. The table is shown below as Fig. 4.3.

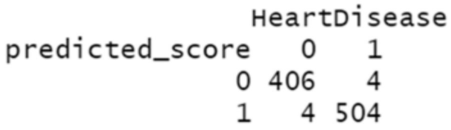


```
     0   1
0   96  27
1   15 137
```

Fig.4.3. classification outcome table

By using these data, both precision (137/(27+137) = 0.83) and recall (137/(15+137) = 0.90) could be calculated and an overall F1 score of 0.86 is very satisfying (F1 = 2 * (precision * recall) / (precision + recall)).

For boosted forest model, same features are selected, with the distribution type to be "Bernoulli" as we are doing a classification task. The predicted score generated are numbers between 0 and 1, so any values greater than 0.5 will be reckoned as 1 and 0 otherwise. By printing out the result (shown in fig. 4.4), the accuracy is 0.991 and mse = 0.0087, which is extremely high. The F1 score is as high as 0.992, which is much higher compared to the logical regression model.



```
                 HeartDisease
predicted_score   0   1
              0 406   4
              1   4 504
```

Fig.4.4. classification outcome table for boosted forest

The main improvement is due to the boosting, as each tree grows sequentially, and learns from the previous one. This mechanism helps to build a forest of weak predictors. However, this model may tend to be a bit overfitting over this particular dataset. But the good performance provided by both models indicates the reliability of the features chosen previously.

Predictions & Conclusions

From the feature selection results generated by the feature importance of random forest and supported by the reliable accuracy from predictions of both models, we can conclude that the top five features that could be used for potential heart disease detection in physical examinations: ChestPainType_ASY, FastingBS , ST_Slope_Flat, Oldpeak, Cholesterol and ExerciseAngina_Y, as explained in the appendix. These factors show high feature importance to the prediction model, meaning they are closely correlated to the occurrence of the heart attack and can be used as indicators for heart disease, with a high 0.992 F1 score justifying its reliability.

The conclusion could help improve the efficiency of current diagnosis of heart disease, which includes mainly three parts: first category is history check, asking about whether having heart disease history or other symptoms in the past; the second category examines the heart rate abnormality, sound of possible lung crackles, wheezing or third heart sound, while the third category will check the chest radiography. There are over 20 different tests for detecting heart disease, which may lead to higher expenses and long time of diagnosis [4]. However, the feature importance table and PCA plot help narrow the scope of these tests and assure a highly accurate result with only 5 tests: 2 in category 2 (ChestPainType_ASY and ExerciseAngina_Y), 2 in category 2 (FastingBS and Cholesterol) and 1 in category 3 (ST_Slope_Flat). Not only the cost of physical examination for detecting heart disease is reduced, but the efficiency improves as well, which could encourage those who were previously discouraged by large medical bill and low efficiency, to take such examinations in order to save millions of potential lives. With two different standards of heart disease testing: one with higher price and accuracy and one with efficiency and lower cost, more potential patients with different income levels could be benefited from these early alerts. And I believe such strategy will not only benefit consumers but will improve the revenue of the hospital as well, creating a win-win situation.

Reference

[1] Heart Disease Facts | cdc.gov. (2021). Retrieved 12 December 2021, from https://www.cdc.gov/heartdisease/facts.html

[2] Heart Disease Facts | cdc.gov. (2021). Retrieved 12 December 2021, from https://www.cdc.gov/heartdisease/facts.html

[3] Heart Failure Prediction. (2021). Retrieved 17 December 2021, from https://www.kaggle.com/fedesoriano/heart-failure-prediction

[4] Shamsham, F., & Mitchell, J. (2021). Essentials of the Diagnosis of Heart Failure. Retrieved 17 December 2021, from https://www.aafp.org/afp/2000/0301/p1319.html

#

#

#

#

#

#

#

#

#

#

#

#

#

Appendix

Age	age of the patient [years]
Sex	sex of the patient [M: Male, F: Female]
ChestPainType	chest pain type [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic]
RestingBP	resting blood pressure [mm Hg]
Cholesterol	serum cholesterol [mm/dl]
FastingBS	fasting blood sugar [1: if FastingBS > 120 mg/dl, 0: otherwise]
RestingECG	resting electrocardiogram results [Normal: Normal, ST: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV), LVH: showing probable or definite left ventricular hypertrophy by Estes' criteria]
MaxHR	maximum heart rate achieved [Numeric value between 60 and 202]
ExerciseAngina	exercise-induced angina [Y: Yes, N: No]
Oldpeak	oldpeak = ST [Numeric value measured in depression]
ST_Slope	the slope of the peak exercise ST segment [Up: upsloping, Flat: flat, Down: downsloping]
HeartDisease	output class [1: heart disease, 0: Normal]

Table 1.1. Feature explanation for Heart Disease dataset

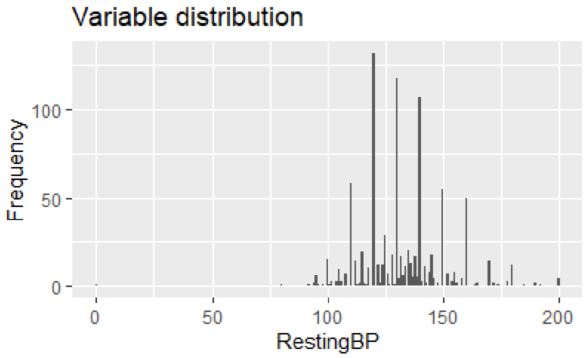


Fig 2.1. RestingBP Histogram

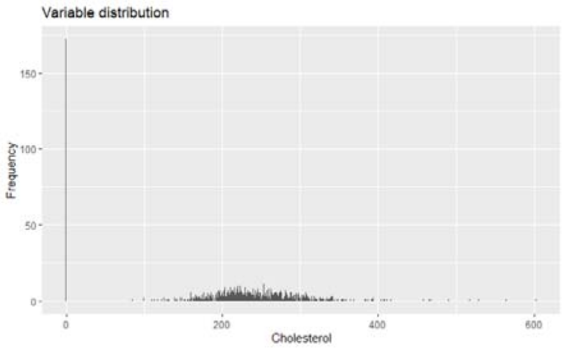


Fig 2.2. Cholesterol Histogram

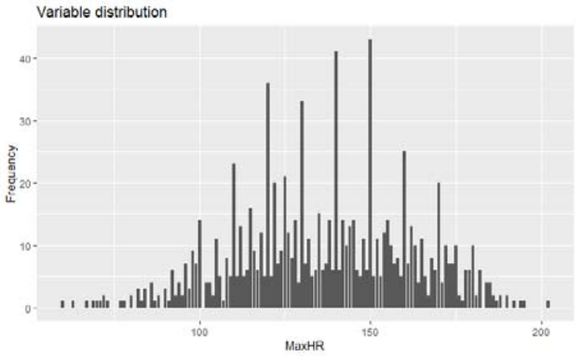


Fig 2.3. MaxHR Histogram

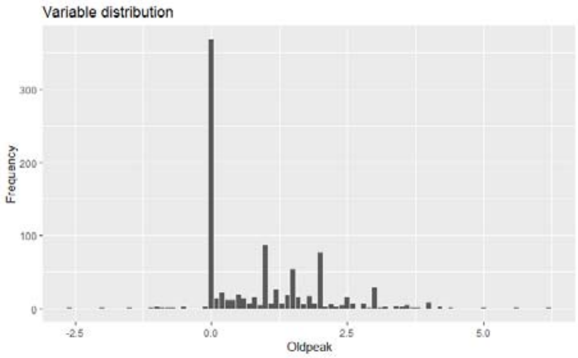


Fig 2.4. Oldpeak Histogram

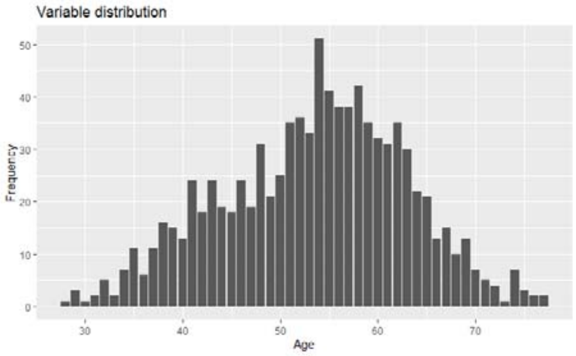


Fig 2.5. Age Histogram

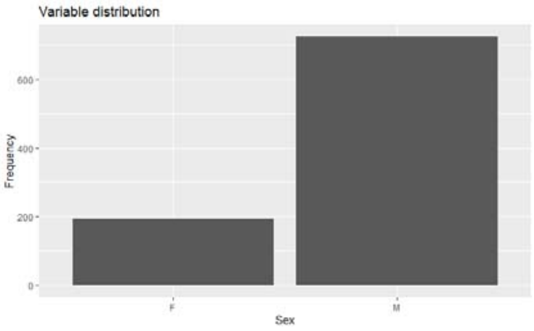


Fig 2.6. Sex Barplot

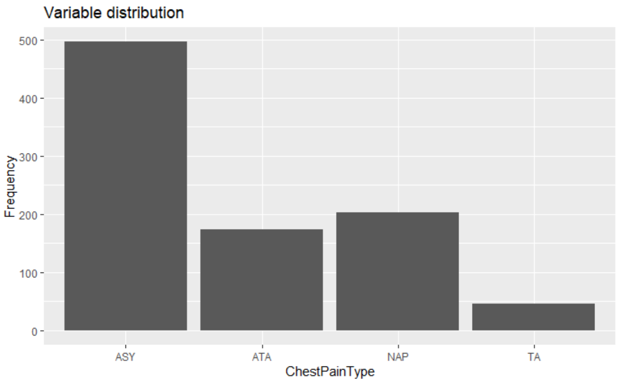


Fig 2.7. ChestPainType Barplot

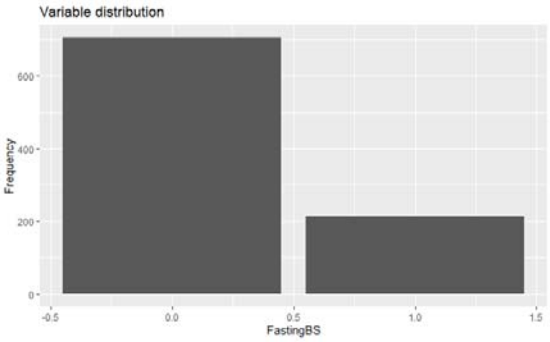


Fig 2.8. FastingBS Barplot

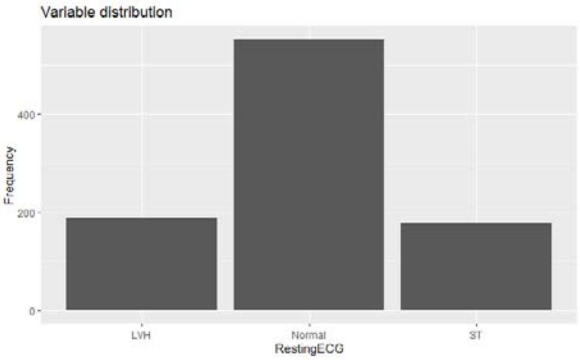


Fig 2.9. RestingECG Barplot

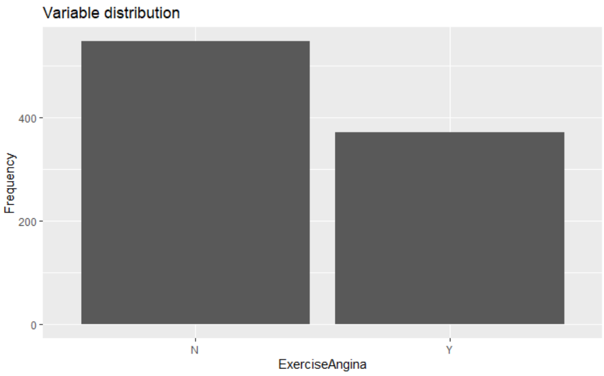


Fig 2.10. ExerciseAngina Barplot

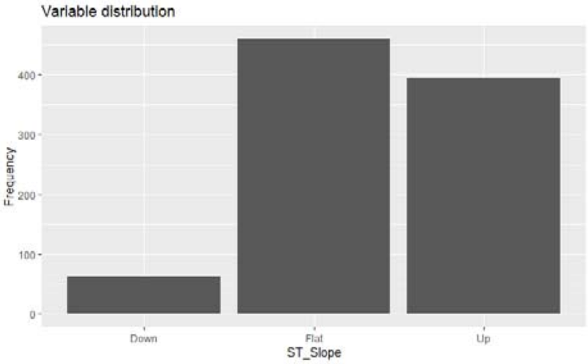


Fig 2.11. ST_Slope

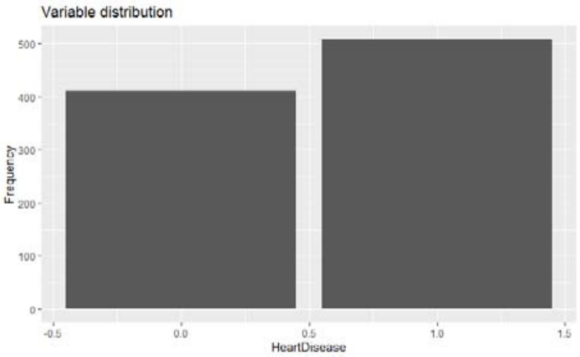


Fig 2.12. HeartDisease Barplot

Code (.rmd format)

~~~
---
title: "Final_project"
author: "YICHEN"
date: "10/12/2021"
output: html_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```

```{r load in data}
df <- read.csv("C:/YICHEN WANG/Course/2021 FALL/MGSC661/final project/heart.csv")
attach(df)
```

```{r data exploration}
library(ggplot2)
hist_fn = function(varname) {
  ggplot(data = df,
        aes(x = {{varname}})) +
    geom_histogram(stat = 'count') +
    labs(title = "Variable distribution",
        y = "Frequency")
}
hist_fn(RestingBP)
hist_fn(Cholesterol)
hist_fn(MaxHR)
hist_fn(Oldpeak)
hist_fn(Age)
hist_fn(Sex)
~~~

~~~
hist_fn(ChestPainType)
hist_fn(FastingBS)
hist_fn(RestingECG)
hist_fn(ExerciseAngina)
hist_fn(ST_Slope)
hist_fn(HeartDisease)

```

```{r dummify}
#install.packages('fastDummies')
library(fastDummies)

df <- dummy_cols(df, select_columns = "Sex")
df <- dummy_cols(df, select_columns = "ChestPainType")
df <- dummy_cols(df, select_columns = "RestingECG")
df <- dummy_cols(df, select_columns = "ExerciseAngina")
df <- dummy_cols(df, select_columns = "ST_Slope")
```

```{r data exploration}
library(dplyr)
df <- select_if(df, is.numeric)
#View(quantvars)
# populating correlation matrix
corr_matrix = cor(df)
corr_matrix <- round(corr_matrix, 2)

# visualize correlation
install.packages("ggcorrplot")
library(ggcorrplot)
ggcorrplot(corr_matrix)
```
~~~

~~~
```{r drop columns}
df = subset(df, select = -c(ST_Slope_Up,Sex_F,ExerciseAngina_N) )
```

```{r feature importance, fig.width = 6, fig.asp = 0.8}
#feature importance
#install.packages("randomForest")
library(randomForest)
rfImp <- randomForest(HeartDisease ~ ., data = df,
                ntree = 10000,
                importance = TRUE)
importance(rfImp)
varImpPlot(rfImp)
```

```{r view pca}
labels = df[,c(7)]
vars = df[,c(1:18)]
pca = prcomp(vars,scale=TRUE)
```

```{r plot the 2d with autoplot()}
#install.packages("ggfortify")
library(ggfortify)

autoplot(pca,data=vars,loadings=TRUE,
      col = ifelse(labels==1,"green","blue"),loadings.label=TRUE)
```


### 4.
```{r percentage of variance}
pve = (pca$sdev^2)/sum(pca$sdev^2)
~~~

~~~
par(mfrow=c(1,2))
plot(pve,ylim=c(0,1))
plot(cumsum(pve),ylim=c(0,1))

```
```{r data split}
require(caTools)
require(methods)
sample = sample.split(df$HeartDisease,SplitRatio=0.7)
train = subset(df,sample==TRUE)
test=subset(df,sample==FALSE)
```


```{r regression}
fit = glm(HeartDisease ~ ChestPainType_ASY +FastingBS
+ST_Slope_Flat+Oldpeak+Cholesterol+ExerciseAngina_Y, data = train, family = "binomial")


test$pred <- predict(fit,test)
test$pred <- ifelse(test$pred > 0, 1,0)
table(test$HeartDisease,test$pred)
accuracy = sum(ifelse(test$pred ==  test$HeartDisease, 1, 0))/ length(test$pred)
mse = mean((test$pred-test$HeartDisease)^2)
accuracy
mse
```


```{r boosted forest}
library(gbm)
set.seed(1)


boosted = gbm(HeartDisease ~ ChestPainType_ASY +FastingBS
+ST_Slope_Flat+Oldpeak+Cholesterol+MaxHR,data=df,distribution = "bernoulli",n.trees = 1000,
interaction.depth=4)
```
~~~

~~~
```{r prediction accuracy comparison}
predicted_score=predict(boosted, newdata=df, n.trees=1000, type="response")
predicted_score = ifelse(predicted_score>0.5, 1, 0)
table(predicted_score,HeartDisease)
accuracy = sum(ifelse(predicted_score ==  df$HeartDisease, 1, 0))/ length(predicted_score)
mse = mean((predicted_score-HeartDisease)^2)
accuracy
mse
```
~~~