PREDICTION OF HEART DISEASE

SCMT 650

GROUP 5

Brandon Dupy, Arpita Deshmukh,
Yuqi Hu, Nick Keeley

Table of Contents

1. EXECUTIVE SUMMARY

Objective: The objective of this study is to build a predictive model to accurately determine if a patient has heart disease or not.

Our objective has required us to utilize data that contains patient heart disease diagnose results and potentially relevant factors. We choose to analyze factors that contributes to heart disease because it can help an insurance company to evaluate health insurance price for a potential policy holder or a pharmaceutical company to better develop medicines for heart disease.

To achieve this, we used a publicly available data collected and provided by the University of California at Irvine in their machine learning repository [2]. Source patient data is provided in four data sets from medical institutions from around the world including Hungarian Institute of Cardiology, University Hospital Switzerland, and medical centers in Virginia, Long Beach, and Cleveland. The outcome variable is diagnosis, while the input variables include patient demographic data, medical test results and exercise stress test results.

We build our models using 4 predictive models, including logistic regression, k-nearest neighbors (KNN), random forest, and support vector machine (SVM). We adopt the following approaches to build the models. Before building the predictive model, we looked at the meaning of each attribute. Then, we cleaned, transformed and removed corrupted data points. Afterwards, we used descriptive statistics and visualizations to better understand the dataset and decided which variables should be included in the models. We build these 4 models using RStudio and compared accuracy and false negative rates to identify the best model.

Based on our study, we find that random forest is the best model for classifying whether a patient has heart disease or not. On the other hand, logistic regression has the worst performance and is not suitable for this dataset. Implementation of random forest model in classifying heart disease can help patients and doctors easily understand what factors impact heart disease the most and assist them to adopt prevention measures.

This study can be further improved if we can obtain larger data sets with more patient attributes, more observation points for the same locations, or broader range of locations. We can also utilize other classification models and machine learning techniques to advance this study.

2. INTRODUCTION

Predictive analytics is a complex yet powerful way to derive value from data. Predictive Analytics processes data using analytics, statistical methods, and machine learning techniques to create a model that is used to forecast future events. It can be applied in a wide range of business markets and academic disciplines to enhance decision making and increase value. The ability to predict an outcome based on numerous input variables leads to powerful insights about customer buying habits, health outcomes, supply chains and manufacturing, economics and investing, as well as many other applications.

The dataset selected for this project contains data of patients with and without heart disease. Heart disease is the leading cause of death in the United States for both men and women, approximately 1 in 4 deaths per year are related to heart disease [1]. Clearly, predictive models for understanding the causes and relationships between bio variables is important to better predict patients who are at high risk of developing the disease.

In this classification problem, 13 attributes out of a total of 75 have been used for analysis. The response value indicating whether or not the patient has heart disease coded as 0 or 1. Patient identifiable data like name and social security number have been stripped from the data, as well as other non-important variables. The objective of this project is to utilize techniques and predictive models to properly classify a patient's heart disease status with a high degree of accuracy.

3. DATA ANALYSIS

The data set for this project is collected and provided by the University of California at Irvine in their machine learning repository [2]. Source patient data is provided in four data sets from medical institutions from around the world including Hungarian Institute of Cardiology, University Hospital Switzerland, and medical centers in Virginia, Long Beach, and Cleveland. Previous machine learning researches have partially cleaned this data, consolidating many of the necessary attributes to a core 13 plus response.

3.1. Issues and Data Cleanup

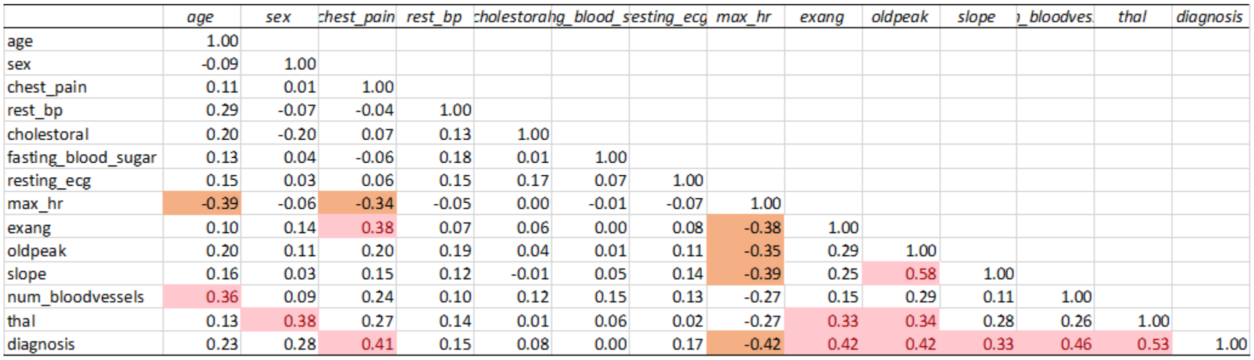
In the original data files, the diagnosis response variable was coded with values from 0-4. A value of 0 refers to the absence of heart disease in the patient, and levels 1 through 4 are increasing levels of disease progression. For this analysis we have recorded the response as 0 or 1 (including values 1-4). Additionally, some columns have been renamed from the original .csv files to clarify what data is included. Unfortunately, some of the data files were corrupted and some rows and fields did not match the data descriptions provided. Therefore, minimal data cleaning has been performed to remove null values and rows that did not match the format of the data description. Due to these issues, confidence falls when combining all data sets together, so we have selected to use the most complete data subset, 'Cleveland', for building and training the predictive models. After cleaning, we have 297 observations and 14 variables. A complete attribute description is provided in Appendix A.

3.2. Descriptive Statistics

Descriptive statistics are useful methods to summarize a given data set and can be presented in various formats such as tables and charts. We included correlation, central tendency, measure of variability and measure of shape for analyzing descriptive statistics. Measure of central tendency describes the central position of a frequency distribution for a group of data. It includes mean, median and mode. Measure of variability describes how spread out a variable is. It includes range, standard deviation, variance, minimum and maximum. Kurtosis and skewness measure the shape of a distribution. We used Excel to perform descriptive statistics. The summarized table is shown below.

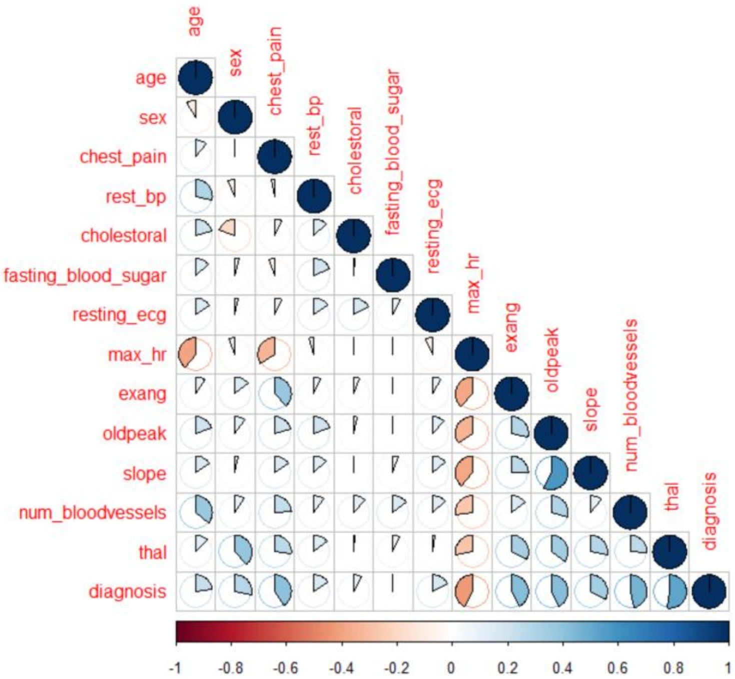
3.2.1. Correlation

We constructed a correlation matrix to identify linear relationships between 2 variables. the red cells are values between 0.3 and 0.7 (indicating a moderate positive linear relationship) while the orange cells values between -0.3 and -0.7 (indicating a moderate negative linear relationship) [3]. The other white cells (except values of 1) imply weak linear relationship. Therefore, no variables have strong correlation and all variables should be included at this stage.



	age	sex	chest_pain	rest_bp	cholestora	ng_blood_s	esting_ecg	max_hr	exang	oldpeak	slope	_bloodves	thal	diagnosis
age	1.00													
sex	-0.09	1.00												
chest_pain	0.11	0.01	1.00											
rest_bp	0.29	-0.07	-0.04	1.00										
cholestoral	0.20	-0.20	0.07	0.13	1.00									
fasting_blood_sugar	0.13	0.04	-0.06	0.18	0.01	1.00								
resting_ecg	0.15	0.03	0.06	0.15	0.17	0.07	1.00							
max_hr	-0.39	-0.06	-0.34	-0.05	0.00	-0.01	-0.07	1.00						
exang	0.10	0.14	0.38	0.07	0.06	0.00	0.08	-0.38	1.00					
oldpeak	0.20	0.11	0.20	0.19	0.04	0.01	0.11	-0.35	0.29	1.00				
slope	0.16	0.03	0.15	0.12	-0.01	0.05	0.14	-0.39	0.25	0.58	1.00			
num_bloodvessels	0.36	0.09	0.24	0.10	0.12	0.15	0.13	-0.27	0.15	0.29	0.11	1.00		
thal	0.13	0.38	0.27	0.14	0.01	0.06	0.02	-0.27	0.33	0.34	0.28	0.26	1.00	
diagnosis	0.23	0.28	0.41	0.15	0.08	0.00	0.17	-0.42	0.42	0.42	0.33	0.46	0.53	1.00

The visualization below shows level of correlation between two variables. We observe that max_hr, chest_pain, exang, oldpeak, slope, num_bloodvessels, and thal have stronger correlation with diagnosis than other predictor variables. Moreover, thal positively correlates with diagnosis the most and max_hr negatively correlates the most.



3.2.2. Measure of central tendency

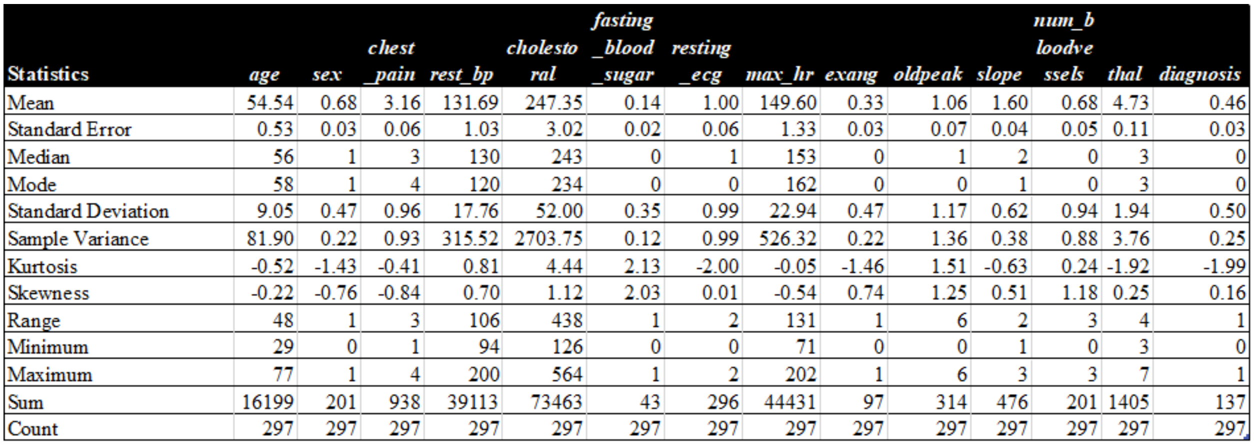
Three main metrics to measure central tendency are mean, median and mode. The average age for the patients is 54.54 years old. The majority of the patients are male and 58 years old. The mode for chest pain type is 4 (asymptomatic) while the mean and the median is type 3 (non-anginal pain). The mean, median and mode of the resting blood pressure are within the normal range, which is from 120/80 to 140/90. The mode of the serum cholesterol is 234 mg/dl. The majority of the fasting blood sugar is below 120 mg/dl. Most of the resting electrocardiographic is normal but the center falls on having ST-T wave abnormality. The average maximum heart rate during patient exercise test is 149.60. Most responses to exercise induced angina are no. ST depression induced by exercise relative to rest has center of 1.06 and mode of 0. Most of the slope of the peak exercise ST segment results is upsloping while the median is flat. The majority of the number of major vessels colored by fluoroscopy is 0. Thalassemia indicating heredity is mostly normal. The response variable, diagnosis, has mode and median of 0 (< 50% diameter narrowing).

3.2.3. Measure of variability

The most common measures of variability are range, variance, standard deviation, minimum and maximum. When the distribution of a dataset has lower variability, the values in the dataset are more consistent. On the other hand, higher variability indicates higher likelihood of extreme values. Wider range, higher sample variance, and higher standard deviation imply higher variability. In this dataset, the variables of rest_bp, cholesterol and max_hr have higher variability and higher chance of outliers.

3.2.4. Measure of shape

A positive value of skewness shows a distribution with a longer tail extending to the right. A negative value of skewness shows a distribution with a longer tail extending to the left. The variables of age, sex, chest_pain, max_hr and exang have negative skewness while the other variables are skewed to the right. Positive kurtosis gives a distinct peak near the center of a distribution, making the distribution narrower. Negative kurtosis gives flatter distribution. The variables of age, sex, chest_pain, resting_ecg, max_hr, exang, slope, thal and diagnosis have flatter distribution while the other variables have positive kurtosis.



Statistics	*age*	*sex*	*chest _pain*	*rest_bp*	*cholesto ral*	*fasting _blood _sugar*	*resting _ecg*	*max_hr*	*exang*	*oldpeak*	*slope*	*num_b loodve ssels*	*thal*	*diagnosis*
Mean	54.54	0.68	3.16	131.69	247.35	0.14	1.00	149.60	0.33	1.06	1.60	0.68	4.73	0.46
Standard Error	0.53	0.03	0.06	1.03	3.02	0.02	0.06	1.33	0.03	0.07	0.04	0.05	0.11	0.03
Median	56	1	3	130	243	0	1	153	0	1	2	0	3	0
Mode	58	1	4	120	234	0	0	162	0	0	1	0	3	0
Standard Deviation	9.05	0.47	0.96	17.76	52.00	0.35	0.99	22.94	0.47	1.17	0.62	0.94	1.94	0.50
Sample Variance	81.90	0.22	0.93	315.52	2703.75	0.12	0.99	526.32	0.22	1.36	0.38	0.88	3.76	0.25
Kurtosis	-0.52	-1.43	-0.41	0.81	4.44	2.13	-2.00	-0.05	-1.46	1.51	-0.63	0.24	-1.92	-1.99
Skewness	-0.22	-0.76	-0.84	0.70	1.12	2.03	0.01	-0.54	0.74	1.25	0.51	1.18	0.25	0.16
Range	48	1	3	106	438	1	2	131	1	6	2	3	4	1
Minimum	29	0	1	94	126	0	0	71	0	0	1	0	3	0
Maximum	77	1	4	200	564	1	2	202	1	6	3	3	7	1
Sum	16199	201	938	39113	73463	43	296	44431	97	314	476	201	1405	137
Count	297	297	297	297	297	297	297	297	297	297	297	297	297	297

3.3. Data Visualization

Figure 1 below shows patient demographic distribution. It is observed that most of the female patients are between 55 and 60 years old and most male patients are between 50-55. Figure 2 shows that in the data set 137 patients have heart disease, of these patients 25 are female and 112 are male. In figure 3, it can be observed that older patients are more likely to be diagnosed with heart disease.

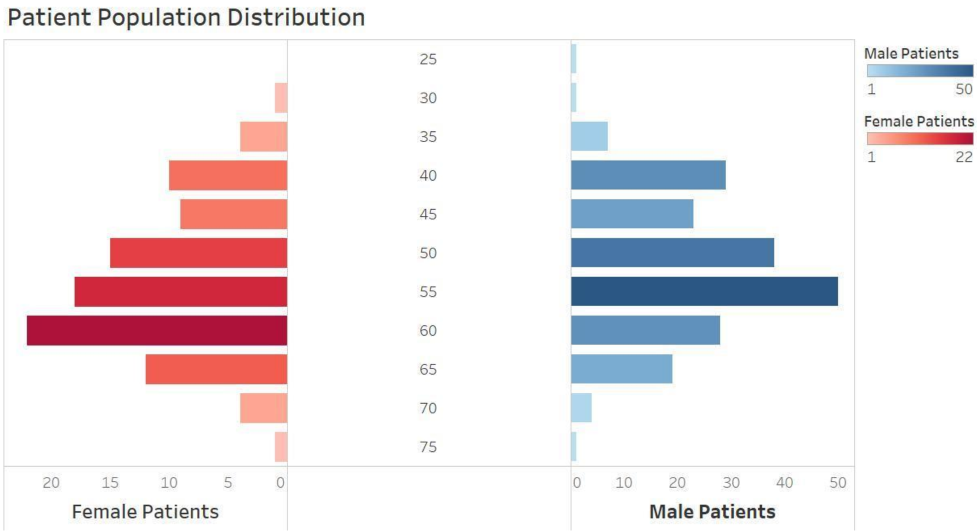


Figure 1: spread of male and female patients by age

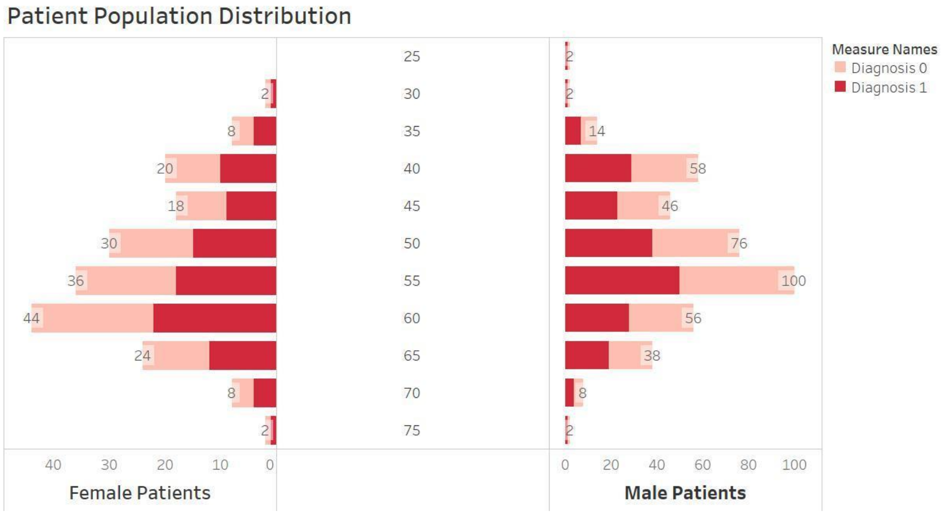


Figure 2: male and female patients with diagnosis.

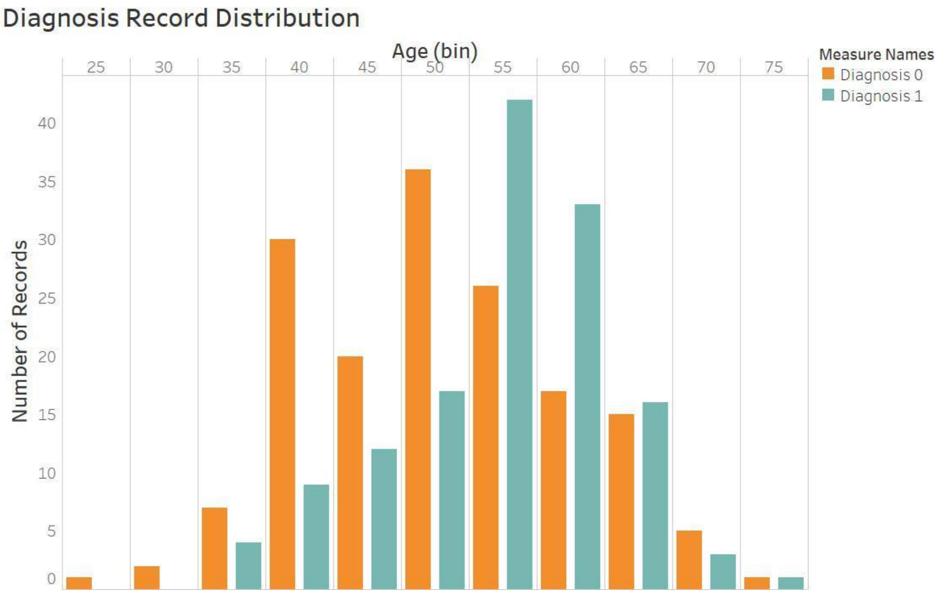


Figure 3: diagnosis spread by age groups.

4. PREDICTIVE MODELS

For all predictive models we started out with a 50/50 split in the observations between train and test, then modified to 70/30 and 60/40 and retested the models to see the effects. Observing the changes in model accuracy and testing different seed values (5, 10, 15, 20, 25) the optimal pairing used in the below models are seed number equal to 25 and training/testing split equal to 60/40.

4.1. Logistic Regression

Variable Selection Method:

Stepwise regression

Stepwise Regression was used to find out only a significant number of parameters out of all the predictors. Steps followed for the same is as shown below:

1. Backward Stepwise Regression

   This process begins with a model containing all the predictors and the eliminate predictors from the model one-at-a-time until only the predictors which contribute to the regression equation remain.

2. Forward Stepwise Regression

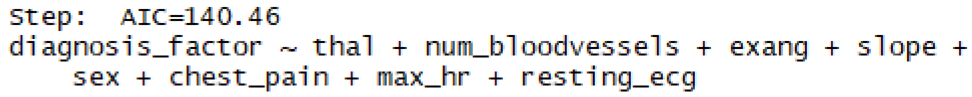
   This process begins with a model containing no predictors and then the predictors are added to the model one-at-a-time until all the predictors are in the model.

3. Backward and Forward Stepwise Regression

This is the hybrid approach in which both the models given in step 1 and 2 are created and passed as parameter to the *step* function as shown below

step(empty_model, direction = "both", scope = formula(full_model))

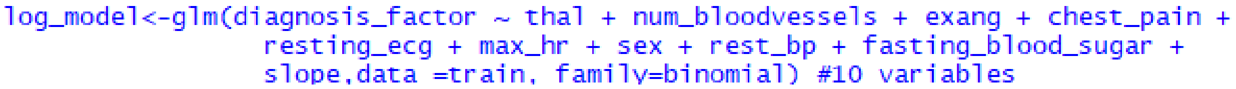
After performing this operation, the significant variables list is suggested as shown in the image below (lowest AIC Value is considered):



```
Step:  AIC=140.46
diagnosis_factor ~ thal + num_bloodvessels + exang + slope +
    sex + chest_pain + max_hr + resting_ecg
```

Model generation

The model equation shown below only contains 10 variables out of the 14 variables. This means that the out of the 14 predictors only 10 variables are significant for prediction of the *diagnosis* variable. The logistic model with only the significant variables is created using these variables.

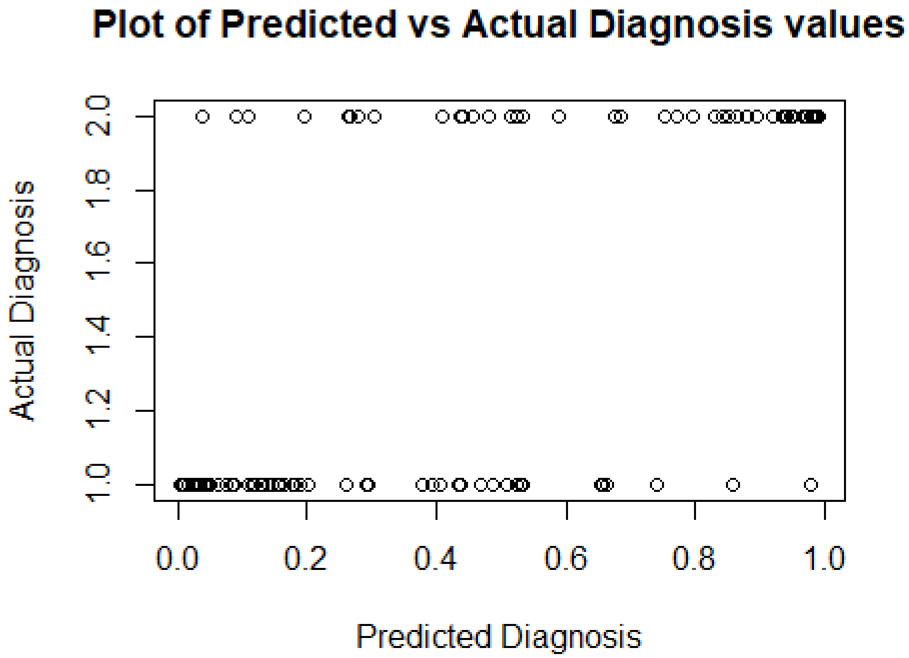


```
log_model<-glm(diagnosis_factor ~ thal + num_bloodvessels + exang + chest_pain +
                resting_ecg + max_hr + sex + rest_bp + fasting_blood_sugar +
                slope,data =train, family=binomial) #10 variables
```

Comparing Actual and the Predicted values

To visualize the plot between the predicted values (values obtained after passing the test data to the logistic regression model) vs actual values (values of the dependent variable of the train data), the plot function has been used. This graph has been plotted below.

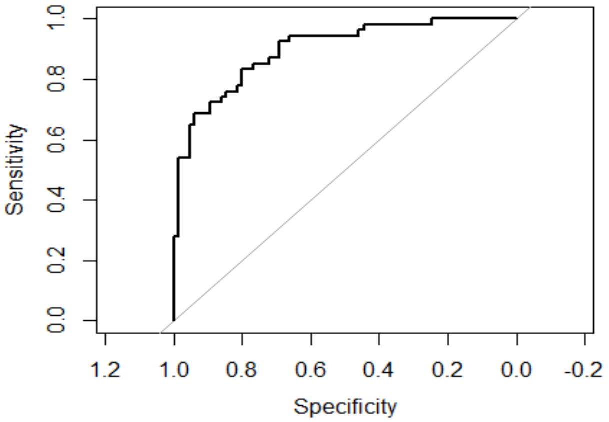
As seen in the graph, the values are either 0 or 1 only. Hence, it is considered a classification problem and can be solved using this model.



Plotting the ROC Curve and finding the AUC Value

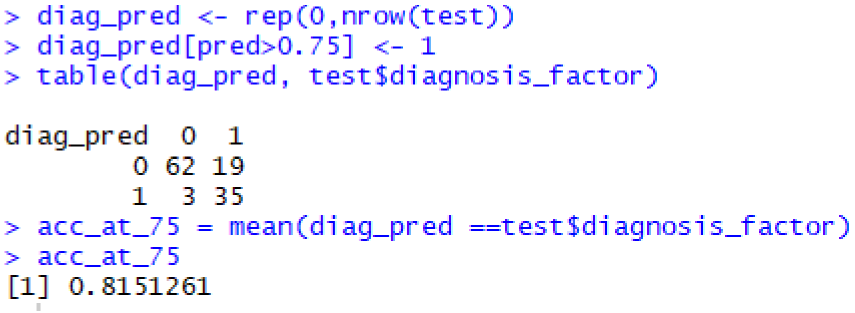
The Receiver Operating Characteristic (ROC) curve is a graphical plot showing the trade-off between Sensitivity (True Positive Rate) and Specificity (True Negative Rate). This graph plots the point by calculating the true positive rate against the false positive rate (if the plot is 1 - specificity instead of only specificity as in this case) for the various confusion matrices obtained at every possible threshold value. Classifier that produce a curve towards the top left corner indicates better performance. The closer the curve is to the 45-degree angle (FPR=TPE), lesser is the accuracy of the model. In conclusion, the ROC curve obtained for this model is pretty good and we can expect the AUC value to be quite decent as well.

Area under the curve: 0.8977



Accuracy of the Model

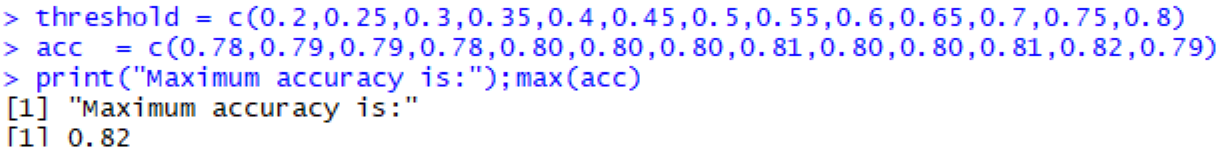
Accuracy corresponds to the value on the threshold value from the ROC curve where the TPR and FPR are significant. Hence, this threshold value would be around 0.75 showing the highest accuracy. Accuracy at 0.75 threshold is 81.5% and the corresponding confusion matrix is as shown below.



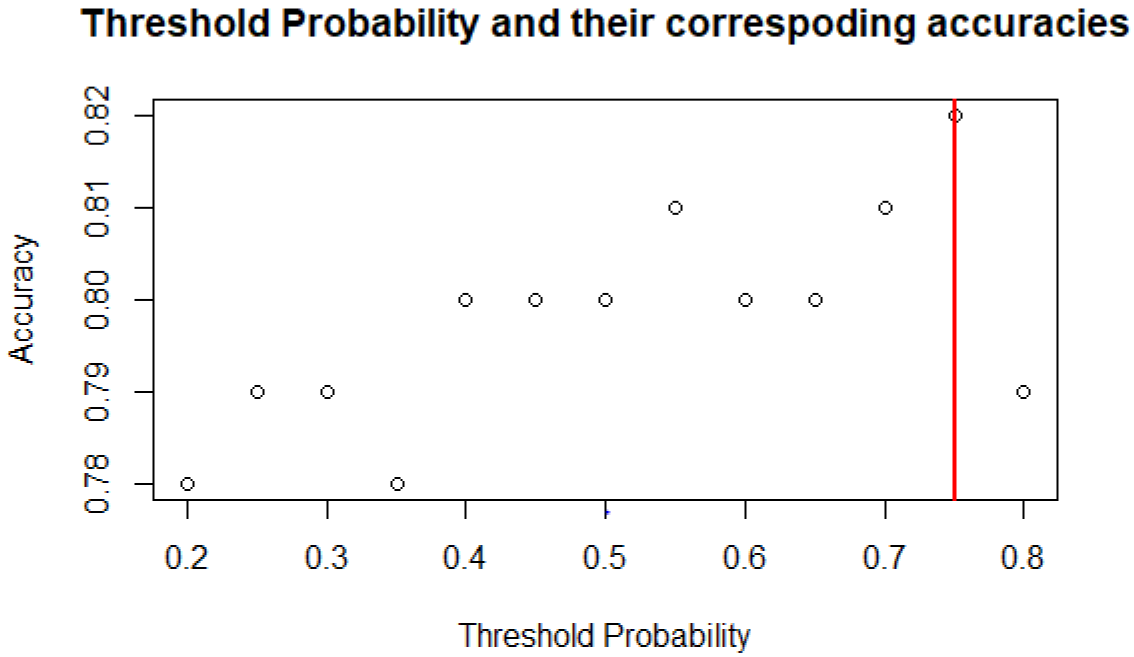
```
> diag_pred <- rep(0,nrow(test))
> diag_pred[pred>0.75] <- 1
> table(diag_pred, test$diagnosis_factor)

diag_pred  0  1
        0 62 19
        1  3 35
> acc_at_75 = mean(diag_pred ==test$diagnosis_factor)
> acc_at_75
[1] 0.8151261
```

Confusion Matrix at 0.75 Threshold Value	**Actual**		
Predicted		**0**	**1**
	0	62	19
	1	3	35

To show the values of accuracy at different threshold value, following value where calculated individually. The image given below shows that value obtained at 0.75 threshold has the highest accuracy. This is depicted by the red line in the plot below.

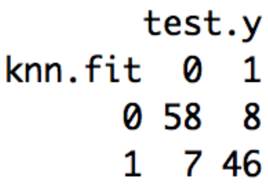


```
> threshold = c(0.2,0.25,0.3,0.35,0.4,0.45,0.5,0.55,0.6,0.65,0.7,0.75,0.8)
> acc   = c(0.78,0.79,0.79,0.78,0.80,0.80,0.80,0.81,0.80,0.80,0.81,0.82,0.79)
> print("Maximum accuracy is:");max(acc)
[1] "Maximum accuracy is:"
[1] 0.82
```

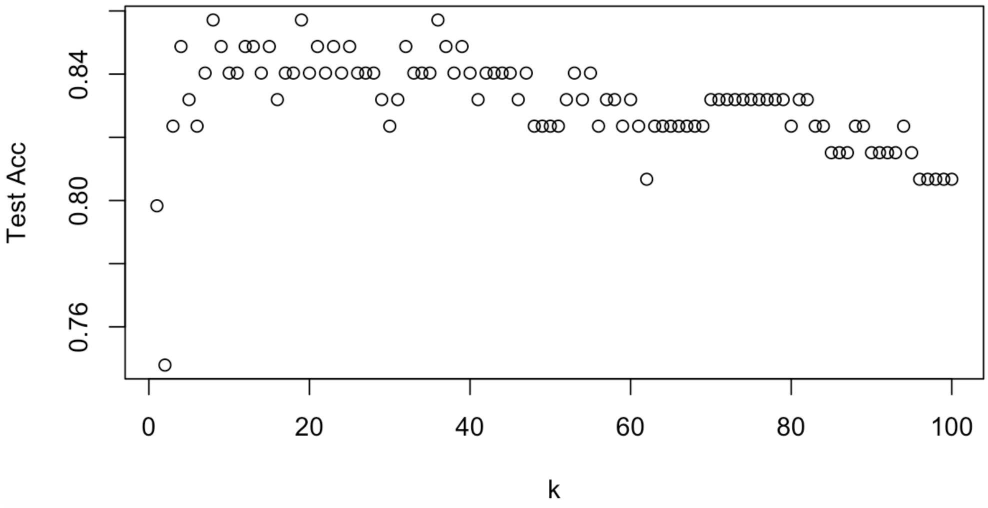


4.2. KNN

A K-Nearest Neighbors (KNN) model was built with a 60/40 training test split on the data and a seed set at 25. The first step of the model was to scale the variables so that all numerical predictors had a value between -1 and 1. KNN classification models use the distances between points to determine how a certain line item should be classified. If predictors are not scaled beforehand, then variables with large ranges than others will distort the model and provide inaccurate results. Once the variables were scaled then a KNN model with K=10, neighbors were built as a baseline. After that the model looped through values of K from 1 to 100 to see which model provided the highest accuracy and lowest false positive rate. As can be seen in the graph below, the value of K=8 created the best accuracy of 87.4% and a false positive rate of 13.2%. Then a confusion matrix was created, as can be seen below, and showed good results that were not skewed to one variable or the other.

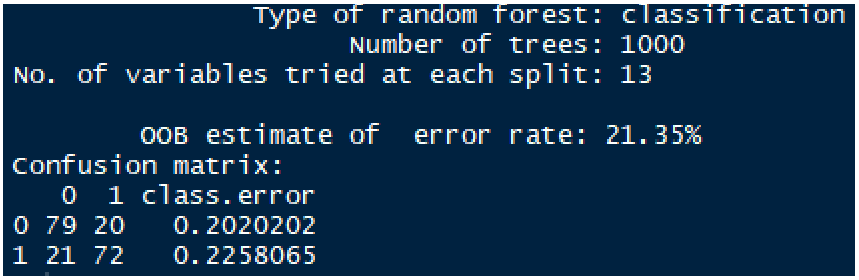


```
       test.y
knn.fit  0  1
      0 58  8
      1  7 46
```

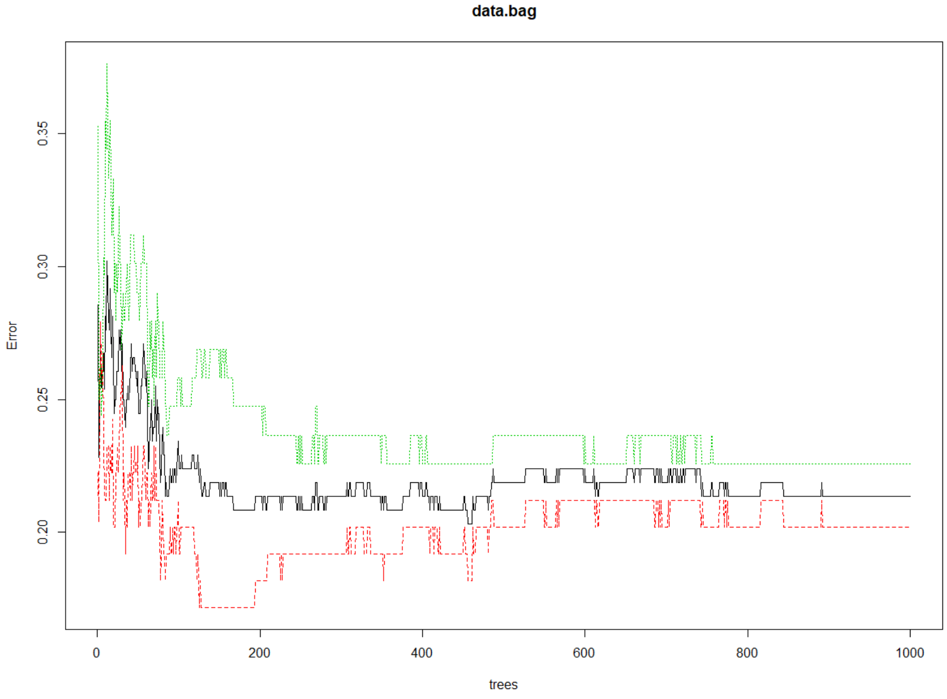


4.3. Random Forest

Bagging

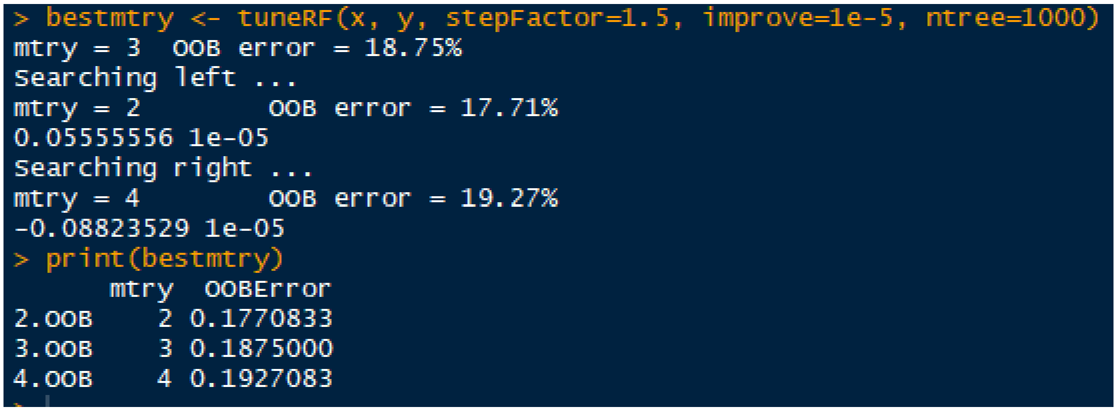


```
                    Type of random forest: classification
                          Number of trees: 1000
No. of variables tried at each split: 13

        OOB estimate of  error rate: 21.35%
Confusion matrix:
   0  1 class.error
0 79 20   0.2020202
1 21 72   0.2258065
```

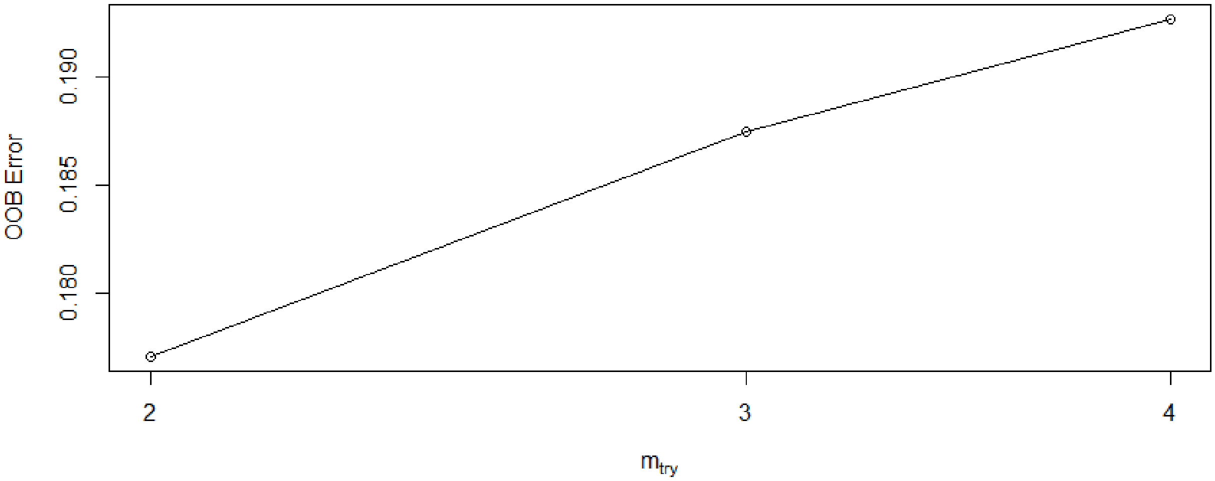


Tuning Approaches for MTRY

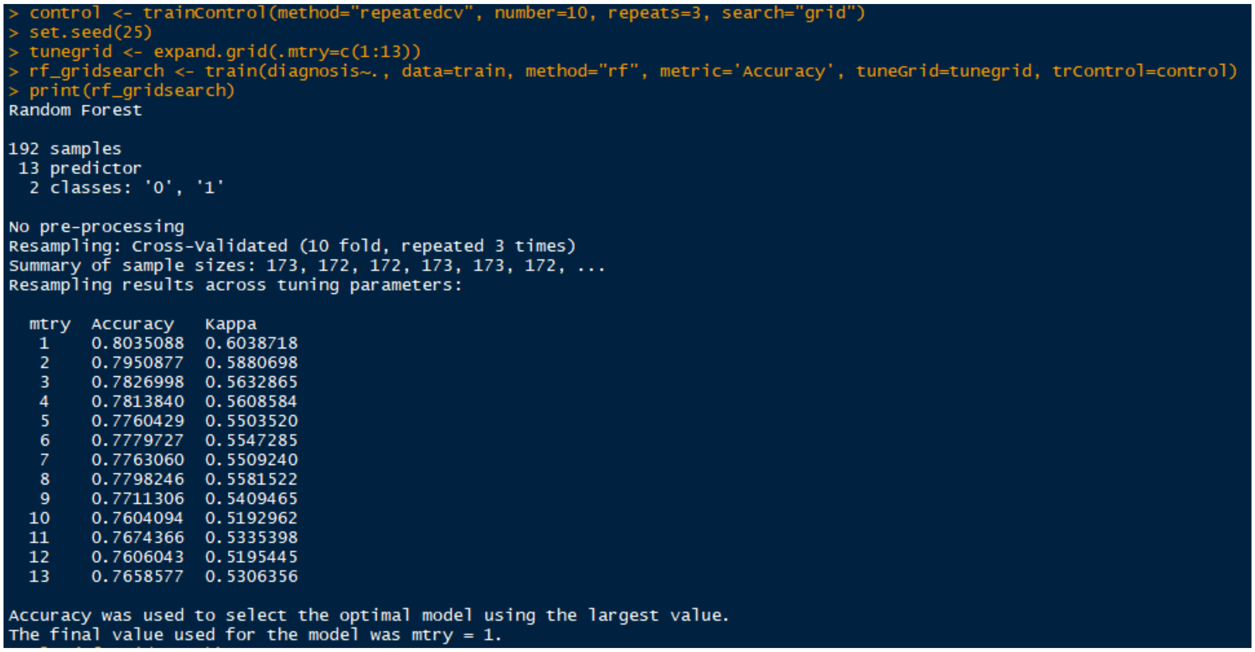
1. using tuneRF (from randomForest package)



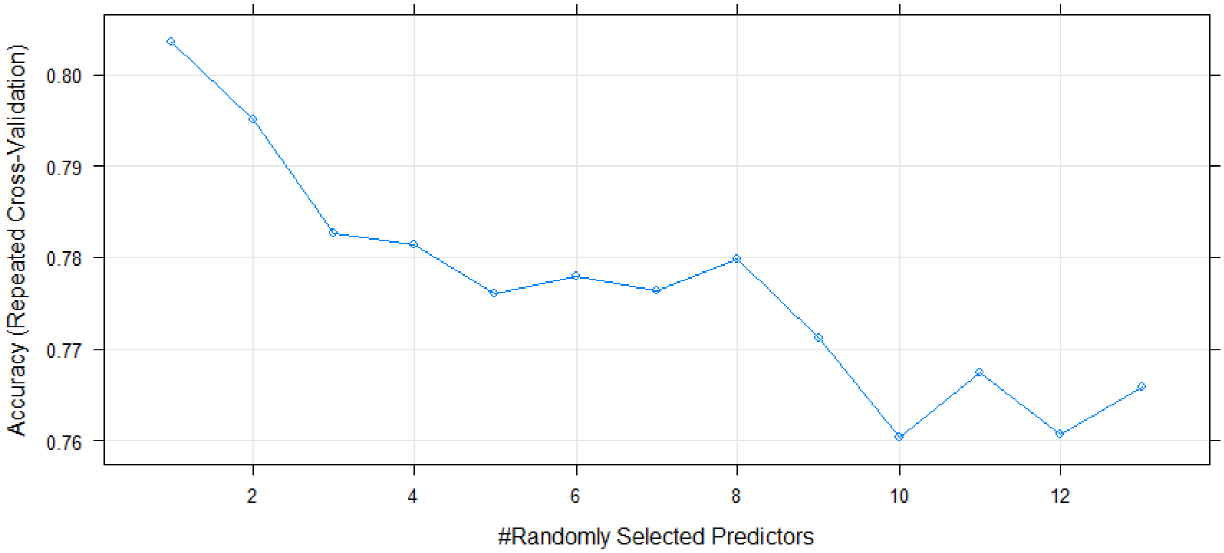
```
> bestmtry <- tuneRF(x, y, stepFactor=1.5, improve=1e-5, ntree=1000)
mtry = 3  OOB error = 18.75%
Searching left ...
mtry = 2        OOB error = 17.71%
0.05555556 1e-05
Searching right ...
mtry = 4        OOB error = 19.27%
-0.08823529 1e-05
> print(bestmtry)
      mtry  OOBError
2.OOB    2 0.1770833
3.OOB    3 0.1875000
4.OOB    4 0.1927083
```



2. using gridSearch from Caret package

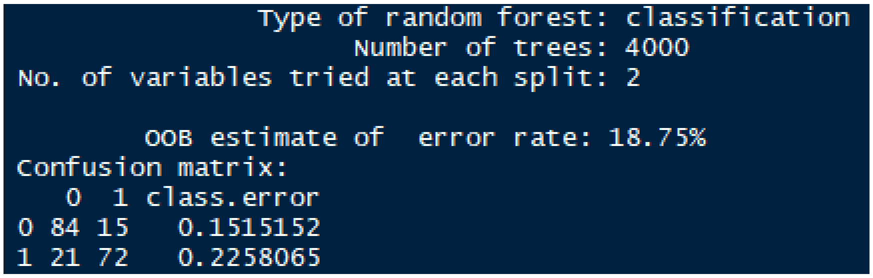


```
> control <- trainControl(method="repeatedcv", number=10, repeats=3, search="grid")
> set.seed(25)
> tunegrid <- expand.grid(.mtry=c(1:13))
> rf_gridsearch <- train(diagnosis~., data=train, method="rf", metric='Accuracy', tuneGrid=tunegrid, trControl=control)
> print(rf_gridsearch)
Random Forest

192 samples
 13 predictor
  2 classes: '0', '1'

No pre-processing
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 173, 172, 172, 173, 173, 172, ...
Resampling results across tuning parameters:

  mtry  Accuracy   Kappa
   1    0.8035088  0.6038718
   2    0.7950877  0.5880698
   3    0.7826998  0.5632865
   4    0.7813840  0.5608584
   5    0.7760429  0.5503520
   6    0.7779727  0.5547285
   7    0.7763060  0.5509240
   8    0.7798246  0.5581522
   9    0.7711306  0.5409465
  10    0.7604094  0.5192962
  11    0.7674366  0.5335398
  12    0.7606043  0.5195445
  13    0.7658577  0.5306356

Accuracy was used to select the optimal model using the largest value.
The final value used for the model was mtry = 1.
```

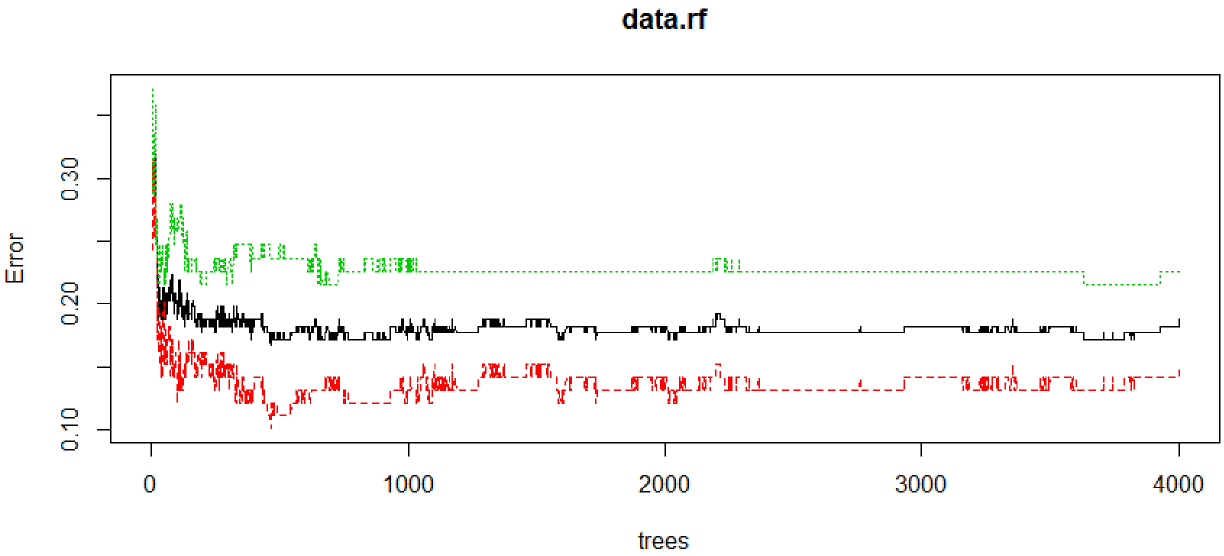


*Looking at the results from the mtry tuning, better results were obtained reducing the mtry to 2. This was validated manually.

*Training the random forest model using MTRY=2 and different ntree values to find optimal pair.

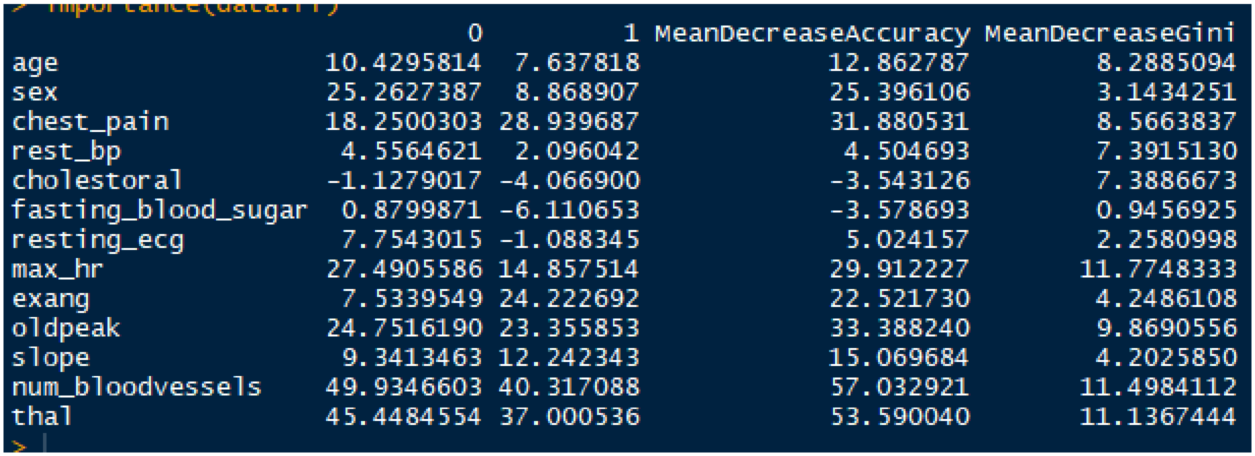


```
                     Type of random forest: classification
                           Number of trees: 4000
No. of variables tried at each split: 2

          OOB estimate of  error rate: 18.75%
Confusion matrix:
   0  1 class.error
0 84 15   0.1515152
1 21 72   0.2258065
```

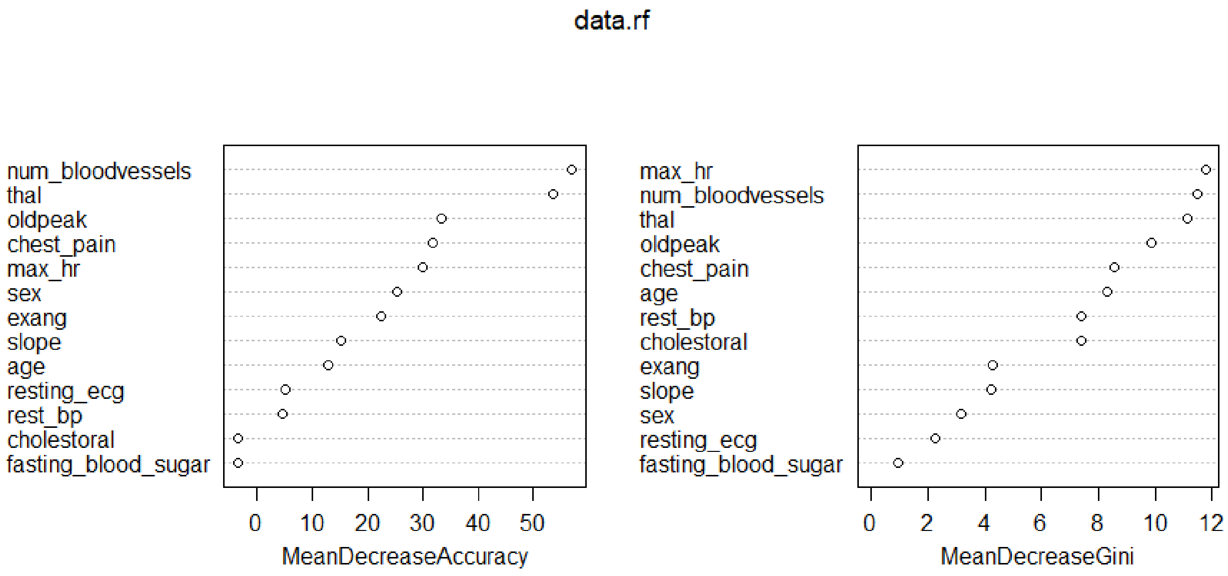


Variable Importance

From the below importance data and plot, it is known the least important variables are fasting_blood_sugar and cholesterol. In order to further refine the accuracy of the predictions, these variables will be removed and the changes observed.



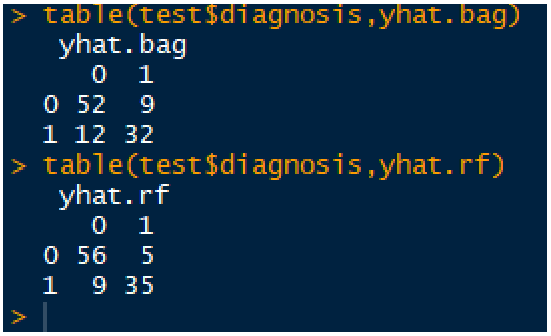
```
> importance(data.rf)
                             0         1 MeanDecreaseAccuracy MeanDecreaseGini
age                 10.4295814  7.637818            12.862787        8.2885094
sex                 25.2627387  8.868907            25.396106        3.1434251
chest_pain          18.2500303 28.939687            31.880531        8.5663837
rest_bp              4.5564621  2.096042             4.504693        7.3915130
cholestoral         -1.1279017 -4.066900            -3.543126        7.3886673
fasting_blood_sugar  0.8799871 -6.110653            -3.578693        0.9456925
resting_ecg          7.7543015 -1.088345             5.024157        2.2580998
max_hr              27.4905586 14.857514            29.912227       11.7748333
exang                7.5339549 24.222692            22.521730        4.2486108
oldpeak             24.7516190 23.355853            33.388240        9.8690556
slope                9.3413463 12.242343            15.069684        4.2025850
num_bloodvessels    49.9346603 40.317088            57.032921       11.4984112
thal                45.4484554 37.000536            53.590040       11.1367444
> 
```



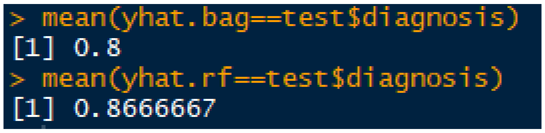
Predictions

yhat.bag [model1] is the prediction using bagging

yhat.rf [model2] is using mtry=2

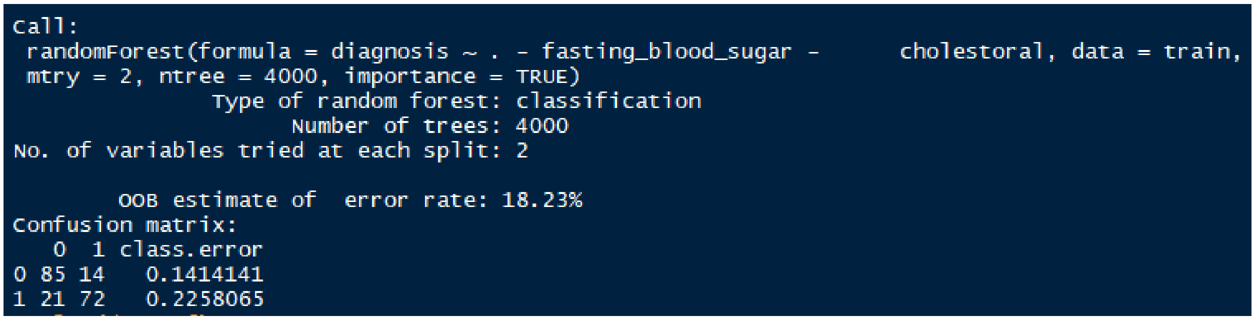


```
> table(test$diagnosis,yhat.bag)
   yhat.bag
     0  1
  0 52  9
  1 12 32
> table(test$diagnosis,yhat.rf)
   yhat.rf
     0  1
  0 56  5
  1  9 35
> 
```



```
> mean(yhat.bag==test$diagnosis)
[1] 0.8
> mean(yhat.rf==test$diagnosis)
[1] 0.8666667
```

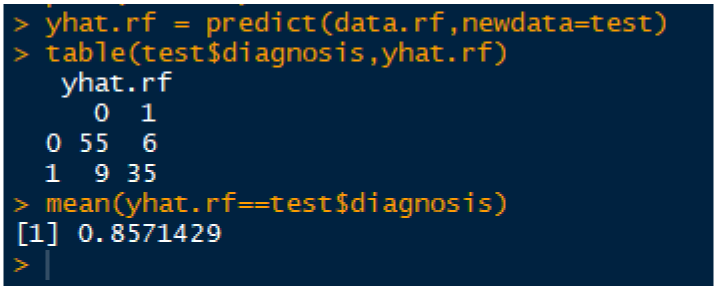
*the random forest model had significantly higher accuracy, which may be further increased by dropping some variables.



```
Call:
 randomForest(formula = diagnosis ~ . - fasting_blood_sugar -      cholestoral, data = train,
 mtry = 2, ntree = 4000, importance = TRUE)
               Type of random forest: classification
                     Number of trees: 4000
No. of variables tried at each split: 2

        OOB estimate of  error rate: 18.23%
Confusion matrix:
   0  1 class.error
0 85 14   0.1414141
1 21 72   0.2258065
```

-dropping fastin_blood_sugar and cholesterol has decreased the OOB estimate from the previous RF model from 18.75% to 18.23%

-using this model [model3] for predictions however, the accuracy falls slightly to 85.7% from 86.7%



```
> yhat.rf = predict(data.rf,newdata=test)
> table(test$diagnosis,yhat.rf)
   yhat.rf
     0  1
  0 55  6
  1  9 35
> mean(yhat.rf==test$diagnosis)
[1] 0.8571429
> 
```

Conclusion

The optimal random forest model [Model2] using Mtry=2, ntree=4000 and dropping no additional variables resulted in an **accuracy of 86.67%**. Model 2 also had a low false negative rate of 12.5%.

Best RF	**Actual**		
Predicted		**0**	**1**
	0	56	5
	1	9	35

4.4. SVM

4.4.1. Seed number and training/testing split

Linear kernel model was used to find the best seed number and the best training/testing split. The following seed numbers were tested: 5, 10, 15, 20, 25 while the training/testing split was held as 50/50 (see table 4.4.1). Seed number of 5 and 25 give the best accuracy. The training/testing split was changed from 50/50, 60/40, 70/30 and 80/20 while seed number was held constant at 25 (see table 4.4.2). All the training/testing splits give close accuracy results. Based on the other models, we decide to use seed number of 25 and training/testing of 60/40.

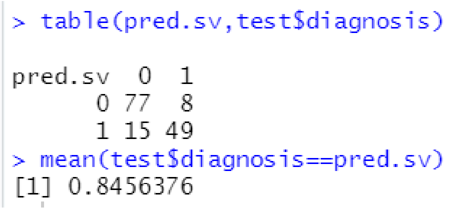
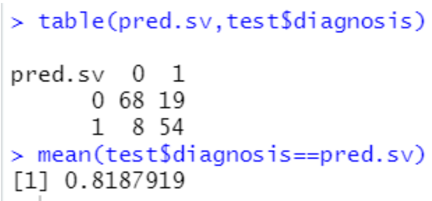
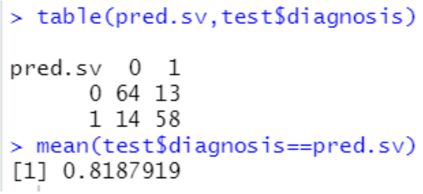
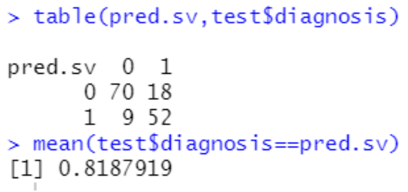
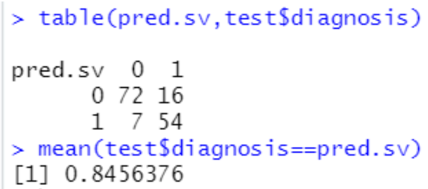
seed = 5	seed = 10
 > table(pred.sv,test$diagnosis) pred.sv 0 1 0 77 8 1 15 49 > mean(test$diagnosis==pred.sv) [1] 0.8456376	 > table(pred.sv,test$diagnosis) pred.sv 0 1 0 68 19 1 8 54 > mean(test$diagnosis==pred.sv) [1] 0.8187919
seed = 15	seed = 20
 > table(pred.sv,test$diagnosis) pred.sv 0 1 0 64 13 1 14 58 > mean(test$diagnosis==pred.sv) [1] 0.8187919	 > table(pred.sv,test$diagnosis) pred.sv 0 1 0 70 18 1 9 52 > mean(test$diagnosis==pred.sv) [1] 0.8187919
seed = 25	
 > table(pred.sv,test$diagnosis) pred.sv 0 1 0 72 16 1 7 54 > mean(test$diagnosis==pred.sv) [1] 0.8456376	

Table 4.4.1. Accuracy variation with different seed numbers

training/testing = 50/50	training/testing = 60/40

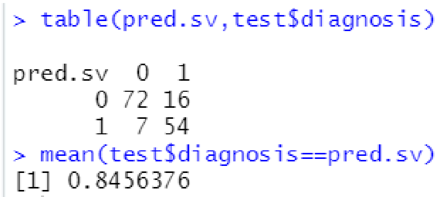
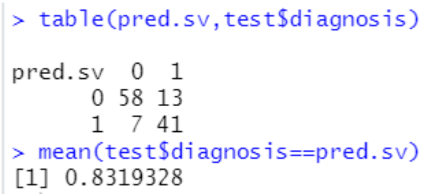
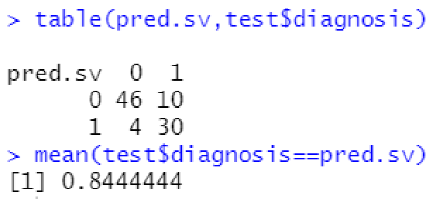
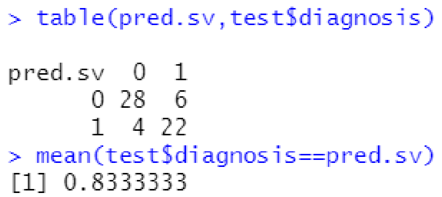
 > table(pred.sv,test$diagnosis) pred.sv 0 1 0 72 16 1 7 54 > mean(test$diagnosis==pred.sv) [1] 0.8456376	 > table(pred.sv,test$diagnosis) pred.sv 0 1 0 58 13 1 7 41 > mean(test$diagnosis==pred.sv) [1] 0.8319328
training/testing = 70/30	training/testing = 80/20
 > table(pred.sv,test$diagnosis) pred.sv 0 1 0 46 10 1 4 30 > mean(test$diagnosis==pred.sv) [1] 0.8444444	 > table(pred.sv,test$diagnosis) pred.sv 0 1 0 28 6 1 4 22 > mean(test$diagnosis==pred.sv) [1] 0.8333333

Table 4.4.2. Accuracy variation with different training/testing split

4.4.2. Linear kernel

The best tuned value for cost is 1. 66 support vectors associated with this model. The accuracy is 83.2%, so the test error rate is 16.8% (see table 4.4.3). The false negative rate is 24.1%.

4.4.3. Radial kernel

The best tuned value for gamma is 0.5. 174 support vectors are associated with this model. The accuracy is 74.8%, so the test error rate is 25.2% (see table 4.4.3). The false negative rate is 16.7%.

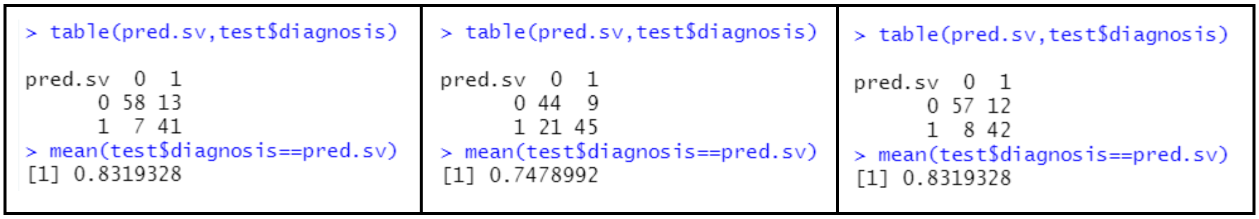
4.4.4. Polynomial kernel

The best tuned value for degree is 3. 116 support vectors are associated with this model. The accuracy is 83.2%, so the test error rate is 16.8% (see table 4.4.3). The false negative rate is 22.2%.

4.4.5. Best kernel model and ROC

The linear model and the polynomial model have the same highest accuracy values (83%). However, the linear model has a slightly higher false negative rate and a slightly higher area under curve (AUC) value than the polynomial model (see figure 4.4.1). Therefore, the polynomial model is the best SVM model because we don’t want to give wrong diagnosis to a patient with heart disease.

Linear kernel	Radial kernel	Polynomial kernel

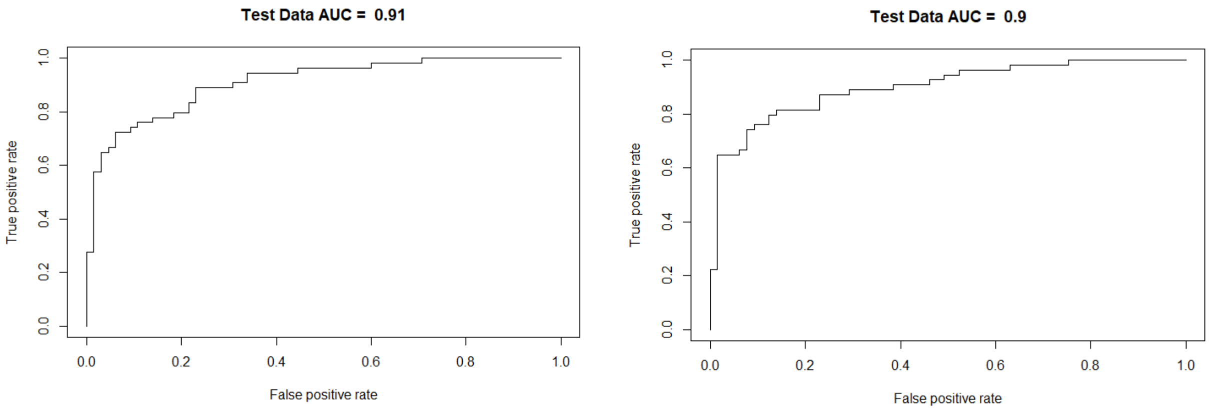


```
> table(pred.sv,test$diagnosis)

pred.sv  0  1
      0 58 13
      1  7 41
> mean(test$diagnosis==pred.sv)
[1] 0.8319328
```

```
> table(pred.sv,test$diagnosis)

pred.sv  0  1
      0 44  9
      1 21 45
> mean(test$diagnosis==pred.sv)
[1] 0.7478992
```

```
> table(pred.sv,test$diagnosis)

pred.sv  0  1
      0 57 12
      1  8 42
> mean(test$diagnosis==pred.sv)
[1] 0.8319328
```

Table 4.4.3. Accuracy variation using different SVM kernels



Linear kernel

Polynomial kernel

Figure 4.4.1. ROC curve for the best kernel model

5. MODEL COMPARISON

We compare the performance of the four models based on two metrics: accuracy and false negative rate. The equations are shown below. We want to have a model with high accuracy in order to perform well on unseen data. Additionally, a model with low false negative rate is desired in order to have a low possibility to give wrong diagnosis to a patient with heart disease. From the table below, we observe that KNN has the highest accuracy while logistic regression has the lowest accuracy. Random forest has the lowest false negative rate and while logistic regress has the highest false negative rate. KNN and random forest have comparable good performance for this data set. If we have to pick one best model, we select random forest because false negative rate is more important. On the other hand, logistic regression is not suitable for classifying heart disease for this data set as it has the worst accuracy and false negative rate.

$$Accuracy = 100 * [(True\ Positive + True\ Negative)/All\ Test\ Observations]$$

*FalseNegRate = 100 * [False Negative / (True Positive + False Negative)]*

	Accuracy	**False Negative Rate**
Logistic Regression	81.5%	35.2%
KNN	87.4%	13.2%
Random Forest	86.7%	12.5%
SVM	83.2%	22.2%

6. CONCLUSION

The objective for this study is to build a predictive model to accurately determine if a patient has heart disease or not. To achieve this, we used a publicly available data collected and provided by the University of California at Irvine in their machine learning repository. We selected 4 predictive models to classify the outcome variable, which are logistic regression, KNN, random forest, and SVM. We used RStudio to build the models and compared them using two important benchmarks (accuracy and false negative rate).

Based on the comparison, we find that random forest and KNN are both good models for classifying whether a patient has heart disease or not. KNN performs slightly better in accuracy, but random forest was more consistent using different observation splits and seed changes. Additionally, the random forest model #2 had the lowest false negative rate which was a critical measure we were looking to reduce based on the implications of misdiagnosing someone. On the other hand, logistic regression has the worst performance and is not suitable for this dataset. Implementation of random forest model in classifying heart disease can help an insurance company to evaluate health insurance price for a potential policy holder or a pharmaceutical company to better develop medicines for heart disease. In addition, it would assist patients and doctors to easily understand what factors impact heart disease the most and assist them to adopt prevention measures.

This study can be further improved if we can obtain larger data sets with more patient attributes, more observation points for the same locations, or broader range of locations. We can also utilize other classification models and machine learning techniques to advance this study.

7. APPENDIX

Data Attribute Description Table

Attribute Name	Attribute Description	Description of Values
age	Age of the patient	In Years
sex	Sex of the patient	0 - Female 1- Male
chest_pain	Chest Pain Type	1 - typical angina 2 - atypical angina 3 - non-anginal pain 4 - asymptomatic
rest_bp	Resting blood pressure	In mm Hg on admission to the hospital
cholesterol	serum cholesterol	In mg/dl
fasting_blood_sugar	Blood Sugar Level when the patient is fasting	Is the fasting blood sugar value > 120 mg/dl 0 - False 1 - True
resting_ecg	Resting electrocardiographic results	0 - normal 1 - having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV) 2 - showing probable or definite left ventricular hypertrophy by Estes' criteria
max_hr	Maximum heart rate achieved during patient exercise test	Data values present in the range of 71 to 202 (inclusive)
exang	Exercise induced angina (Angina is chest pain caused	0 - No 1 - Yes

	by reduced blood flow to heart)	
oldpeak	ST depression induced by exercise relative to rest (ST depression refers to a finding on an electrocardiogram often a sign of myocardial ischemia.)	Data Values present in the range of 0.0 to 6.2 (inclusive)
slope	the slope of the peak exercise ST segment	1 - upsloping 2 - flat 3 - downsloping
num_bloodvessels	number of major vessels colored by fluoroscopy	Values ranges includes 0,1,2,3
thal	thalassemia (heredity)	3 - normal 6 - fixed defect 7 - reversible defect
diagnosis	The predicted response attribute (angiographic disease status)	0 - < 50% diameter narrowing 1 - > 50% diameter narrowing

8. REFERENCES

[1] "Heart Disease Facts & Statistics," Centers for Disease Control and Prevention. [Online]. Available: https://www.cdc.gov/heartdisease/facts.htm. [Accessed: 06-May-2019].

[2] UCI Machine Learning Repository: Heart Disease Data Set. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Heart Disease. [Accessed: 06-May-2019].

[3] The Correlation Coefficient: Definition. [Online]. Available: http://www.dmstat1.com/res/TheCorrelationCoefficientDefined.html. [Accessed: 06-May-2019].