From Data to Diagnosis: Predicting Heart Disease Severity with Machine Learning

Horace Tsai

Contents

1 Abstract

Looking at various models, Random Forest, Classification Trees, Linear Discriminant Analysis (LDA), and Neural Nets and comparing their accuracy in predicting the presence of heart disease in a patient. The models were derived from patients undergoing angiogrpahy at the Cleveland Clinic in Cleveland, Ohio. The response was measured as a numerical scale from $(0 - 4)$, 0 being no presence and 4 being heart disease being severely present in said patient. Leaving the response as is, the models performed at approximately 58 % accurate. Grouping the response scale into *None*, *Mild*, *Severe*. The random forest performed the best. Where the classification tree performed slightly better than the classification tree at the numerical scale. Changing the response to a binary response, our models were more accurate with one off that the neural net performed the worst at approximately 32 % accuracy. The linear discriminant model with the binary response performed just as well as the random forest that was grouped. However, recommending a model to use to predict heart disease presence, the random forest grouped model would be the one to use. This model provides adequately accurate predictions without losing too much information in the response.

2 Introduction

Heart disease, known as cardiovascular disease, refers to a class of disorders that affect the heart and blood vessels, potentially leading to conditions such as coronary artery disease, heart failure, and arrhythmias. Per the Centers for Disease Control and Prevention (CDC), heart disease is the leading cause of death for men and women in the United States. Back in 2021, 1 in every 5 deaths is caused by heart disease. Key factors that lead to heart disease are:

- High Cholesterol
- High Blood Pressure
- Diabetes
- Weight
- Family History
- History of Smoking

However, many factors for heart disease are modifiable through lifestyle changes. Having a healthy diet, regular exercise, avoiding tobacco, and managing stress can all substantially reduce one's risk of developing heart disease. Bridging the gap between clinical studies and statistical analysis, we will provide a comprehensive analysis on a patients risk of heart disease based on their medical history and conditions.

3 Project Goals

The goals of this project is to analyze the given features of patients in our data set and help correctly predict the severity of the presence of heart disease in patients. In order to do so, we will implement various types of models to help us reach our goal.

Severity of the presence of heart disease is measured on a numerical scale of $0 - 4$

- 0 being no indication of heart disease
- 4 being heart disease is severely present

4 Literature Review

There is a substantial amount of literature and research done in identifying heart disease and predicting in patients.

In *International Application of New Probability Algorithm for the Diagnosis of Coronary Artery Disease* written Back in August 1989 by R. Detrano, A. Janosi, W. Steinbrunn, M. Pfisterer, J. J. Schmid, S. Sandhu, K. H. Guppy, and S. Lee, V. Froelicher were able to use various Bayesian algorithms to predict the probability of disease in a patient. Their findings were that their algorithms overpredicted the probability of disease. Meaning that their algorithms over-estimated probabilities that were higher than the actual observed. However, their research and analysis showed that coronary disease probabilities derived from their algorithms were reliable and useful when applied to patients experiencing coronary disease symptoms.

In another paper, *Comparison of Logistic Regression and Bayesian Based Algorithms to Estimate Posttest Probability in Patients with Suspected Coronary Artery Disease Undergoing Exercise ECG* posted backed in April 1992 written by A. P. Morise, R. D. Duval, R. Detrano, M. Boibbio, and G. A. Diamond focused on compared two different models, logistic regression and Bayesian analysis and compared the performance of each model. This study looked at patients who underwent exercise testing within 3 months prior to coronary angiography which is a procedure that uses a special dye and x-rays to see how blood flows through the arteries in your heart. The conclusion of their study showed that logistic regression had a better discrimination than the Bayesian analysis for estimating the probability of coronary disease following exercise electrocardiography (ecg).

These were just two pieces of numerous pieces of literature that relate statistical analysis and heart disease together.

5 Data Features and Wrangling

Our dataset comes from the UC Irvine (UCI) machine learning repository. This dataset was donated back in June of 1988 consisting of 76 different features and 920 rows of data. Each row of data corresponds to a unique patient. However, all published experiments and literature refer to a subset of 14 of the 76 features and primarily focus on data originating from the Cleveland Clinic. Our modeling and analysis will primarily focus on the 14 features and in particular the data that was observed from the Cleveland database. We will now go over and explain the features of the dataset.

5.1 Features

5.1.1 ID

ID refers to a unique patient ID for the dataset

5.1.2 Age

Age refers to the patient age in years

5.1.3 Dataset

Dataset refers to the origin of the study. There are 4 different datasets:

- Cleveland
- Hungary
- VA Long Beach
- Switzerland

5.1.4 Sex

Sex refers to the sex of the patient:

- Male
- Female

5.1.5 CP

CP refers to chest pain type of the patient. There are 4 different types:

- Typical Angina
- Atypical Angina
- Non- Anginal
- Asymptomatic

Angina is a type of chest pain caused by reduced flow flow to the heart.

5.1.6 Trestbps

Trestbps refers to the resting blood pressure measured in mmHG

5.1.7 Chol

Chol refers to the serum cholesterol in mg/dl

5.1.8 Fbs

Fbs refers to whether the fasting blood sugar is greater than 120 mg/dl. This is a binary feature measured by:

- True
- False

5.1.9 Restecg

Restecg refers to resting electrocardiographic results. There are 3 different types:

- Normal
- ST abnormality
- LV Hypertrophy

5.1.10 Talach

Talach refers to the maximum heart rate achieved.

5.1.11 Exang

Exang refers to whether exercise induced angina. This is a binary feature measured by:

- True
- False

5.1.12 Oldpeak

Oldpeak refers to the ST depression induced by exercise relative rest.

5.1.13 Slope

Slope refers to the slope of the peak exercise in the ST segment. There are 3 different types:

- downsloping
- upsloping
- flat

5.1.14 ca

ca refers to the coronary artery/ number of vessels (0-3) colored by fluroscopy.

5.1.15 Thal

Thal refers to thalassemia. There are 3 different types:

- Normal
- Fixed defect
- Reversible defect

5.1.16 Num

Num refers to the severity presence of heart disease and is also our response feature. This is a measured on a scale from $(0 - 4)$ where 0 has no presence and 4 has a severely high presence of heart disease.

5.2 Data Wrangling

As mentioned previously, we will be using data that originated from the Cleveland database. This is the most complete subset of the dataset compared to the other databases where they have a much higher amount of missing data. In terms of our Cleveland subset data, we did have to omit a few rows of data due to the some of the missing values. The rows we had to omit were less than 5.

Our response feature, *num*, we had to manipulate that column. We grouped the numbers by the following:

- 0 to “none” (no presence of heart disease)
- $1 - 2$ to “mild” (mildly presence of heart disease)
- $3 - 4$ to “severe (severely high presence of heart disease)

We also later made *num* as a binary response feature,

- 0 if no presence of heart disease
- 1 if any signs of heart disease

6 Exploratory Data Analysis (EDA)

In this section, we are going to present some of the exploratory data analysis and our findings.

6.1 Number of Patients in Each Presence Level

Table 1: Number of Patients by Heart Disease Severity

	x
0	165
1	55
2	36
3	35
4	13

Table 1 shows us the number of people in each severity category. With the most patients 165, belonging to 0 no signs of heart disease and 13 patients are shown to have high presence of heart disease (4).

6.2 Average Patient Age

Table 2: Mean Age

sex	grp.mean
Female	55.72165
Male	53.71014

Here Table 2. shows us the average age for females and males.

6.3 Age Histogram

Figure 1 shows the age distribution by gender along with a dashed line showing the mean age. From the figure, there are few outliers for both genders older than 75 and younger than $30. It seems like a majority of the patient's ages are concentrated in the 40-70 range which makes sense since around that age group is where one would start to see signs of heart disease.

6.4 Serverity of Heart Disease by Gender

```
## Margins computed over dimensions
## in the following order:
## 1: number
## 2: male
```

Figure 2 shows a barplot of the severity of heart disease grouped by gender. By just glancing at each bar, one can see that males are more likely to have presence of heart disease than females. This assumption falls in line with what is generally said, males are more likely to have heart disease than females.

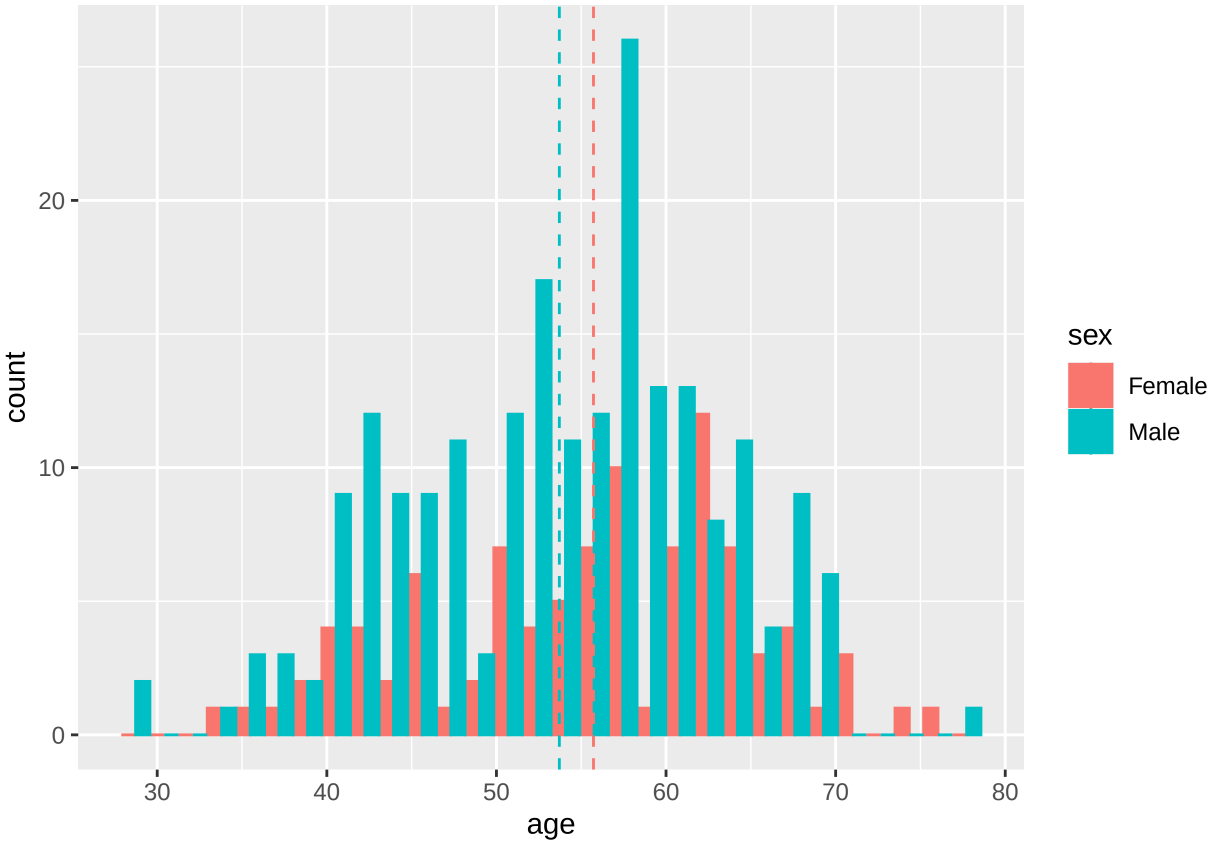

Figure 1: Age Histogram

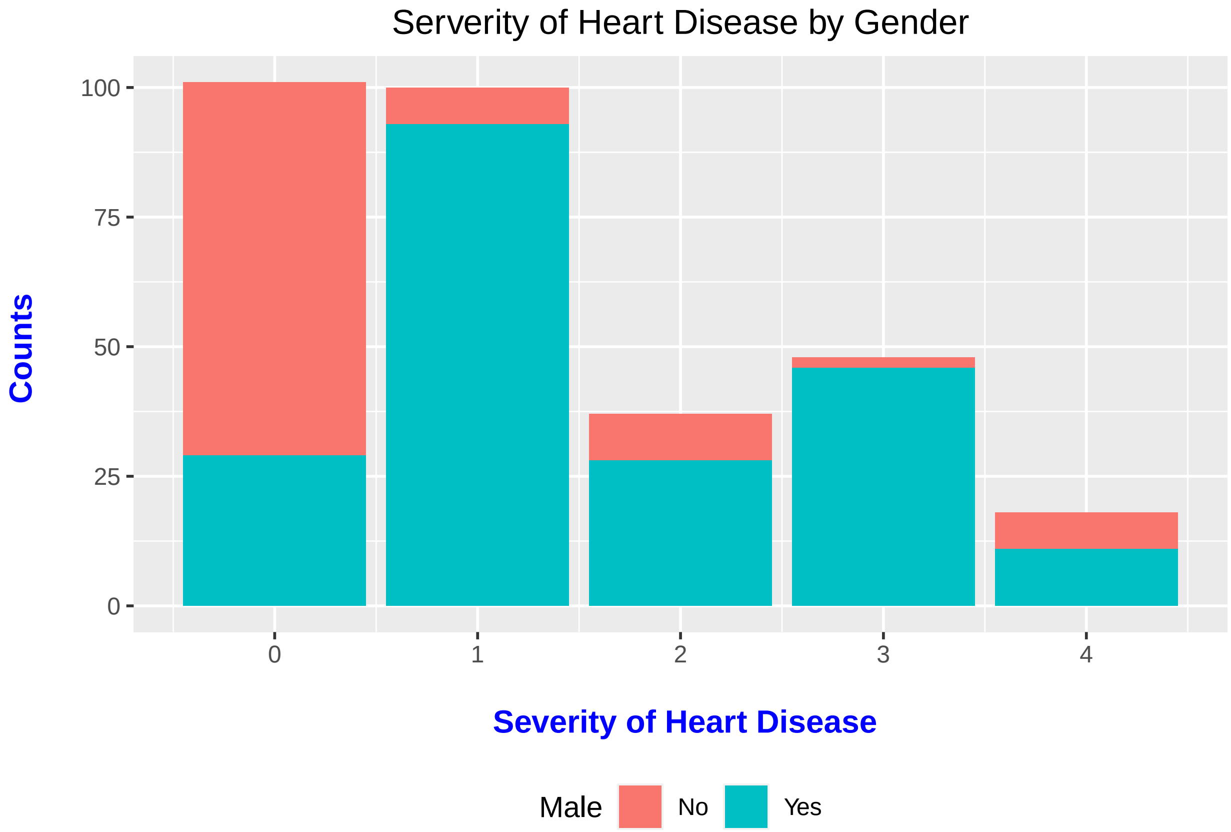

Figure 2: Serverity of Heart Disease by Gender

6.5 Distribution of Cholestorol Levels by Severity

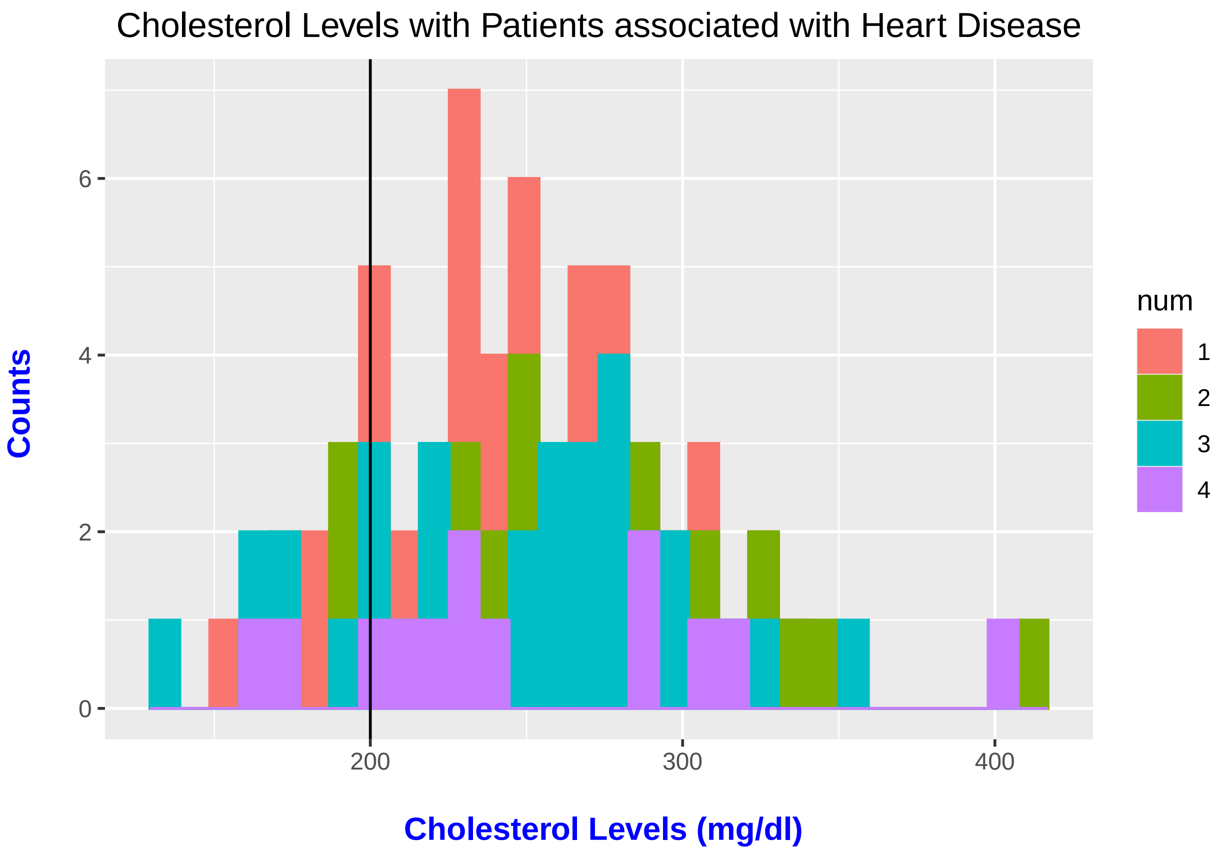

Figure 3: Cholesterol Levels with Patients associated with Heart Disease

Figure 3 shows the distribution of cholesterol levels of the patients that have presence of heart disease grouped by severity. Along with a black line that indicates the cutoff for *high* cholesterol (200 mg/dl). As you can tell, a majority of the patients that are to the right of the black line do show signs of heart disease. This is not surprising as high cholesterol are some of the factors associated with heart disease.

6.6 Distribution of Blood Pressure by Severity

Figure 4 shows the distribution of blood pressure of the patients that have presence of heart disease grouped by severity. Along with a black line that indicates the cutoff for *high* blood pressure(90 mmHG) As you can tell, almost all of the patients that are to the right of the black line do show signs of heart disease. This is not surprising as high blood pressure are some of the factors associated with heart disease. This shows that blood pressure is in fact a key factor associated with heart disease.

6.7 Thalassemia Across All Patients

Figure 5 shows the count of thalassemia across patients group by heart disease presence. We do see 1 missing value in this case, hence the "NA" on the plot. We see a majority of the patients being associated with

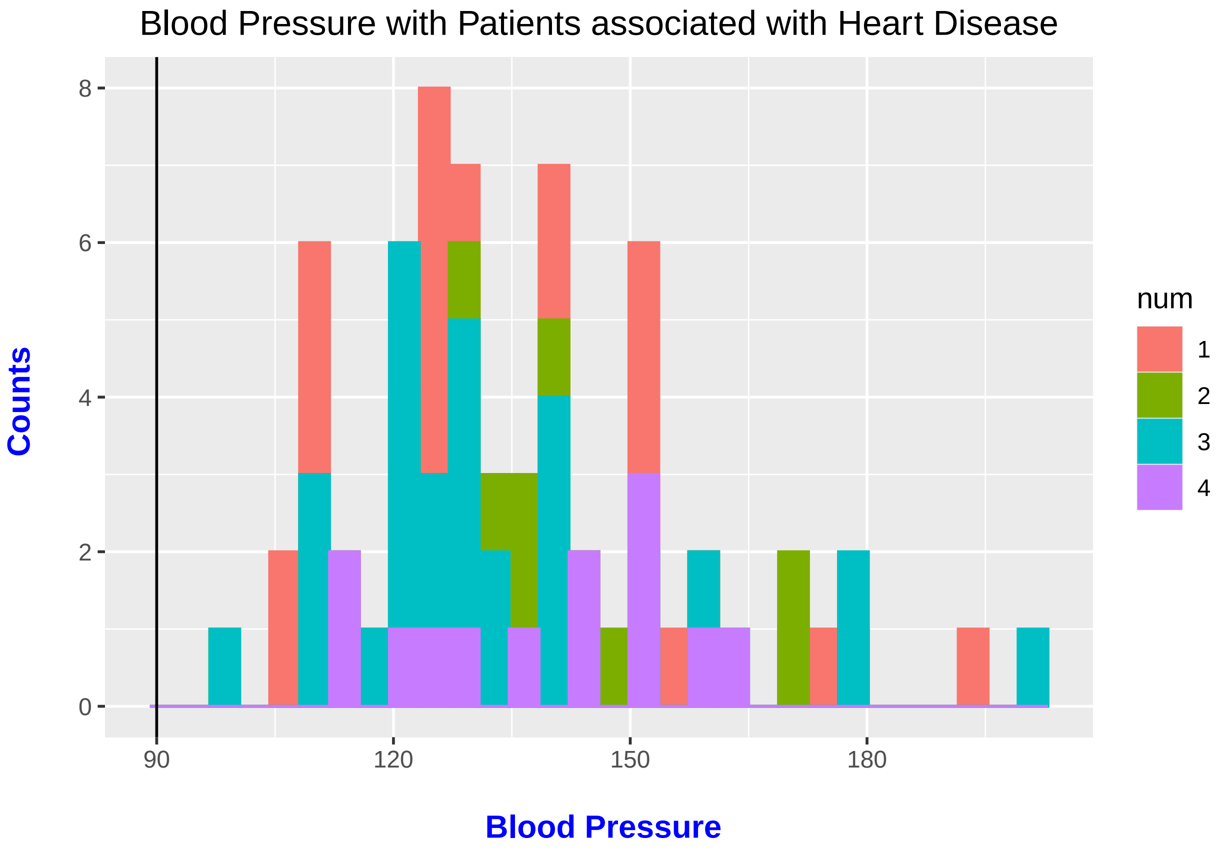

Figure 4: Blood Pressure with Patients associated with Heart Disease

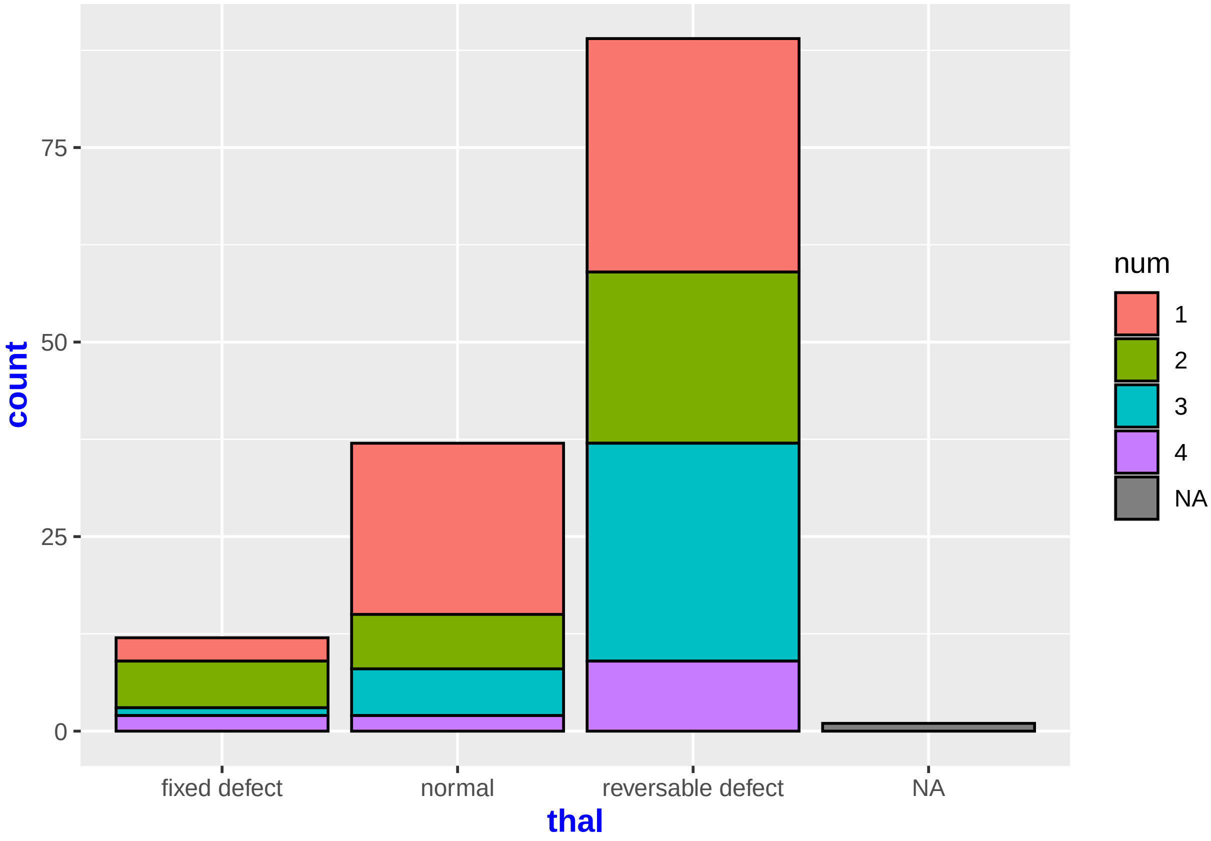

Figure 5: Barplot of Thalassemia

“reversible defect” meaning they are being treated for thalassemia. Thalassemia causes chronic anemia (low iron) and in order to treat this, patients are given blood transfusions. Due to the transfusions being given, patients typically have hemochromatosis (disease with iron overload). This results in the hardening of heart muscles therefore resulting in heart disease. Hence why we do see the very high number of “reversible defect” in our data.

6.8 Patients with Typical Angina

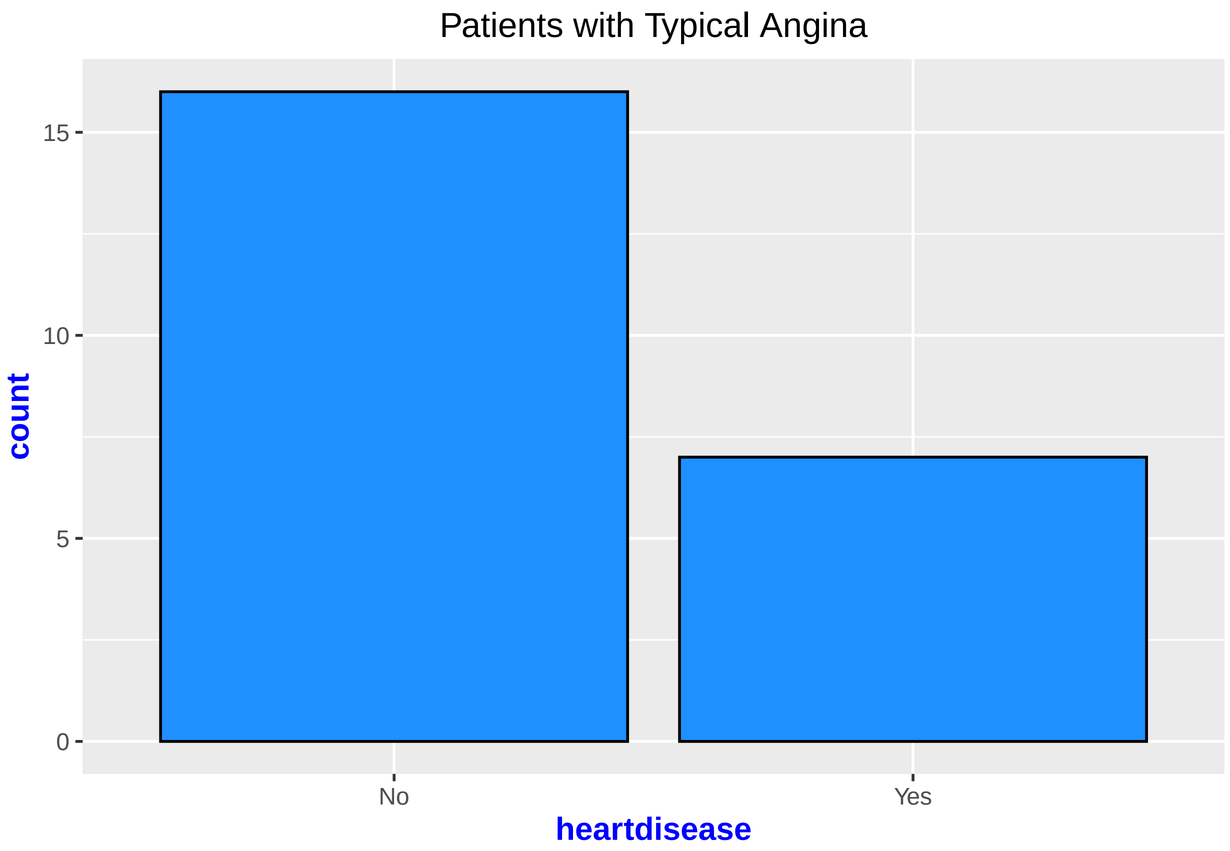

Figure 6: Barplot of Typical Angina

Figure 6 shows the patients with typical angina and whether that have heart disease or not. One might think this bar plot was a bit surprising since typical angina is chest pain. However, typical angina is present when a patient is having a heart attack.

6.9 Patients with Atypical Angina

Figure 7 shows the patients with atypical angina and whether that have heart disease or not. This plot was not very surprising and in line with what one would think. Atypical angina is chest pain associated with digestive/ lung disease. Hence why we see significantly more patients that are associated with atypical with no signs of heart disease.

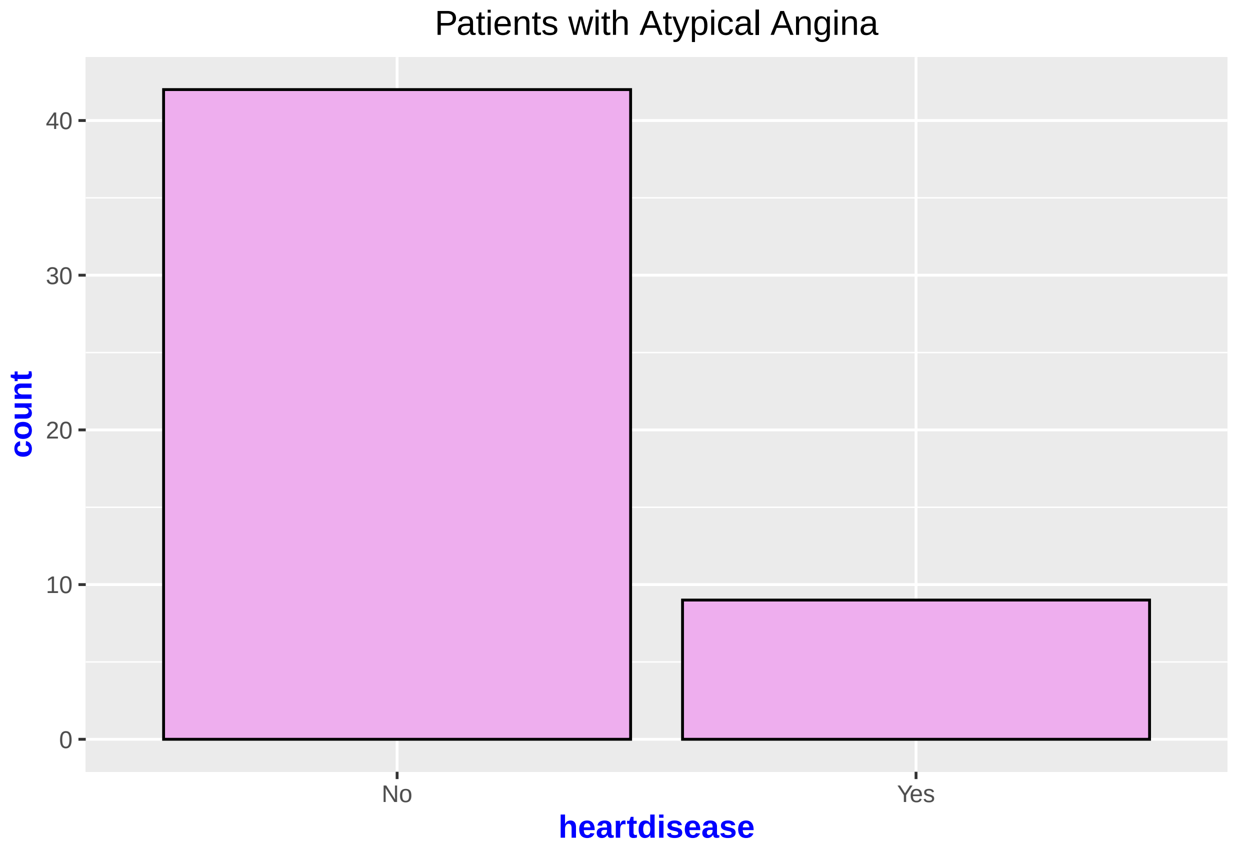

Figure 7: Barplot of Atypical Angina

6.10 Patients with Non- Anginal Angina

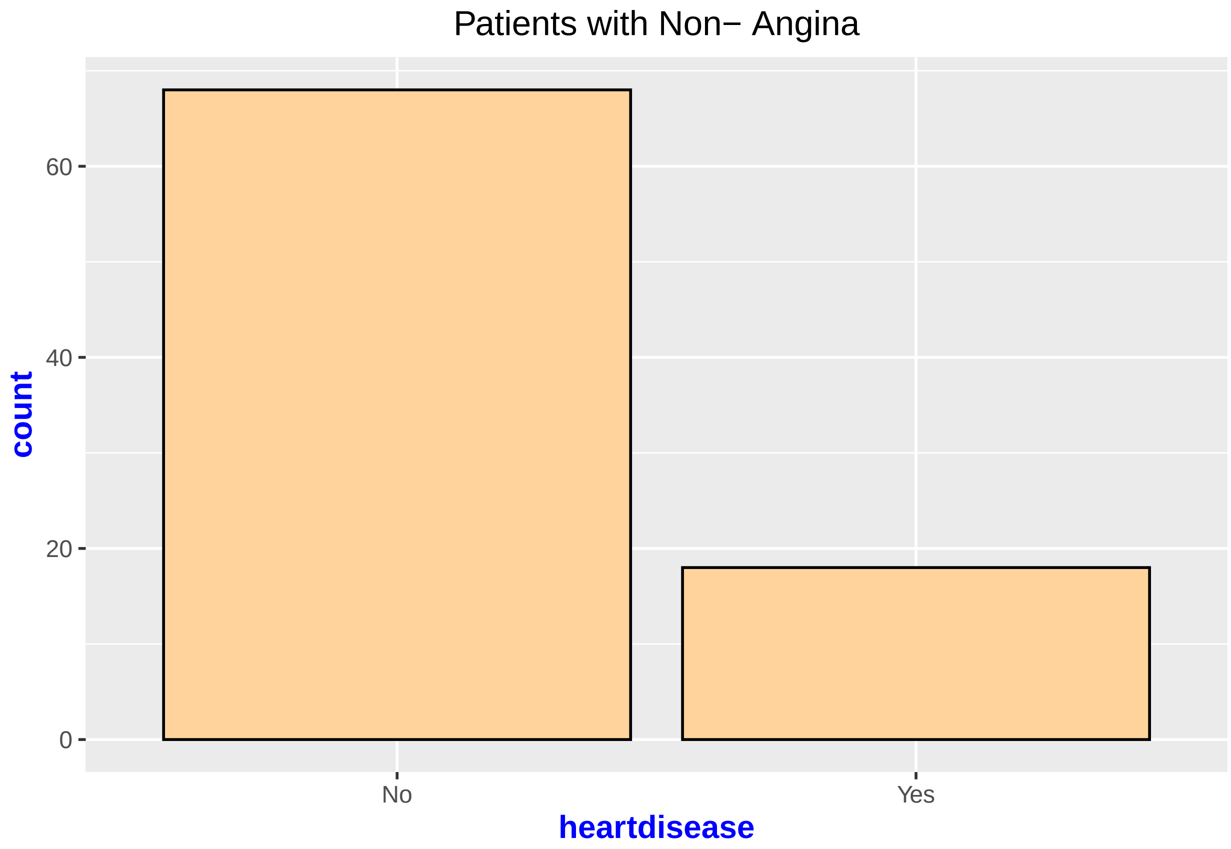

Figure 8: Barplot of Non- Anginal Angina

Figure 8 shows the patients with non- anginal angina and whether that have heart disease or not. This plot was not very surprising and in line with what one would think. Non- anginal angina is chest pain associated with muscular or skeletal pain such as rib fractures or working out. Hence why we see significantly more patients that are associated with non- anginal angina with no signs of heart disease.

6.11 Patients with Asymptomatic Angina

Figure 9 shows the patients with asymptomatic angina and whether that have heart disease or not. This plot was surprising since one would think that asymptomatic would indicate no symptoms.

6.12 Patients EKG

Figure 10 shows the patient's resting EKG type grouped by heart disease presence. We see most of our data in the *lv hypertrophy* and *normal* categories but not many patients in *st-t abnormality*. This is most likely since ST abnormality only occurs while someone is having a heart attack. LV hypertrophy is left ventricular hypertrophy. This is a sign of chronic heart disease that could be a result of chronic high blood pressure and heart valvular disease. We do see a high number of patients with no signs associated with LV hypertrophy, this could be since this is too early on to associate heart disease with these patients.

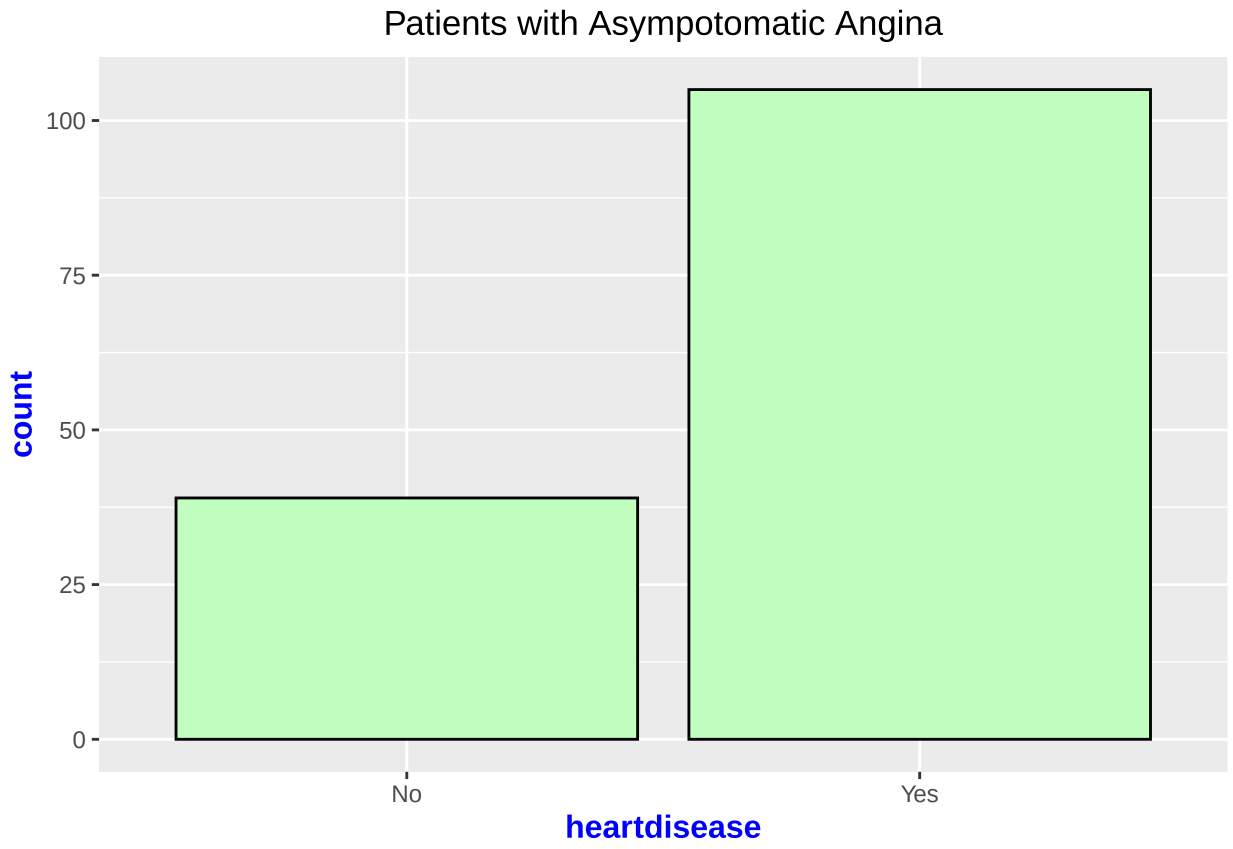

Figure 9: Barplot of Asymptomatic Angina

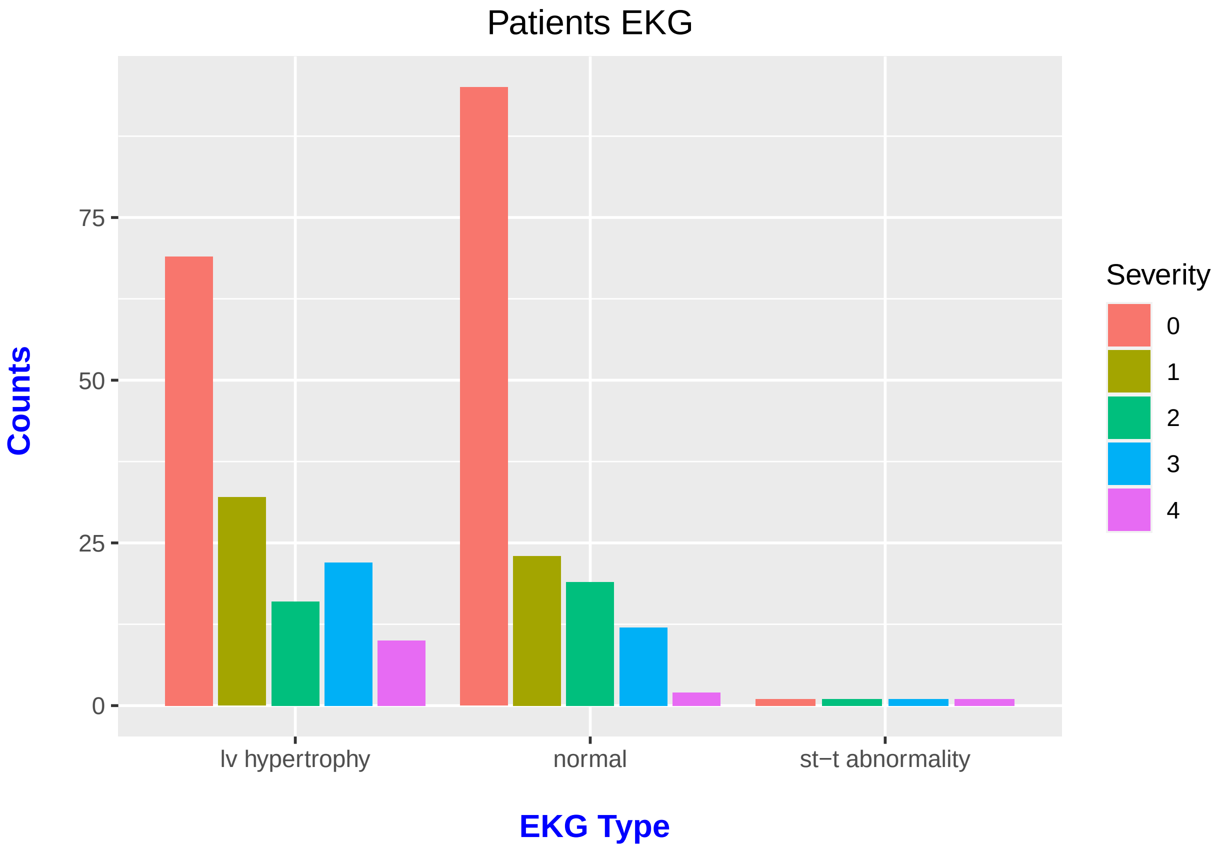

Figure 10: Barplot of Patient's EKG grouped by Heart Disease Presence

7 Modeling

In this section, we are going to implement various models to try and accurate predict our response feature. We are going to manipulate our response in the following ways:

- Leave as is, keeping the $0 - 4$ numerical scale
- Grouping the scale to *None*, *Mild*, *Severe*
- Making it a binary response, Whether Patient has Presence of Heart Disease

7.1 Using the Numerical Scale

7.1.1 Random Forest

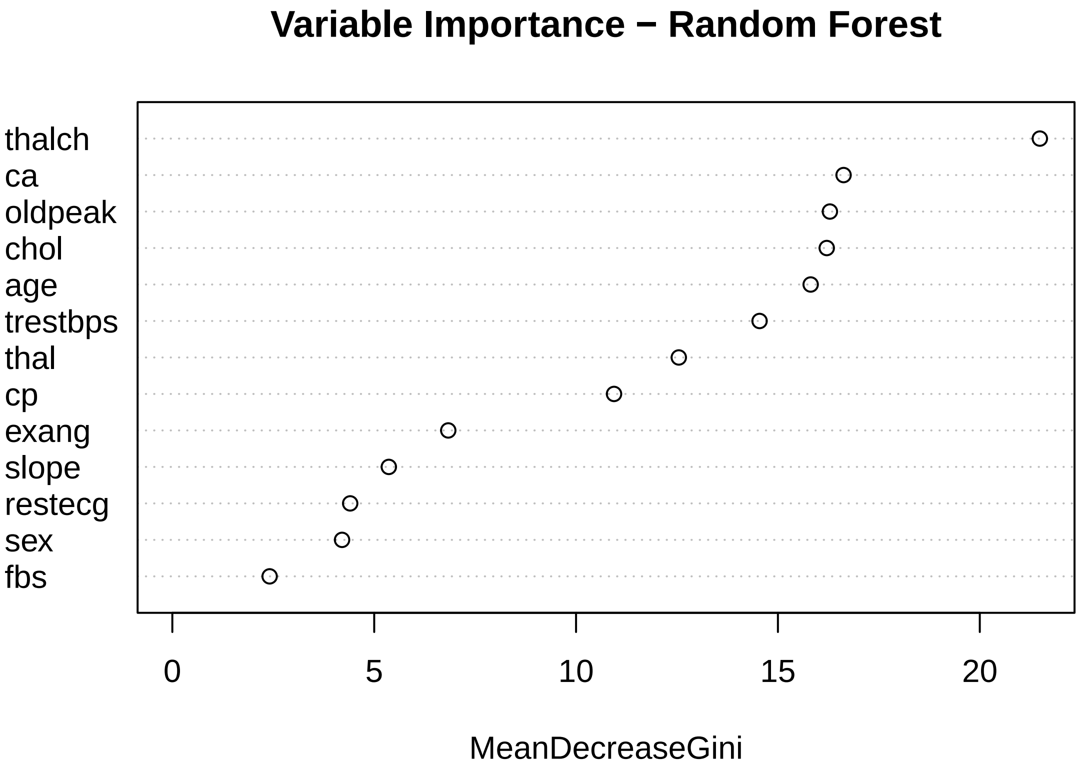

Figure 11: Variable Importance - Random Forest

Performing a random forest, we were 0.59375 percent correct in our model. Figure 11. shows that the *thalch* feature was the most important feature in determining the presence of heart disease. Where *fbs* was the least determining factor.

7.1.2 Classification Tree

Performing a classification tree, we were 0.578125 percent correct in our model. Figure 12 shows the tree itself. At the top of the node is the overall probability of a patient's presence of heart disease. Showing the

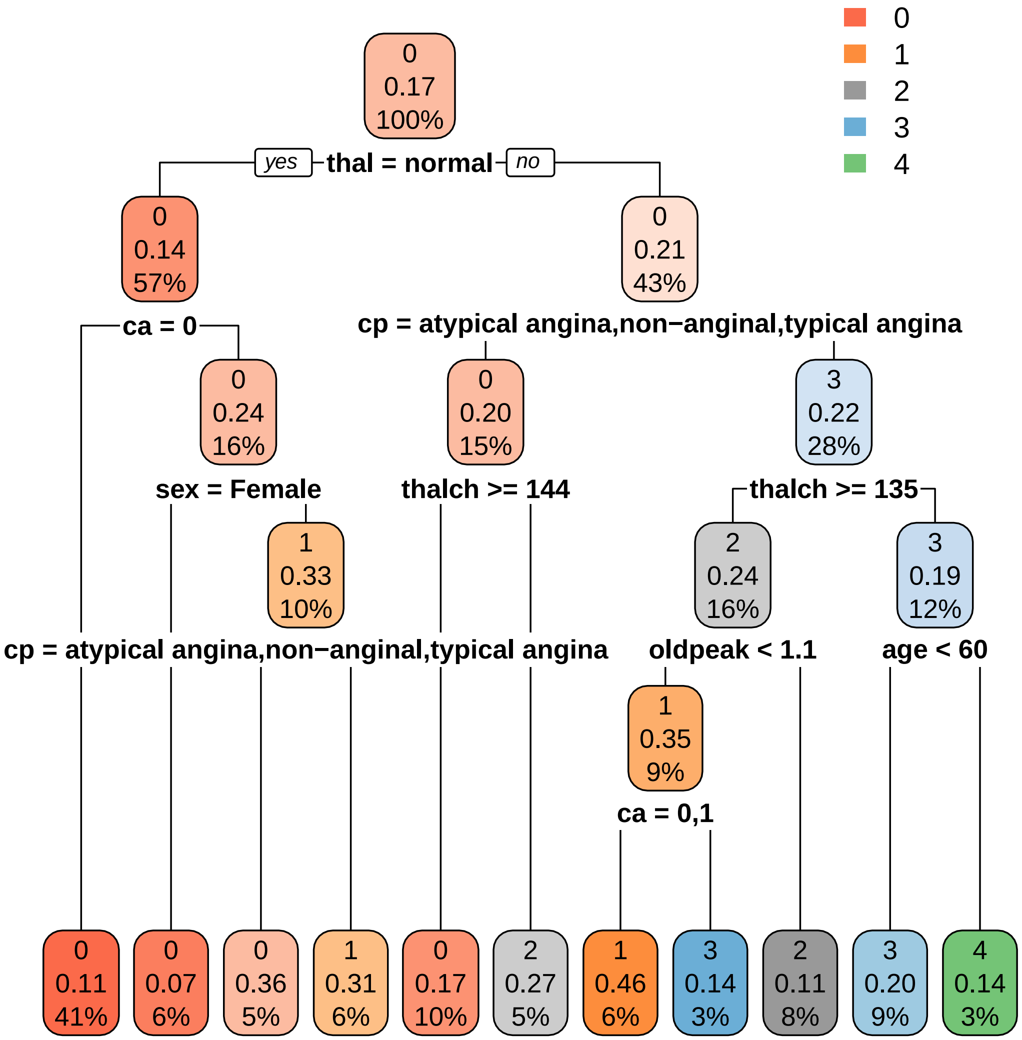

Figure 12: Classification Tree

proportion that a patient is 0 no signs of presence of heart disease. In this case 17 percent. Node then asks whether the thalassemia is normal or otherwise or otherwise. If normal, the tree then moves down the left side showing that a patient that has a normal thalassemia, has a probability of 14 percent of being 0, no signs of presence of heart disease, 57 percent of patients with normal thalassemia. The next node asks if $ca = 0$ if yes, then probability of the patient being 0, no signs of heart disease is 11 percent.

7.2 Grouping the Numerical Response Variable to *None*, *Mild*, *Severe*

For this section, we group our response the following way:

- $num = 0$ would be *None*
- $num = 1$ or $num = 2$ would be *Mild*
- $num = 3$ or $num = 4$ would be *Severe*

7.2.1 Random Forest

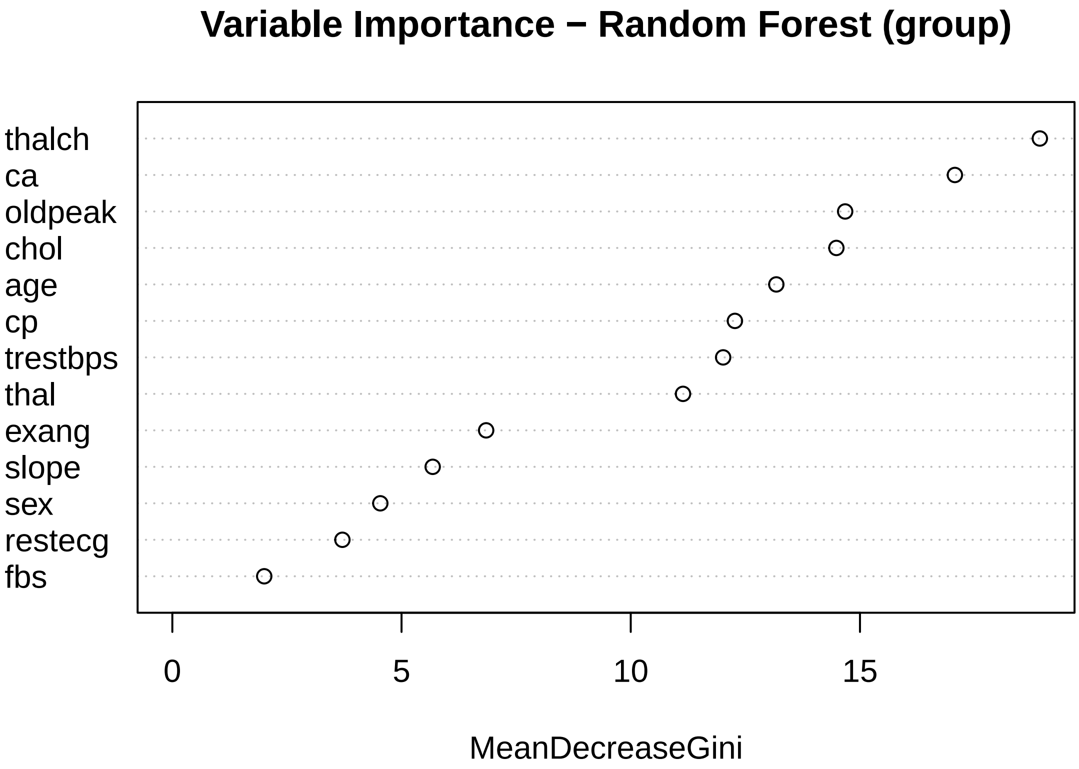

Figure 13: Variable Importance - Random Forest

Performing a random forest, we were 0.734375 percent correct in our model. Figure 13. shows that the *thalch* feature was the most important feature in determining the presence of heart disease. Where *fbs* was the least determining factor. This is in line with our first random forest. This random forest performed significantly better than our previous one.

7.2.2 Classification Tree

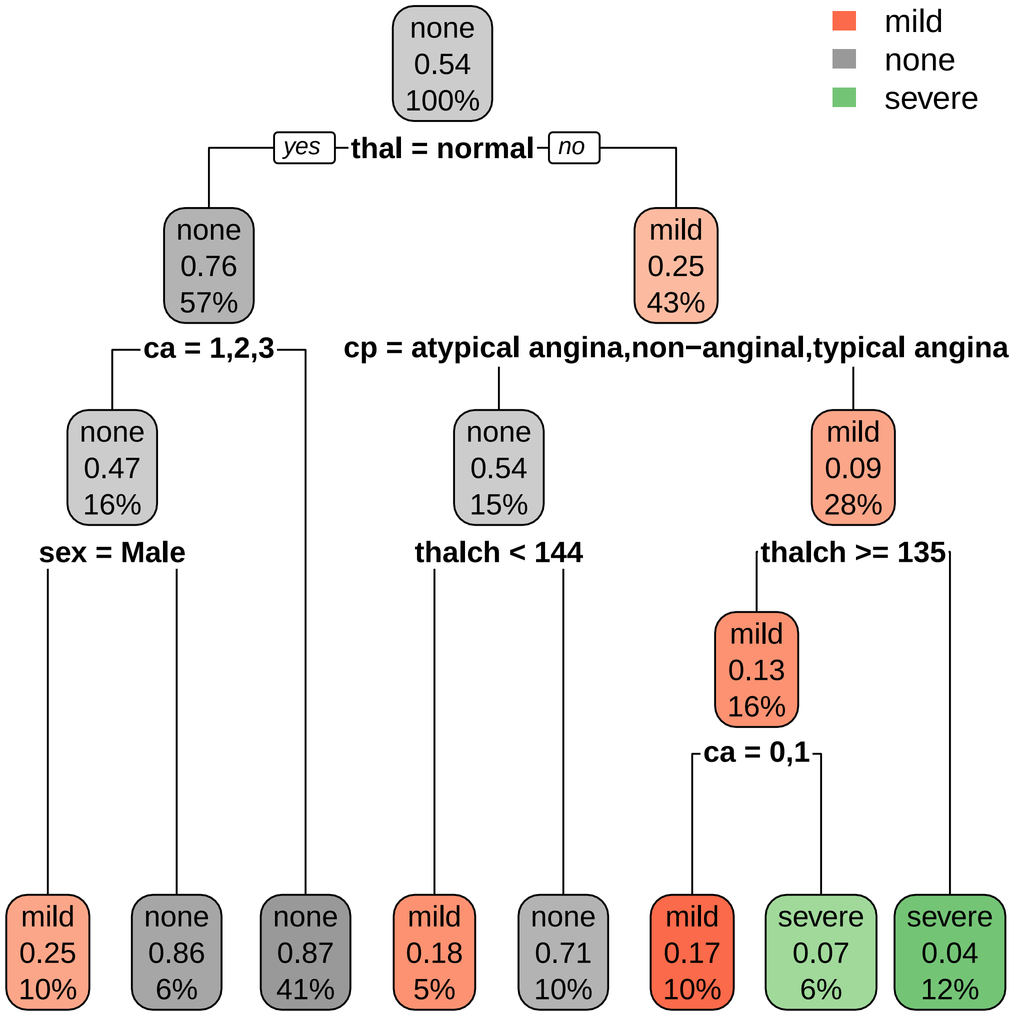

Figure 14: Classification Tree(Grouped)

Performing a classification tree, we were 0.59375 percent correct in our model only slightly better than our previous tree.

7.3 Binary Response, Patients with Presence of Heart Disease

For this section, we turn our response binary:

- $num = 0$ would be 0
- $num = 1$ or $num = 2$ or $num = 3$ or $num = 4$ would be 1

7.3.1 Random Forest

Performing a random forest, we were 0.703125 percent correct in our model. Figure 15. shows that the *thalch* and *ca* features was the most important feature in determining the presence of heart disease. Where *fbs*

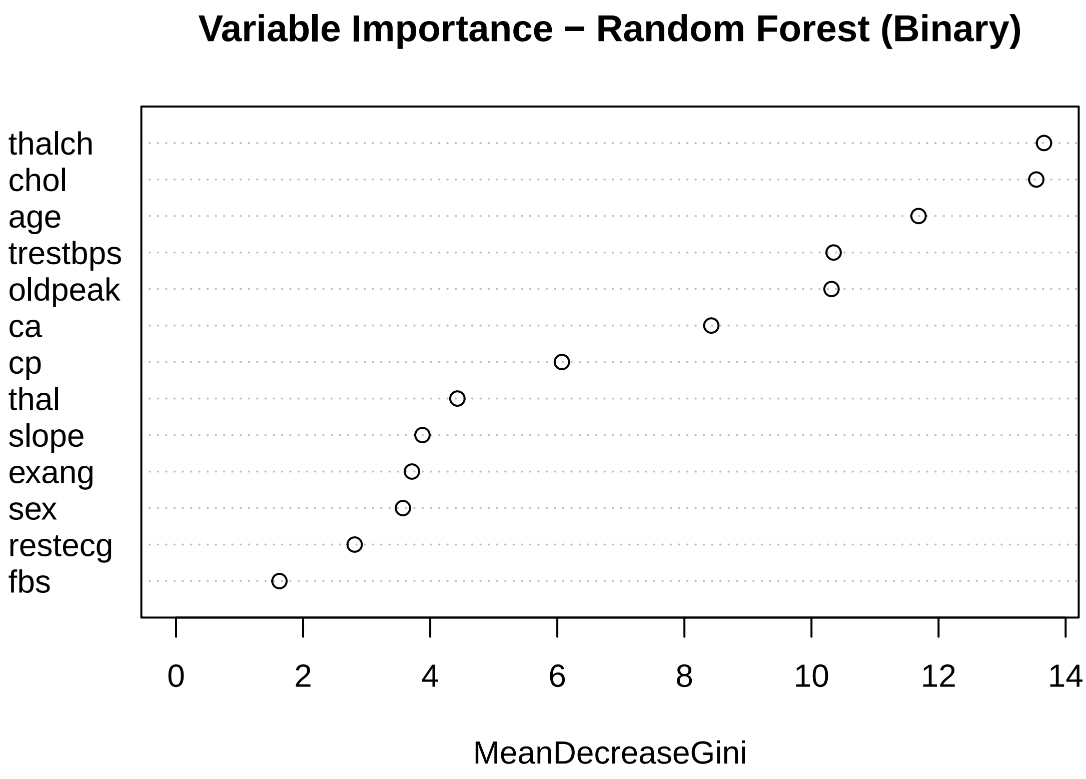

Figure 15: Variable Importance - Random Forest

was the least determining factor. This is in line with our first random forest. This random forest performed slightly worse than our previous one.

7.3.2 Classification Tree

Performing a classification tree, we were 0.671875 percent correct in our model. This tree performed significantly better than the previous ones. Please Refer to Figure 16. for the classification tree.

7.3.3 Linear Discriminant Analysis (LDA)

Table 3: LDA Confusion Matrix

	0	1
0	12	7
1	10	35

Performing Linear Discriminant Analysis (LDA), we were 0.734375 percent correct in our model. Our confusion matrix is shown in table 3.

7.3.4 Neural Net

```
## [1] 0.328125
```

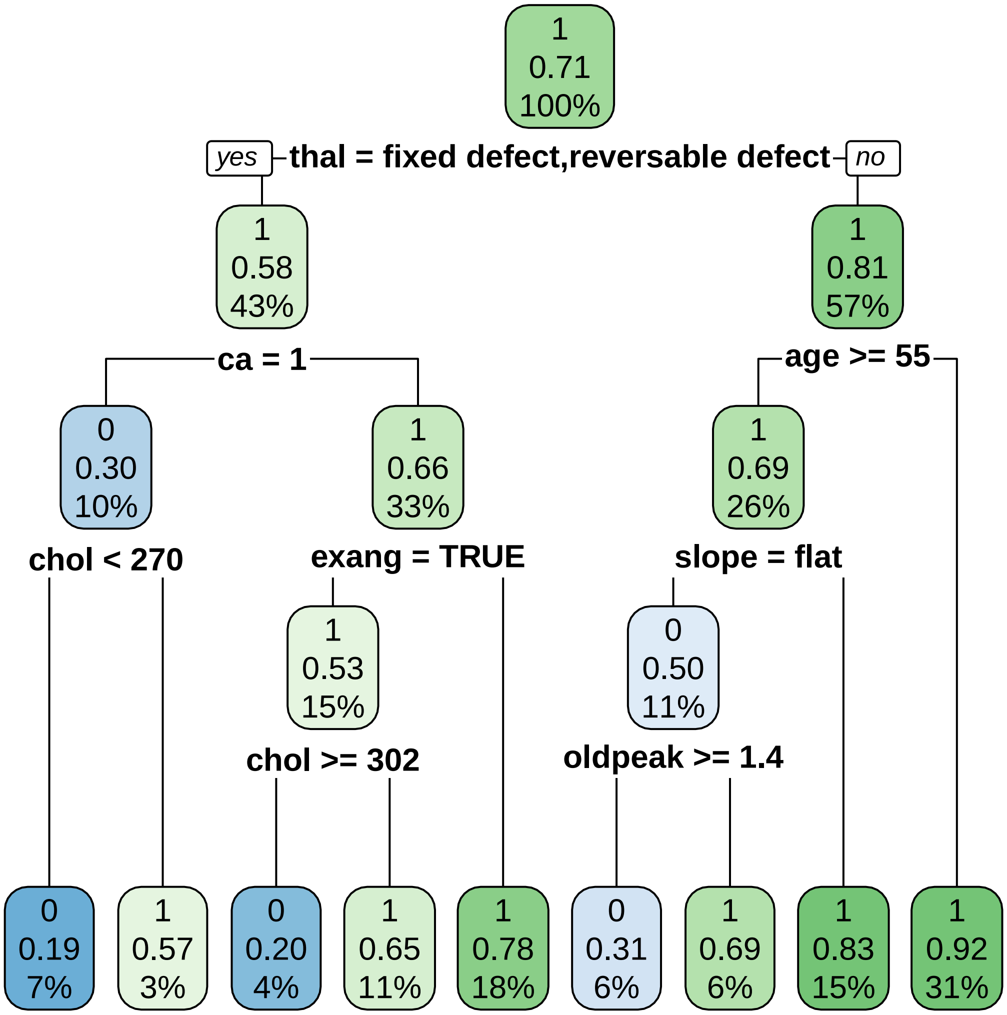

Figure 16: Classification Tree (Binary)

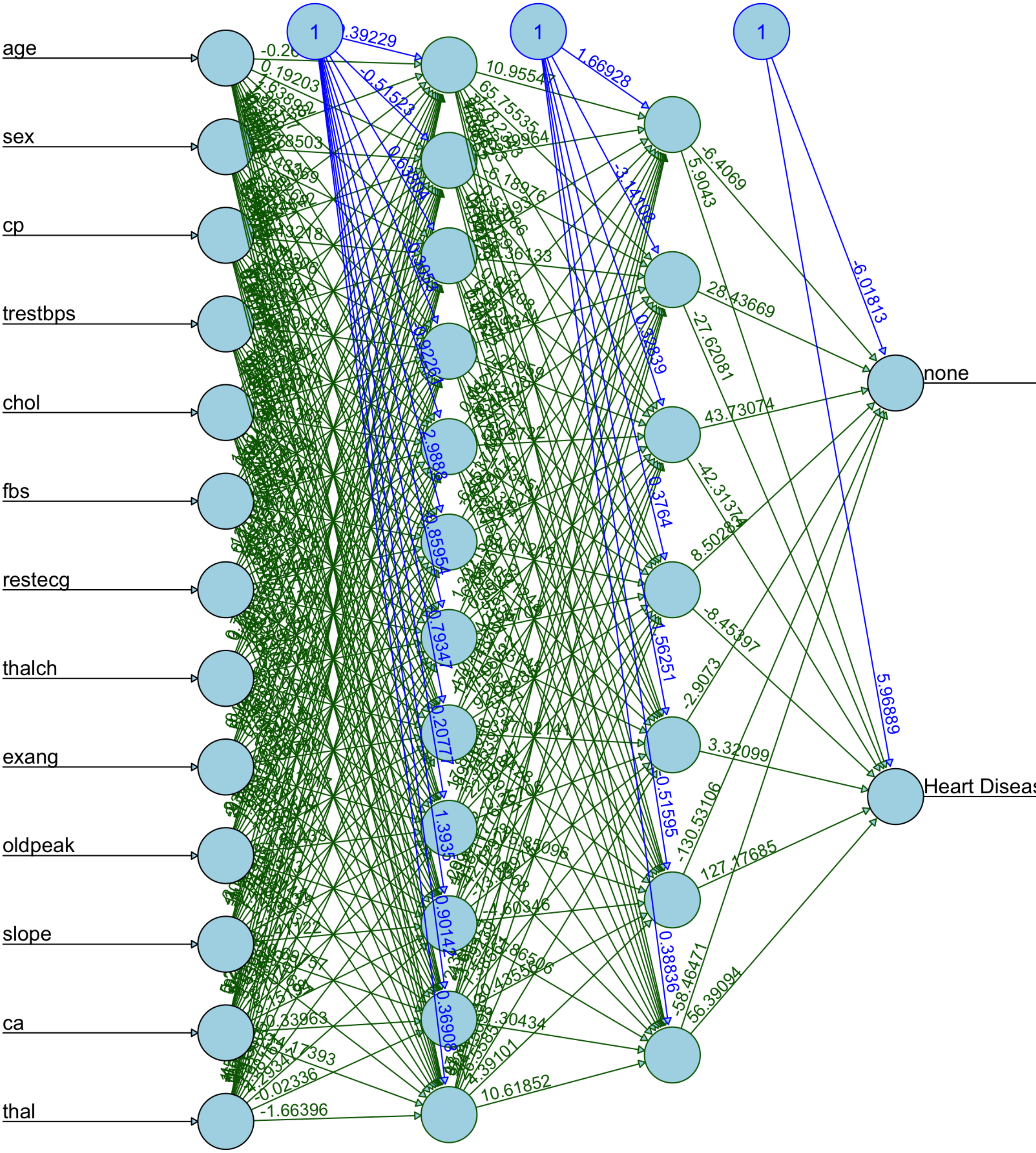

Figure 17: Neural Net (Binary)

Performing the neural net, we were 0.328125 percentage correct in our model. This is the lowest percentage out of all the models so far. There might need to do some fine tuning with neural net.

8 Results

Summarizing our results, we have the following:

8.1 Numerical Scale(no change to response)

Table 4: Numerical Scale Results by Model and Accuracy

Model	Accuracy
Random Forest	0.593750
Classification Tree	0.578125

Table 4 shows that both the random forest and the classification tree models performed similarly.

8.2 Grouping the Numerical Response Variable to ***None***, ***Mild***, ***Severe***

Table 5: Grouped Results by Model and Accuracy

Model	Accuracy1
Random Forest	0.734375
Classification Tree	0.593750

Table 5 shows that grouping the response feature to *None*, *Mild*, *Severe* the random forest performed significantly more accurate compared to the classification tree. The classification tree performed slightly between than the previous tree.

8.3 Binary Response, Patients with Presence of Heart Disease

Table 6: Binary Results by Model and Accuracy

Model	Accuracy2
Random Forest	0.703125
Classification Tree	0.671875
LDA	0.734375
Neural Net	0.328125

Table 6 shows that the LDA model performed the best followed by the random forest. Whereas the neural net performed the worst.

9 Discussion

Through our exploratory data analysis and statistical models, we were able to see that some of our important features in determining the presence of heart disease are as follows:

- ca (Coronary Artery/ Number of major vessels (0-3) colored by fluroscopy)
- thalch (Maximum heart rate achieved)
- chol (Cholesterol in mg/dl)
- age (Age in Years)
- oldpeak (ST Depression induced by exercise relative to rest)

Through our statistical modeling, it seems like grouping the response feature, *num* to smaller categories or even making it binary performed the best in predicting the presence of heart disease in a patient.

However, due to the time constraint, I was not able to perform some more modeling and get a more accurate prediction. I would have liked to take the top 5 most important features and rerun these models and see if we could get a better more accurate prediction. Adding on to that, I would have liked to fine tune my neural net more to get an accuracy closer to the other models as well as performing a logistic regression and fine tuning that model by performing forwards and backwards elimination to select a subset of features. Clustering the dataset was something I hoped to do but again, due to the time constraint. I was not able to get the clustering to output correctly.

Given the time, our models were able to adequately predict the presence of heart disease in a patient. I would recommend our random forest model that was grouped since it gave a 0.734375 probability of correctly predicting the presence of heart disease without losing too much information.

10 References

- “Cardiovascular Diseases (Cvds).” World Health Organization, World Health Organization, www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds). Accessed 14 Nov. 2023.

- Detrano, R et al. “International application of a new probability algorithm for the diagnosis of coronary artery disease.” The American journal of cardiology vol. 64,5 (1989): 304-10. doi:10.1016/0002-9149(89) 90524-9

- “Health Threats from High Blood Pressure.” Www.Heart.Org, 1 June 2023, www.heart.org/en/health-topics/high-blood-pressure/health-threats-from-high-blood-pressure.

- “Heart Disease Prevalence - Health, United States.” Centers for Disease Control and Prevention, Centers for Disease Control and Prevention, www.cdc.gov/nchs/hus/topics/heart-disease-prevalence.htm. Accessed 14 Nov. 2023.

- “High Cholesterol.” Mayo Clinic, Mayo Foundation for Medical Education and Research, 11 Jan. 2023, www.mayoclinic.org/diseases-conditions/high-blood-cholesterol/symptoms-causes/syc-20350800#:~:text=With%20high%20cholesterol%2C%20you%20can,a%20heart%20attack%20or%20stroke.

- “Learn More about Thalassemia.” Centers for Disease Control and Prevention, Centers for Disease Control and Prevention, 19 Oct. 2023, www.cdc.gov/ncbddd/thalassemia/index

- Morise, A P et al. “Comparison of logistic regression and Bayesian-based algorithms to estimate posttest probability in patients with suspected coronary artery disease undergoing exercise ECG.” Journal of electrocardiology vol. 25,2 (1992): 89-99. doi:10.1016/0022-0736(92)90113-e

11 Appendix - R Script

```r
# heart_data
# =read.table('~/Documents/CSUF/data/cleveland.data', sep =
# '\t', na.strings = c('','NA'))
heart_disease = read.csv("~/Documents/CSUF/data/heart_disease.csv",
    header = T, na.strings = c("", "NA"))
dim(heart_disease)
head(heart_disease)
heart_disease$num

library(ggplot2)
library(plyr)
library(dplyr)
library(stats)
library(ggfortify)
library(randomForest)
library(caret)
library(tidyr)
library(partykit)
library(neuralnet)
library(nnet)
library(leaps)
library(naivebayes)
library(class)
library(boot)
library(MASS)
library(rpart)

number = as.factor(heart_disease$num)
summary(number)

female = as.factor(heart_disease$sex == "Female")
summary(female)

tbl = table(number, female)
tbl

tbl_sum = addmargins(tbl, FUN = sum)
tbl_sum

counts = c(267, 144, 235, 30, 99, 10, 99, 8, 26, 2)
```

```r
num = c(0, 1, 2, 3, 4)
female = c("no", "yes")
new_data = data.frame(counts, num, female)
new_data

ggplot(data = new_data, aes(x = num, y = counts, fill = female)) +
    geom_bar(stat = "identity") + labs(x = "\n Severity of Heart Disease",
    y = "Counts \n", title = "Serverity of Heart Disease by Female",
    fill = "Female") +  fill = "Female") +
theme(plot.title = element_text(hjust = 0.5), axis.title.x = element_text(face = "bold",
    colour = "blue", size = 12), axis.title.y = element_text(face = "bold",
    colour = "blue", size = 12), legend.position = "bottom")

means = ddply(heart_disease, "sex", summarise, grp.mean = mean(age))
head(means)

ggplot(data = heart_disease, aes(x = age, fill = sex, color = sex)) +
    geom_histogram(position = "dodge") + geom_vline(data = means,
    aes(xintercept = grp.mean, color = sex), linetype = "dashed")

# subset
cleveland_data = heart_disease[heart_disease$dataset == "Cleveland",
    ]

head(cleveland_data)

means_subset = ddply(cleveland_data, "sex", summarise, grp.mean = mean(age))
head(means)

# age histo
ggplot(data = cleveland_data, aes(x = age, fill = sex, color = sex)) +
    geom_histogram(position = "dodge") + geom_vline(data = means,
    aes(xintercept = grp.mean, color = sex), linetype = "dashed")
# bar
number = as.factor(cleveland_data$num)
summary(number)

male = as.factor(cleveland_data$sex == "Male")
summary(male)

tbl = table(number, male)
tbl
```

```
tbl_sum = addmargins(tbl, FUN = sum)
tbl_sum

counts = c(72, 93, 9, 46, 7, 29, 7, 28, 2, 11)
num = c(0, 1, 2, 3, 4)
male = c("No", "Yes")
new_data = data.frame(counts, num, male)
new_data

ggplot(data = new_data, aes(x = num, y = counts, fill = male)) +
    geom_bar(stat = "identity") + labs(x = "\n Severity of Heart Disease",
    y = "Counts \n", title = "Serverity of Heart Disease by Male",
    fill = "Male") + theme(plot.title = element_text(hjust = 0.5),
    axis.title.x = element_text(face = "bold", colour = "blue",
        size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "bottom")

# cholerstol histo

cleveland_data_subset = cleveland_data[cleveland_data$num > 0,
    ]
# cleveland_data_subset = subset(cleveland_data, num > 0)
head(cleveland_data_subset)
cleveland_data_subset$num = as.factor(cleveland_data_subset$num)

ggplot(data = cleveland_data_subset, aes(x = chol, fill = num,
    color = num)) + geom_histogram(position = "identity") + labs(x = "\n Cholesterol Levels (mg/dl)",
    y = "Counts \n", title = "Cholesterol Levels with Patients associated with Heart Disease",
    fill = "num") + geom_vline(xintercept = 200) +  fill =
    fill = "num") + geom_vline(xintercept = 200) +  "num")
    fill = "num") + geom_vline(xintercept = 200) +  +
    fill = "num") + geom_vline(xintercept = 200) +  geom_vline(xintercept
    fill = "num") + geom_vline(xintercept = 200) +  = 200)
    fill = "num") + geom_vline(xintercept = 200) +  +
theme(plot.title = element_text(hjust = 0.5), axis.title.x = element_text(face = "bold",
    colour = "blue", size = 12), axis.title.y = element_text(face = "bold",
    colour = "blue", size = 12), legend.position = "right")

# BP histo

ggplot(data = cleveland_data_subset, aes(x = trestbps, fill = num,
    color = num)) + geom_histogram(position = "identity") + labs(x = "\n Blood Pressure",
    y = "Counts \n", title = "Blood Pressure with Patients associated with Heart Disease",
    fill = "num") + geom_vline(xintercept = 90) + theme(plot.title = element_text(hjust = 0.5),
    axis.title.x = element_text(face = "bold", colour = "blue",
        size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "right")
# num1
subset1 = cleveland_data_subset[cleveland_data_subset$num ==
    1, ]

ggplot(data = subset1, aes(x = trestbps, fill = "num")) + geom_histogram(col = I("black")) +
    labs(x = "\n Blood Pressure", y = "Counts \n", title = "Blood Pressure with Patients associated wit
    scale_fill_manual(name = "num", values = "coral1", labels = c("1")) +
    theme(plot.title = element_text(hjust = 0.5), axis.title.x = element_text(face = "bold",
        colour = "blue", size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "right")
# num2
subset2 = cleveland_data_subset[cleveland_data_subset$num ==
    2, ]

ggplot(data = subset2, aes(x = trestbps, fill = "num")) + geom_histogram(col = I("black")) +
    labs(x = "\n Blood Pressure", y = "Counts \n", title = "Blood Pressure with Patients associated wit
    scale_fill_manual(name = "num", values = "chartreuse3", labels = c("2")) +
    theme(plot.title = element_text(hjust = 0.5), axis.title.x = element_text(face = "bold",
        colour = "blue", size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "right")
# num3
subset3 = cleveland_data_subset[cleveland_data_subset$num ==
    3, ]

ggplot(data = subset3, aes(x = trestbps, fill = "num")) + geom_histogram(col = I("black")) +
    labs(x = "\n Blood Pressure", y = "Counts \n", title = "Blood Pressure with Patients associated wit
    scale_fill_manual(name = "num", values = "turquoise3", labels = c("3")) +
    theme(plot.title = element_text(hjust = 0.5), axis.title.x = element_text(face = "bold",
        colour = "blue", size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "right")

# num4
subset4 = cleveland_data_subset[cleveland_data_subset$num ==
    4, ]

ggplot(data = subset4, aes(x = trestbps, fill = "num")) + geom_histogram(col = I("black")) +
    labs(x = "\n Blood Pressure", y = "Counts \n", title = "Blood Pressure with Patients associated wit
    scale_fill_manual(name = "num", values = "orchid3", labels = c("4")) +
    theme(plot.title = element_text(hjust = 0.5), axis.title.x = element_text(face = "bold",
        colour = "blue", size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "right")
# thal
as.factor(cleveland_data_subset$thal)
cleveland_data_subset["thal"][cleveland_data_subset["thal"] ==
    ""] <- "na"
# cleveland_data_subset[-grep('na',
# cleveland_data_subset$thal)]
# cleveland_data_subset[!cleveland_data_subset$thal ==
# 'na', ]

cleveland_data_subset = cleveland_data_subset[!cleveland_data_subset$thal ==
    "na", ]

ggplot(data = cleveland_data_subset, aes(x = thal, fill = "Counts")) +
    geom_bar(col = I("black")) + theme(plot.title = element_text(hjust = 0.5),
    axis.title.x = element_text(face = "bold", colour = "blue",
        size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "right")


cleveland_data_subset = transform(cleveland_data_subset, heartdisease = ifelse(cleveland_data_subset$nu
    0, "No", "Yes"))

cleveland_data = transform(cleveland_data, heartdisease = ifelse(cleveland_data$num ==
    0, "No", "Yes"))

# typical Angina
typical_angina = cleveland_data[cleveland_data$cp == "typical angina",
    ]
ggplot(data = typical_angina, aes(x = heartdisease, fill = "Heart Disease")) +
    geom_bar(fill = "dodgerblue", col = I("black")) + labs(title = "Patients with Typical Angina") +
    theme(plot.title = element_text(hjust = 0.5), axis.title.x = element_text(face = "bold",
        colour = "blue", size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "right")

# atypical Angina
atypical_angina = cleveland_data[cleveland_data$cp == "atypical angina",
    ]
ggplot(data = atypical_angina, aes(x = heartdisease, fill = "Heart Disease")) +
    geom_bar(fill = "plum2", col = I("black")) + labs(title = "Patients with Atypical Angina") +
    theme(plot.title = element_text(hjust = 0.5), axis.title.x = element_text(face = "bold",
        colour = "blue", size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "right")

# non_anginal
non_anginal = cleveland_data[cleveland_data$cp == "non-anginal",
    ]
ggplot(data = non_anginal, aes(x = heartdisease, fill = "Heart Disease")) +
    geom_bar(fill = "burlywood1", col = I("black")) + labs(title = "Patients with Non- Angina") +
    theme(plot.title = element_text(hjust = 0.5), axis.title.x = element_text(face = "bold",
        colour = "blue", size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "right")

# asymptomatic
asymptomatic = cleveland_data[cleveland_data$cp == "asymptomatic",
    ]
ggplot(data = asymptomatic, aes(x = heartdisease, fill = "Heart Disease")) +
    geom_bar(fill = "darkseagreen1", col = I("black")) + labs(title = "Patients with Asympotomatic Angi
    theme(plot.title = element_text(hjust = 0.5), axis.title.x = element_text(face = "bold",
        colour = "blue", size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "right")
# ekg

ggplot(data = cleveland_data, aes(x = factor(restecg), fill = factor(num))) +
    geom_bar(position = "dodge2") + labs(title = "Patients EKG",
    x = "\n EKG Type", y = "Counts \n", fill = "Severity") +
    theme(plot.title = element_text(hjust = 0.5), axis.title.x = element_text(face = "bold",
        colour = "blue", size = 12), axis.title.y = element_text(face = "bold",
        colour = "blue", size = 12), legend.position = "right")

### data wrangling
heart_disease = cleveland_data[-c(1, 4, 17)]
str(heart_disease)
attach(heart_disease)
heart_disease$sex = as.factor(heart_disease$sex)
heart_disease$cp = as.factor(heart_disease$cp)
heart_disease$fbs = as.factor(heart_disease$fbs)
heart_disease$restecg = as.factor(heart_disease$restecg)
heart_disease$exang = as.factor(heart_disease$exang)
heart_disease$slope = as.factor(heart_disease$slope)
heart_disease$ca = as.factor(heart_disease$ca)
heart_disease$thal = as.factor(heart_disease$thal)
heart_disease$num = as.factor(heart_disease$num)
heart_disease$age = as.numeric(heart_disease$age)
heart_disease$trestbps = as.numeric(heart_disease$trestbps)
```

```r
heart_disease$chol = as.numeric(heart_disease$chol)
heart_disease$thalch = as.numeric(heart_disease$thalch)
heart_disease$oldpeak = as.numeric(heart_disease$oldpeak)

heart_disease = na.omit(heart_disease)

### split to train and testing 80/20

set.seed(538)
index = sample(2, nrow(heart_disease), replace = TRUE, prob = c(0.8,
    0.2))
training = heart_disease[index == 1, ]
testing = heart_disease[index == 2, ]

##### Random Forest

RF = randomForest(training$num ~ ., data = training)

RF_pred = predict(RF, testing)
testing$num_pred = RF_pred

CM = table(testing$num, testing$num_pred)
CM

RF_accuracy = sum(diag(CM)/sum(CM))
RF_accuracy

varImpPlot(RF, main = "Variable Importance - Random Forest")

##### Classification Tree
myf = num ~ age + sex + cp + trestbps + chol + fbs + restecg +
    thalch + exang + oldpeak + slope + ca + thal

ctree_heart = ctree(myf, data = training)
table(predict(ctree_heart), training$num)

plot(ctree_heart)

ctree_test = predict(ctree_heart, newdata = testing)
ctree_tbl = table(ctree_test, testing$num)
ctree_accuracy = sum(diag(ctree_tbl)/sum(ctree_tbl))
ctree_accuracy
```

```
dtree = rpart(myf, data = training, method = "class")
rpart.plot(dtree, extra = 106)

dtree_pred = predict(dtree, testing, type = "class")
dtree_tbl = table(dtree_pred, testing$num)

dtree_accuracy = sum(diag(dtree_tbl)/sum(dtree_tbl))

##### NN

# NN = neuralnet(num ~., data = heart_disease, hidden =
# c(5,3)) NN = nnet(num ~ . , data = training, size = 10,
# maxit = 100)

##### making response None/ Mild/ Severe

heart_disease1 = cleveland_data[-c(1, 4, 17)]
str(heart_disease1)
attach(heart_disease1)
heart_disease1$sex = as.factor(heart_disease1$sex)
heart_disease1$cp = as.factor(heart_disease1$cp)
heart_disease1$fbs = as.factor(heart_disease1$fbs)
heart_disease1$restecg = as.factor(heart_disease1$restecg)
heart_disease1$exang = as.factor(heart_disease1$exang)
heart_disease1$slope = as.factor(heart_disease1$slope)
heart_disease1$ca = as.factor(heart_disease1$ca)
heart_disease1$thal = as.factor(heart_disease1$thal)
heart_disease1$num = as.factor(heart_disease1$num)
heart_disease1$age = as.numeric(heart_disease1$age)
heart_disease1$trestbps = as.numeric(heart_disease1$trestbps)
heart_disease1$chol = as.numeric(heart_disease1$chol)
heart_disease1$thalch = as.numeric(heart_disease1$thalch)
heart_disease1$oldpeak = as.numeric(heart_disease1$oldpeak)

# sum(heart_disease1$num == 0) sum(heart_disease1$num == 1)
# sum(heart_disease1$num == 2) sum(heart_disease1$num == 3)
# sum(heart_disease1$num == 4)

heart_disease1$num = as.numeric(heart_disease1$num)

# sum(heart_disease1$num == 0) sum(heart_disease1$num == 1)
# sum(heart_disease1$num == 2) sum(heart_disease1$num == 3)
# sum(heart_disease1$num == 4) sum(heart_disease1$num == 5)
```

```
heart_disease1$num[heart_disease1$num == 1] = "none"
heart_disease1$num[heart_disease1$num == 2 | heart_disease1$num ==
    3] <- "mild"
heart_disease1$num[heart_disease1$num == 4 | heart_disease1$num ==
    5] <- "severe"

heart_disease1$num = as.factor(heart_disease1$num)

heart_disease1 = na.omit(heart_disease1)

### split to train and testing 80/20

set.seed(538)
index1 = sample(2, nrow(heart_disease1), replace = TRUE, prob = c(0.8,
    0.2))
training1 = heart_disease1[index1 == 1, ]
testing1 = heart_disease1[index1 == 2, ]

##### Random Forest

RF1 = randomForest(training1$num ~ ., data = training1)

RF_pred1 = predict(RF1, testing1)
testing1$num_pred = RF_pred1

CM1 = table(testing1$num, testing1$num_pred)
CM1

RF_accuracy1 = sum(diag(CM)/sum(CM))
RF_accuracy1

varImpPlot(RF1, main = "Variable Importance - Random Forest (group)")

##### Classification Tree
myf = num ~ age + sex + cp + trestbps + chol + fbs + restecg +
    thalch + exang + oldpeak + slope + ca + thal

ctree_heart = ctree(myf, data = training1)
table(predict(ctree_heart), training1$num)

plot(ctree_heart)
```

```
ctree_test1 = predict(ctree_heart, newdata = testing1)
ctree_tbl1 = table(ctree_test1, testing1$num)
ctree_accuracy1 = sum(diag(ctree_tbl1)/sum(ctree_tbl1))
ctree_accuracy1

##### making response binary

heart_disease2 = cleveland_data[-c(1, 4, 17)]
str(heart_disease2)
attach(heart_disease2)
heart_disease2$sex = as.factor(heart_disease2$sex)
heart_disease2$cp = as.factor(heart_disease2$cp)
heart_disease2$fbs = as.factor(heart_disease2$fbs)
heart_disease2$restecg = as.factor(heart_disease2$restecg)
heart_disease2$exang = as.factor(heart_disease2$exang)
heart_disease2$slope = as.factor(heart_disease2$slope)
heart_disease2$ca = as.factor(heart_disease2$ca)
heart_disease2$thal = as.factor(heart_disease2$thal)
heart_disease2$num = as.factor(heart_disease2$num)
heart_disease2$age = as.numeric(heart_disease2$age)
heart_disease2$trestbps = as.numeric(heart_disease2$trestbps)
heart_disease2$chol = as.numeric(heart_disease2$chol)
heart_disease2$thalch = as.numeric(heart_disease2$thalch)
heart_disease2$oldpeak = as.numeric(heart_disease2$oldpeak)

heart_disease2$num[heart_disease2$num != 0] = 1
heart_disease2$num = as.factor(heart_disease2$num)
heart_disease2 = na.omit(heart_disease2)

### split to train and testing 80/20

set.seed(538)
index2 = sample(2, nrow(heart_disease2), replace = TRUE, prob = c(0.8,
    0.2))
training2 = heart_disease2[index2 == 1, ]
testing2 = heart_disease2[index2 == 2, ]

##### Random Forest

RF2 = randomForest(training2$num ~ ., data = training2)

RF_pred2 = predict(RF2, testing2)
testing2$num_pred = RF_pred2
```

```
CM2 = table(testing2$num, testing2$num_pred)
CM2

RF_accuracy2 = sum(diag(CM2)/sum(CM2))
RF_accuracy2

varImpPlot(RF2, main = "Variable Importance - Random Forest (Binary)")

### Logistic regression
logreg = glm(num ~ ., data = training2, family = binomial("logit"))
summary(logreg)

model_full = logreg
model_null = glm(num ~ 1, data = training2, family = binomial("logit"))
model_forward = step(model_null, trace = F, scope = list(lower = formula(model_null),
    upper = formula(model_full)), direction = "forward")


reg_model = regsubsets(num ~ ., data = training2, nvmax = 5)
reg_sum = summary(reg_model)
reg_sum$outmat


### LDA
lda.all = lda(num ~ ., data = training2)
lda.pred.all = predict(lda.all, testing2)
CM3 = table(lda.pred.all$class, testing2$num)
CM3
lda_accuracy = sum(diag(CM3)/sum(CM3))
lda_accuracy

## NN

heart_disease3 = cleveland_data[-c(1, 4, 17)]
str(heart_disease3)
attach(heart_disease3)
heart_disease3$sex = as.factor(heart_disease3$sex)
heart_disease3$cp = as.factor(heart_disease3$cp)
heart_disease3$fbs = as.factor(heart_disease3$fbs)
heart_disease3$restecg = as.factor(heart_disease3$restecg)
heart_disease3$exang = as.factor(heart_disease3$exang)
heart_disease3$slope = as.factor(heart_disease3$slope)
```

```r
heart_disease3$ca = as.factor(heart_disease3$ca)
heart_disease3$thal = as.factor(heart_disease3$thal)
heart_disease3$num = as.factor(heart_disease3$num)
heart_disease3$age = as.numeric(heart_disease3$age)
heart_disease3$trestbps = as.numeric(heart_disease3$trestbps)
heart_disease3$chol = as.numeric(heart_disease3$chol)
heart_disease3$thalch = as.numeric(heart_disease3$thalch)
heart_disease3$oldpeak = as.numeric(heart_disease3$oldpeak)

heart_disease3$num[heart_disease3$num != 0] = 1

# heart_disease3$num = as.numeric(heart_disease3$num)

heart_disease3 = heart_disease3 %>%
    mutate_if(is.factor, as.numeric)
heart_disease3$num[heart_disease3$num == 1] = "none"
heart_disease3$num[heart_disease3$num == 2] <- "Heart Disease"
heart_disease3$num = as.factor(heart_disease3$num)
heart_disease3 = na.omit(heart_disease3)

### split to train and testing 80/20

set.seed(538)
index3 = sample(2, nrow(heart_disease3), replace = TRUE, prob = c(0.8,
    0.2))
training3 = heart_disease3[index2 == 1, ]
testing3 = heart_disease3[index2 == 2, ]

NN = neuralnet(num ~ ., data = training3, hidden = c(12, 7),
    linear.output = F, lifesign = "full", rep = 3)
plot(NN, col.hidden = "darkgreen", col.hidden.synapse = "darkgreen",
    show.weights = T, information = T, fill = "lightblue")


NN_pred = neuralnet::compute(NN, testing3[, c(1:13)])
NN_pred1 = data.frame()
for (i in 1:dim(testing3)[1]) {
    NN_pred1 = rbind(NN_pred1, which.max(NN_pred$net.result[i,
        ]))
}

NN_pred1$X1L = gsub(1, "None", NN_pred1$X1L)
NN_pred1$X1L = gsub(2, "Heart Disease", NN_pred1$X1L)
```

```r
NN_tbl = table(as.factor(NN_pred1$X1L), testing3$num)
NN_accuracy = sum(diag(NN_tbl)/sum(NN_tbl))
NN_accuracy

## dtree
myf = num ~ age + sex + cp + trestbps + chol + fbs + restecg +
    thalch + exang + oldpeak + slope + ca + thal
dtree = rpart(myf, data = training, method = "class")
rpart.plot(dtree, extra = 106)
dtree_pred = predict(dtree, testing, type = "class")
dtree_tbl = table(dtree_pred, testing$num)
dtree_accuracy = sum(diag(dtree_tbl)/sum(dtree_tbl))

### dtree 1

##### Classification Tree
myf = num ~ age + sex + cp + trestbps + chol + fbs + restecg +
    thalch + exang + oldpeak + slope + ca + thal
dtree1 = rpart(myf, data = training1, method = "class")
rpart.plot(dtree1, extra = 106)
dtree_pred1 = predict(dtree1, testing1, type = "class")
dtree_tbl1 = table(dtree_pred1, testing1$num)
dtree_accuracy1 = sum(diag(dtree_tbl1)/sum(dtree_tbl1))

### dtree 2

##### Classification Tree
myf = num ~ age + sex + cp + trestbps + chol + fbs + restecg +
    thalch + exang + oldpeak + slope + ca + thal

dtree2 = rpart(myf, data = training2, method = "class")

rpart.plot(dtree2, extra = 106)

dtree_pred2 = predict(dtree2, testing2, type = "class")
dtree_tbl2 = table(dtree_pred2, testing2$num)

dtree_accuracy2 = sum(diag(dtree_tbl2)/sum(dtree_tbl2))
```