Worcester Polytechnic Institute

DS 502: Statistical Methods for Data Science

A Predictive Model to Predict the Presence of Heart Disease in a Patient

Authors: Ashley Schuliger and Shine Lin Thant

1. Introduction

Heart disease is the current leading cause for death in the United States. In 2021, the American Heart Association claimed that approximately 49.2% of the United States population aged above 20 years has some form of a cardiovascular disease[1]. In 2020, approximately 655,000 deaths occurred as a result of cardiovascular disease in the United States[2]. Deaths from a heart disease most commonly occur from a heart attack or stroke. Early diagnosis can prevent these fatal occurrences since it allows medical professionals to provide the proper treatment to each patient and their specific heart disease. However, diagnosis can be challenging as there are a variety of factors that can contribute to heart disease, including age, chest pain, and the results of a thallium test of the patient. Thus, there is a necessity for a model that can use these factors and their complex relationships to diagnose heart disease in patients. Along with this, there is a need for heart disease awareness in order to prevent patients from acquiring heart disease in the first place. By understanding what may cause a person to have heart disease, we can provide guidelines for leading a healthy life and retroactively prevent heart disease. This form of prevention is important for hospitals all over the United States as it will decrease the number of patients with heart disease and thus, the number of deaths due to cardiovascular disease each year. In the subsequent sections, we propose the use of a predictive machine learning model that will use a variety of attributes about a patient to determine whether or not they have a heart disease. We will tackle two statistical questions dealing with the prediction of heart disease in a patient and identifying the important factors in diagnosing specific heart diseases. We provide the methodology used for building a series of machine learning models to predict heart disease and analyze these factors as well as the validation process that we used in order to obtain each set of models. We then experiment with each of our models in order to find the optimal model to address each statistical question.

2. Methodology

In the following section, we discuss the methodology that we used to process our data as well as the models we developed to answer our statistical questions. We also discuss the validation process that we used in order to evaluate our models in subsequent sections.

a. Data Preprocessing

Our Kaggle dataset consists of approximately 1,026 patient medical records. Each medical record consists of a set of 13 features: age, sex, chest pain type (typical angina, atypical angina, non-anginal asymptomatic, or none), resting blood pressure, serum cholestoral in mg/dl, fasting blood sugar (labeled 1 if greater than 120 mg/dl and 0 otherwise), resting electrocardiographic results (labeled as normal, stt abnormality, lv hypertrophy), maximum heart rate achieved during a thallium test, whether the patient has exercise-induced angina, ST depression induced by exercise relative to rest, the slope of the peak exercise ST segment on an ECG graph, number of major vessels (0-3) colored by flourosopy, and results of a thalium test (normal, fixed defect, reversable defect). Each medical record is also labeled with a 1 or 0 depending on whether or not the patient has a heart disease. In terms of data preprocessing and cleaning, we focused on removing nulls, balancing, creating dummy variables, and splitting our dataset.

Null removal proved to be a point of interest in our project. A “null” value is defined as a value of an empty string, NA, or 0 in a numerical feature with a general higher range of values. In general, removing null values from a dataset is important as the presence of nulls in a dataset can lead to misleading results. For example, when we built a general decision tree on the entire dataset with nulls versus a reduced set with nulls removed, we achieved an entirely different set of important features from one tree to the other. We also found that our testing accuracies improved by 6% when null values were removed from the dataset. When choosing how to remove the nulls from our dataset, we analyzed the distribution of null values across our feature set. Out of 13 features, 2 of our features contained few null values, while 5 of our features contained a significant number of nulls, averaging at approximately 357 null values per feature. The features with a significant number of null values were serum cholesterol in mg/dl, whether the patient has exercise-induced angina, the slope of the peak exercise ST segment on an ECG graph, number of major vessels (0-3) colored by fluoroscopy, and results of a thallium test (normal, fixed defect, reversible defect). Initially, removing all of the records with null values from the dataset seems to be a reasonable tactic. However, since the records with null values do not overlap, the remaining dataset without nulls contains only 308 medical records. This is approximately 70% smaller than our original dataset. As a result, we experimented with two techniques for null removal. The first tactic involves simply removing the five features that contain a significant number of null values. This technique reduces the dimensionality of our dataset while keeping the cardinality of our dataset at 1,026. This is advantageous as larger datasets are less susceptible to overfitting and achieve a higher testing accuracy in general. This technique is limiting, however, since it removes entire features that might be important for predicting heart disease. The second technique we experimented with is removing all of the records with null values. This method allows us to incorporate all 13 features and analyze the relationship

between these features and the response variable. However, as mentioned above, this technique reduces the cardinality of our dataset by 70%, which increases the ratio of dimensionality to cardinality. This may result in our models overfitting to the dataset. We chose to compare these null removal techniques by building two decision trees on both null-removed datasets and comparing the test accuracies achieved. For our null removal technique that removed entire features, we achieved a testing accuracy of 73.51% for an unpruned decision tree and a testing accuracy of 77.84% for a pruned decision tree. In comparison, our null removal technique that reduced the cardinality of the dataset achieved a testing accuracy of 79.71% for an unpruned decision tree and a testing accuracy of 79.71% for a pruned decision tree. This indicates that the null removal technique that reduces the size of the dataset performs better in our application. Based on these results, we used this null removal technique throughout our research in this report.

In order to answer different statistical questions for our analysis, we split the original dataset and created different subsets of the original data to use in different parts of our analysis. Our response variable "num" has 5 different classes in total, namely "0", "1", "2", "3", and "4". For our predictive analysis and model, we converted and combined all of class "2", "3" and "4" to just class "1" to have a simple model that now consists of only two classes for the response variable, with class "0" indicating that a patient does not have heart disease and class "1" indicating that a patient has heart disease.

For our interpretable analysis and model, we filtered the original dataset to remove all of class "0" from our response variable, and just keep the remaining four classes "1", "2", "3", and "4", to indicate severity of the disease, with class "1" being the least severe to class "4" being the most severe.

After splitting the dataset and upon further inspection of the distribution of our data, we discovered that the dataset was highly imbalanced corresponding to the response variable, with 411 observations of class "0", 265 observations of class "1", 109 observations of class "2", 107 observations of class "3", and only 28 observations of class "4", as shown below in the chart. Hence, in order to reduce bias and improve the integrity of our results, we used Smote classification to balance our dataset. After balancing, the dataset now contains 48 observations of class "0", 46 observations of class "1", 26 observations of class "2", 29 observations of class "3", and 20 observations of class "4, which allows for a much more balanced distribution of the response variable classes in our dataset.

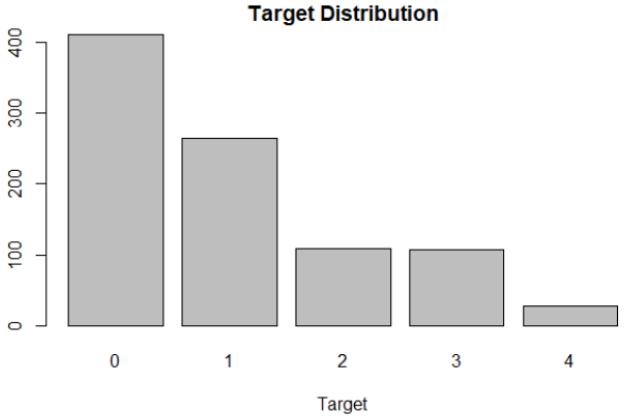

Figure #1: Imbalanced Distribution

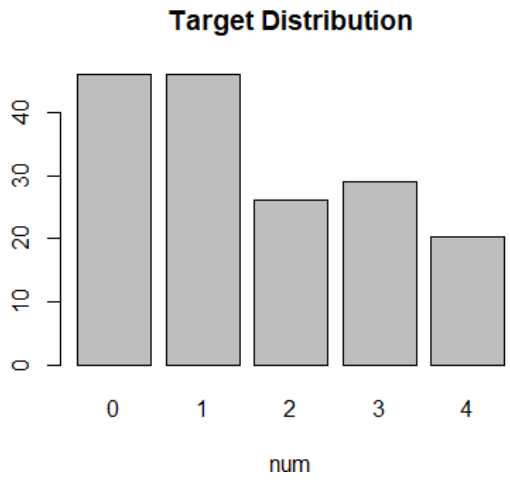

Figure #2: Balanced Distribution

We also preprocessed the data by transforming all qualitative features into dummy variables. Qualitative variables are often converted into numbers in practice, but this encoding places a hierarchical relationship on these variables, which is inaccurate and misleading. Dummy variables are a solution as they create a new column for each value of a qualitative feature and label them as 1 or 0 depending on if the record holds that value or not respectively. This is necessary when using any regression models, such as linear regression and logistic regression. In our application, we transformed sex, chest pain type, fasting blood sugar (labeled 1 if greater than 120 mg/dl and 0 otherwise), resting electrocardiographic results, whether the patient has exercise-induced angina, the slope of the peak exercise ST segment on an ECG graph, number of major vessels (0-3) colored by flourosopy, and results of a thalium test into dummy variables.

b. Model Structures

As mentioned previously, our research has two goals: to predict whether or not a patient has heart disease, and to identify the factors that are important in differentiating between different types of heart disease. In this section, we provide a methodology by which to predict the presence of heart disease in a patient as well as identify the factors that lead to heart disease. We also discuss a separate methodology to analyze various factors and their relationship to different types of heart disease.

i. Statistical Question #1

The problem of predicting heart disease in a patient is defined here as a binary classification problem, where a patient with or without a heart disease is labeled with a 0 or 1 respectively. As previously mentioned, this problem requires a solution that can be used for both prediction as well as interpretability. One way to complete this prediction is using decision trees. Decision trees are a strong choice here as they can both provide an accurate prediction as well as an interpretable model. The interpretability of the model is crucial as its target audience is medical professionals who do not have machine learning experience. A decision tree provides them an easy model by which to make a quick and logical prediction for their patients. Along with this, decision trees are ideal when the features have a non-linear relationship between one another. Since we were unsure about the relationship between our features, we chose to use decision trees in order to compare their performance to models that assume a linear relationship between features. We built a decision tree using all of the predictors in order to create a baseline for our decision tree models. We then pruned down our tree using cross-validation in order to find the optimal number of branches. Pruning down our tree allowed us to obtain a more interpretable model that resulted in more accurate results for our prediction.

However, single decision trees naturally have a high variance. That is, decision trees are easily susceptible to changes in branches and structure depending on the train set. Bagging solves this problem by creating multiple decision trees and calculating the prediction as the average of the trees. This model was a strong choice for our problem since it would reduce the chance of our model overfitting to our dataset. As mentioned previously, we removed all of the null rows from our dataset, which reduced the volume of our data by 70%. Since our final dataset has a high dimensionality and low cardinality, it is susceptible to overfitting, so we chose to build a bagged tree to reduce the potential variance of our model. The limitation that we will find while using bagging is that the results are

not interpretable. Although we can identify important features used for bagging, we cannot look directly at the relationship between our features and the response variable.

Bagging provides a strong model for classification applications. However, the trees created through bagging often have similar structures and important features. Random forests are a solution as they decorrelate the formed trees. This technique uses a separate random subset of predictors to create a tree, thus forming a collection of widely different decision trees from which to extract a final prediction. This wider range of decision trees makes our final model more generalizable and robust, which could result in an overall higher test accuracy than a bagged tree. Thus, we built a random forest using feature subsets of six features to build the intermediate trees. Similarly to bagging, however, random forests do not provide a sufficient amount of interpretability for our problem at hand. Thus, if our random forest produces significantly higher testing accuracies than our other models, we will need to provide another model with better interpretability to complement our random forest model.

We also experimented with using logistic regression to predict whether or not a patient has a heart disease. Logistic regression performs well in binary classification problems as it naturally tends to an output of 0 or 1.We wanted to experiment with logistic regression in comparison to decision trees since it assumes a linear relationship between features. We first experimented with building a logistic regression model using all of the predictors. We expected this model to be overly complex or overfit, so we also built a model using a set of important variables defined by the construction of the bagged tree.

Along with this, we experimented with vector machines and a variety of kernels. Support vectors use two separable features of the dataset in order to build a decision line for the predicted classes. This model is a strong technique for our application since it performs well in binary classification applications. In our application, we chose to build three support vectors on the age and maximum heart rate achieved during a thallium test of each patient. Figure 3 below shows the relationship between these two variables.

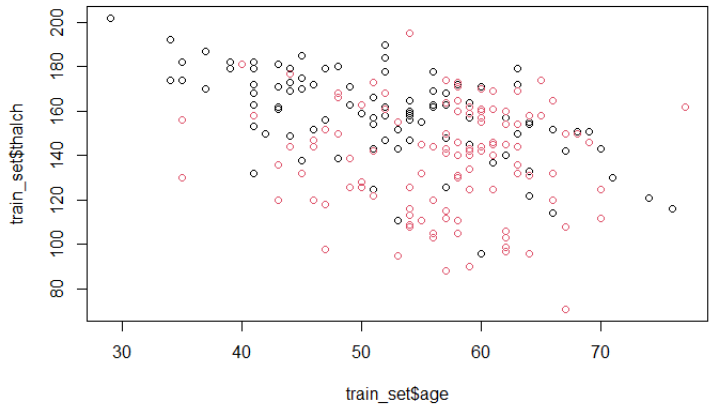

Figure 3: The relationship between age and maximum heart rate achieved during a thallium test.

Although these features are not completely separable, they are the most separable pair of features in our dataset. We do not expect support vector machines to outperform our other models, but we still wanted to test them on our dataset. In terms of kernels, we experimented with a linear, polynomial, and radial decision boundary for separating our two classes.

ii. Statistical Question #2

For the first part of our second statistical question, we aim to make predictions for different classes of heart disease, namely class “1”, “2”, “3”, and “4”. Hence, we used LDA and QDA since we wanted to do classification and we had four classes in our response variable. We ran both LDA and QDA models to understand the linearity of our data and to examine which model has a better performance. We wanted to see how well our model can classify each class of heart disease, and to determine the accuracy of the multi-class classification. Before running LDA and QDA, we first ran a Random Forest model with our entire dataset to determine the most significant variables for our response variable. Then we used the four most significant variables in our LDA and QDA models as predictors to further improve the quality of our analysis.

For the second part of our statistical question, we wanted to understand and examine what factors contribute to and are most significant in each type/class of heart disease. More specifically, these were the type of questions that we wanted to answer:

- What are the important factors in a mild case of heart disease such as classes “1” and “2”?
- What are the most significant variables that contributed in severe cases of heart disease such as classes “3” and “4”?
- Do the same factors contribute to both mild and severe cases, if so, how many, and what are they?

For this purpose, we used Logistic Regression because we were interested in inference and interpretability of the model. We used the same four significant predictors that we previously used in our LDA and QDA models and since we wanted to essentially understand the relationship of the predictors to each individual class, we ran logistic regression on different subsets of the heart disease against healthy patients. Hence, we decided to perform Logistic Regression since our task was reduced to binary classification and Logistic Regression is suitable for that purpose. More specifically we ran five Logistic Regression models in the following manner:

- Model 1: Class “0” vs Class “1”
- Model 2: Class “0” vs Class “2”
- Model 3: Class “0” vs Class “3”
- Model 4: Class “0” vs Class “4”
- Model 5: Class “1” vs Class “4”

After we ran the models, we were able to use the statistics and results to determine how each predictor is related to each class, which will be later discussed in the results section.

c. Validation Process

In order to ensure the accuracy of our chosen model, we split our dataset into a training and testing set. We built the model on the training set, and we used our held-out validation set to calculate our testing accuracy. Since our model is built on the training set, a high training accuracy is usually not indicative of a successful model as some high accuracies can indicate overfitting of the model. The testing accuracy of a model on a set of data it has never been exposed to validates that the model not only predicts on data it has seen but that it also can generalize the prediction strategy to data points outside of the set. Thus, we used the testing accuracy of our models on this test set in order to compare their performance to one another. The model with the highest testing accuracy is considered to be the best model.

3. Evaluation and Results

a. Statistical Question #1

We start by providing the results and structures of our decision tree models, both pruned and unpruned. Our unpruned tree was built using 15 terminal nodes, while our pruned tree was built using 7 terminal nodes. The number of nodes for our pruned tree was derived by performing cross validation. Figure 4 below shows the cross validation results of the number terminal nodes and the error rate of each pruned tree.

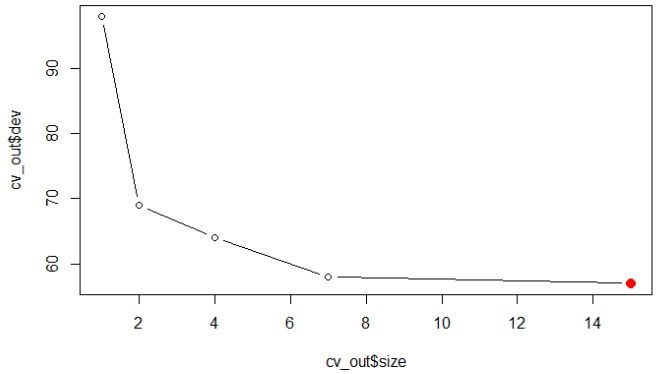

Figure 4: The error rates obtained for each pruned tree through cross validation.

As shown above, the lowest error is achieved at 14 terminal nodes, but the error stops significantly decreasing at 7 terminal nodes. Since a simpler model is often more accurate and interpretable than a complex model, we chose to use the pruned tree with 7 terminal nodes. Figures 5 and 6 below show the structures of our unpruned and pruned decision trees.

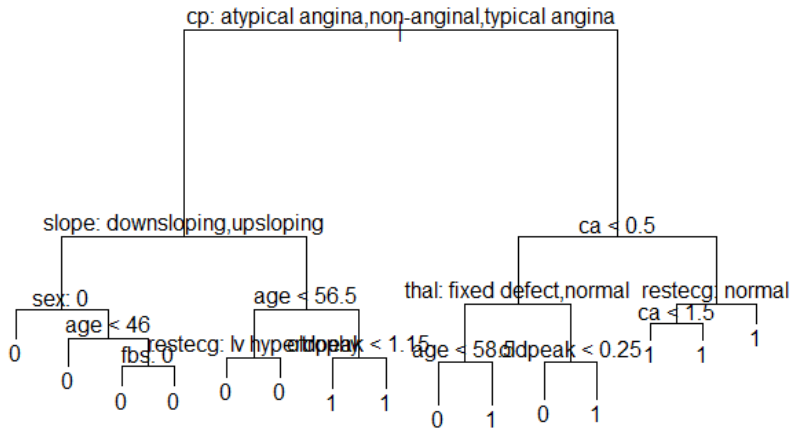

Figure 5: Unpruned decision tree

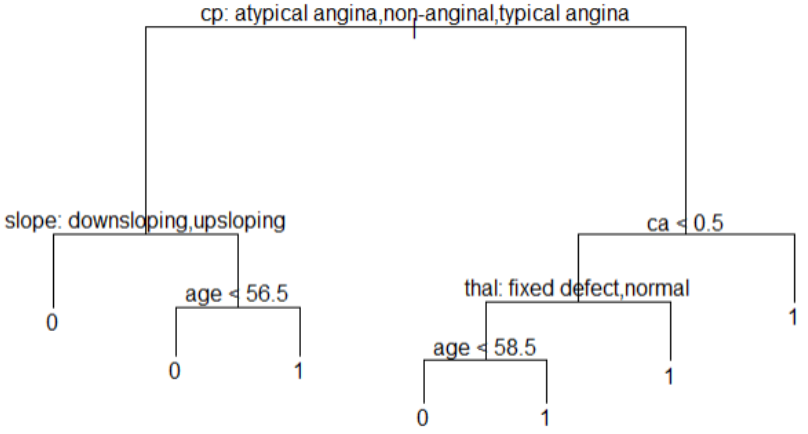

Figure 6: Pruned decision tree

The above unpruned and pruned decision trees both achieved a training accuracy of approximately 87% and a testing accuracy of approximately 79.71%. Although this model provides a strong prediction for heart disease in a patient, the 8% difference between our training and testing accuracies suggests that our model is slightly overfitting to our data. As a result, we chose to experiment with a bagged tree. In our bagging process, we built 500 trees with all of the predictors considered at each split. Since bagging is not interpretable, we do not have a figure for the structure of our bagged tree. Bagging resulted in a testing accuracy of 82.61%, which is an improvement from our pruned tree. We also tested our random forest model to attempt to achieve a high testing accuracy. This random forest model utilized 500 trees and a subset of six features at each split. Similarly, we do not have a figure to portray the structure of our random forests. Our model achieved a testing accuracy of 85.51%, which far outperformed both our pruned decision tree as well as our bagged tree. Thus, for all of our decision tree models, we choose our random forest model as the best for predicting whether or not a patient has a heart disease.

We also collected results for two logistic regression models described in previous sections. The summary of our logistic regression model using all of our predictors is described below.

```
Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)      -3.998424   3.411568  -1.172 0.241189
age               0.003628   0.031867   0.114 0.909349
sex1              1.720169   0.700929   2.454 0.014123 *
trestbps          0.026821   0.014687   1.826 0.067828 .
chol              0.003639   0.004575   0.795 0.426411
fbs1             -0.532014   0.840691  -0.633 0.526845
thalch           -0.012852   0.014878  -0.864 0.387682
oldpeak           0.455402   0.321302   1.417 0.156377
ca                1.754551   0.432851   4.053 5.05e-05 ***
typical_angina1  -3.145135   0.914984  -3.437 0.000587 ***
non_anginal1     -2.651279   0.657588  -4.032 5.53e-05 ***
atypical_angina1 -1.029299   0.660158  -1.559 0.118956
hypertrophy1      0.453463   0.493671   0.919 0.358329
abnormality1      0.322202   2.927572   0.110 0.912364
downsloping1     -1.072630   1.152184  -0.931 0.351877
upsloping1       -1.707725   0.623076  -2.741 0.006129 **
fixed1            0.006149   0.991699   0.006 0.995052
reversable1       1.689022   0.551685   3.062 0.002202 **
---
```

Figure 7: Summary of our logistic model built with all predictors.

Based on the above summary, the significant features used to build this model were typical angina and non-anginal chest pain, the number of major vessels (0-3) colored by fluoroscopy, and a reversible defect thallium test results. The coefficients show the relationship between the features and the response variable. Based on the coefficients above, if a patient has either typical angina or non-anginal chest pain, the likelihood that the patient has a heart disease decreases. As the number of major vessels colored by fluoroscopy increases, however, the likelihood that a patient has heart disease increases. This is logical as often major vessels that appear in fluoroscopy have blood clots or other conditions that indicate a heart disease. Also, if a patient appears to have a reversible effect during a thallium test, this correlates to a higher likelihood of having a heart disease. This makes sense since the reversible effect is when blood flows through a region of the body during exercise but not during rest. This indicates that there may be a blockage in the patient's arteries, which could indicate a heart disease. This logistic regression model achieved a testing accuracy of 86.67% and a testing accuracy of 81.43%. Although this model achieved a lower testing accuracy than our random forests, the interpretability is logical and accurate. As for our logistic regression model with a select number of features, we achieved the following model.

```
Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)      -2.84527    1.37970  -2.062 0.039185 *
typical_angina1  -2.68513    0.79608  -3.373 0.000744 ***
atypical_angina1 -1.12755    0.58350  -1.932 0.053312 .
non_anginal1     -2.40641    0.54809  -4.391 1.13e-05 ***
oldpeak           0.78575    0.25408   3.092 0.001985 **
fixed1            1.17342    0.78742   1.490 0.136171
reversable1       2.23821    0.45743   4.893 9.93e-07 ***
ca                1.29336    0.31835   4.063 4.85e-05 ***
age               0.02788    0.02536   1.099 0.271701
---
```

Figure 8: Summary of our logistic model built with a select set of predictors.

The important features in this model and their relationships to the response variable are the same as those in the previous model. This logistic regression model achieved a testing accuracy of 84.77% and a testing accuracy of 78.57%, which is worse than the performance of our full logistic regression model. Thus, we choose our logistic regression model with all of our predictors as our optimal logistic model.

Before choosing our final model, we also collected results from three support vector machines. When choosing the best support vector machines, we performed cross validation with a set of cost and gamma values and chose the model that produced the lowest error. Figure 9 below shows the results of our support vector classifier with a cost of 0.1.

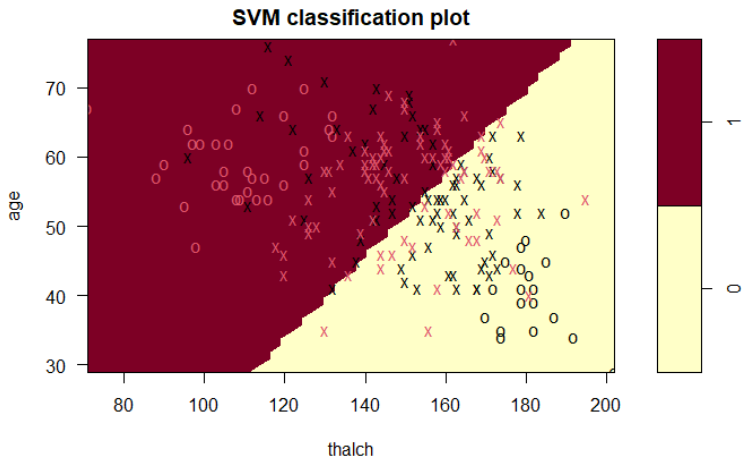

Figure 9: Support vector classifier.

As shown above, the decision boundary does not accurately capture the true separation in the data. This claim is further proved by the model's testing accuracy of 68.12%. This accuracy is significantly lower than our other models, so this model will not be considered. Figure 10 below shows the results of our support vector machine with a polynomial kernel, a cost of 0.1, and a degree of 3.

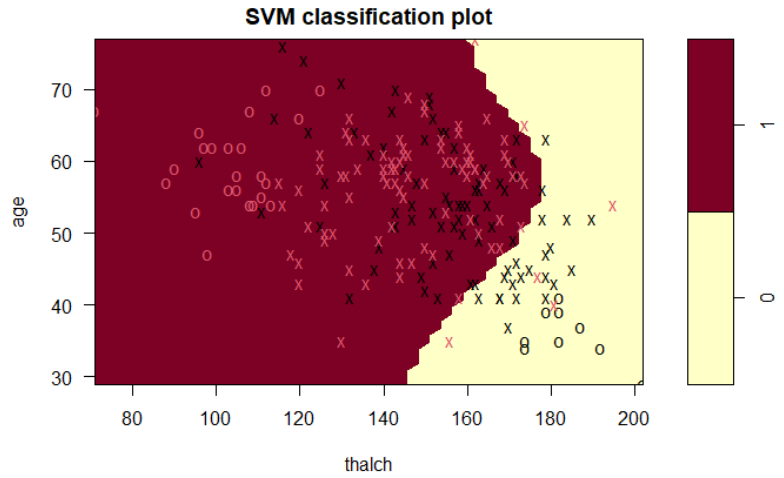

Figure 10: Support vector machine with a polynomial kernel.

This decision boundary also does not accurately model the separation of the classes. This model achieved a testing accuracy of 55.07%, which is only slightly better than a random guess. Thus, this model is not sufficient for predicting heart diseases in patients. Figure 11 below shows the results of our final support vector machine with a radial kernel and a cost of 10.

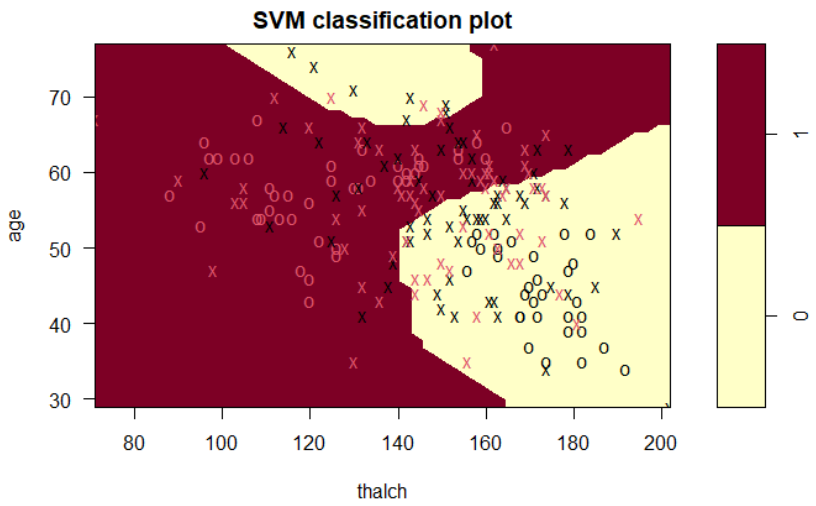

Figure 11: Support vector machine with a radial kernel.

As shown above, this kernel performs better on our dataset and somewhat captures the true decision boundary between these two features. This model achieves a testing accuracy of 73.9%. Although this model does not perform poorly, our previous models outperform it. Thus, support vector machines will not be considered for the optimal model.

Based on the above experiments and results, our random forest model is the optimal model for prediction of whether or not a patient has heart disease as this model achieved the highest test accuracy. Although this model provides a strong prediction, it is not interpretable. Thus, this model must be complemented with our logistic regression model as it achieved an 81% accuracy and achieves interpretability. With a combination of these two models, we can fully address the problem stated in our first statistical question.

b. Statistical Question #2

For the first part of our second statistical question, we firstly ran a Random Forest model to determine which predictors were important, and as shown below in Figure 12.

	IncNodePurity
X	15.186623
id	14.596845
age	23.376346
sex	4.961190
trestbps	17.539591
chol	17.952009
fbs	2.380134
thalch	37.067467
exang	20.554966
oldpeak	59.805420
ca	62.329133
typical_angina	3.485232
non_anginal	3.456912
atypical_angina	2.484647
hypertrophy	3.622562
abnormality	1.391930
downsloping	1.635287
upsloping	8.739954
fixed	1.987086
reversable	26.995920

Figure 12: Random Forest Importance

It is evident that *ca*, *oldpeak*, *thalch*, and *reversable* are the four most significant predictors corresponding to the response variable. Hence, next we ran LDA and QDA models using the aforementioned predictors to determine how well they classify classes “1”, “2”, “3”, and “4”.

```
lda.class  1  2  3  4
        1 13  3  5  2
        2  1  3  1  2
        3  0  1  1  0
        4  1  0  2  1
> mean(lda.class != heart.test$num)
[1] 0.5
```

Figure 13: LDA Test Accuracy

```
qda.class  1  2  3  4
        1 11  3  4  1
        2  4  1  2  4
        3  0  2  3  0
        4  0  1  0  0
> mean(qda.class != heart.test$num)
[1] 0.5833333
```

Figure 14: QDA Test Accuracy

As you can see in the results as provided in the screenshots above, LDA provided a test accuracy of **50%** and QDA provided a test accuracy of **42%**. Evidently, we discovered that LDA performed better for our data than QDA which means there is a certain linearity in the relationship between the predictors and the response. However, as one can tell, these numbers from these models are not as astonishing as those from some of our other models that we discussed earlier in Statistical Question #1, which means LDA and QDA models cannot distinguish between the different classes as accurately as other methods we used such as Random Forest, Bagging etc.

Hence, in the second part of our analysis, we performed binary classification with five Logistic Regression models, using the same four significant variables *ca*, *oldpeak*, *thalch*, and *reversable*, as we previously described in our methodology to determine which variables are significant in each class. Below we have provided the results for our five models.

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.757909   1.525103   0.497    0.619
ca           1.237980   0.300091   4.125 3.70e-05 ***
oldpeak      0.194950   0.222511   0.876    0.381
thalch      -0.018107   0.009628  -1.881    0.010 *
reversable   1.966099   0.420086   4.680 2.87e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 15: Model 1: class “0” vs class “1”

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.80508    1.97090   0.916 0.359739
ca           1.14382    0.34906   3.277 0.001050 **
oldpeak      1.12857    0.31709   3.559 0.000372 ***
thalch      -0.03881    0.01334  -2.910 0.003610 **
reversable   1.98263    0.61211   3.239 0.001199 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 16: Model 2: class “0” vs class “2”

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  5.59830    3.00944   1.860 0.062850 .
ca           2.25529    0.67545   3.339 0.000841 ***
oldpeak      0.76362    0.46193   1.653 0.098307 .
thalch      -0.08015    0.02367  -3.386 0.000710 ***
reversable   5.50404    1.42455   3.864 0.000112 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 17: Model 3: class “0” vs class “3”

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.64927    4.66057  -0.783  0.43362
ca           1.61395    0.63098   2.558  0.01053 *
oldpeak      1.73707    0.56109   3.096  0.00196 **
thalch      -0.01957    0.02851  -0.687  0.49239
reversable   2.34363    1.18132   1.984  0.04727 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 18: Model 4: class “0” vs class “4”

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -9.93555    4.06234  -2.446  0.01445 *
ca           1.46488    0.51085   2.868  0.00414 **
oldpeak      1.05392    0.43873   2.402  0.01630 *
thalch       0.03174    0.02190   1.449  0.14722
reversable   1.51823    0.94298   1.610  0.10739
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 19: Model 5: class “1” vs class “4”

And based on the results provided, we determined the important predictors for each class, which can be summarized as follows:

- For mild cases
  - Case 1: *Ca, Thalch, Reversable*
  - Case 2: *Ca, Oldpeak, Thalch, Reversable*
- For severe cases
  - Case 3: *Ca, Thalch, Reversable*
  - Case 4: *Ca, Oldpeak, Reversable*
- Mild vs Severe (Case 1 vs Case 4)
  - *Ca, Oldpeak*

We found that for all of the cases across our first four models, *ca* and *reversible* are two variables that are always very significant, while the significance of the rest, highlighted in blue, alternates across different cases. This means that the presence of heart disease in a patient can be determined just by looking at the number of major vessels colored by fluoroscopy (*ca*), and the reversible defect thallium test results (*reversible*).

The results for our fifth and final model indicate that *ca* and *oldpeak* are significant in mild vs severe classification of heart disease, which further implies that in order to differentiate between a mild case of heart disease vs a severe one in a patient, it is important to take a look at the number of major vessels showed up during fluoroscopy (*ca*) and ST depression induced by exercise relative to rest (*oldpeak*).

4. Conclusion

In this paper, we proposed a series of machine learning models for both predicting whether or not a patient has a heart disease as well as determining the important factors for diagnosing different types of heart disease. Our final model for our first statistical question consists of a random forest model for prediction and a logistic regression model for interpretation. For our second statistical question, logistic regression models proved to be the optimal way to analyze various factors and their relationship to different types of heart disease. Implementation of our models on our training and testing sets validated the accuracy of our resulting models. The proposed models provide strong predictions and interpretability for whether or not a patient has heart disease and the factors that contribute to different types.

References

1. Virani, S. S., Alonso, A., Aparicio, H. J., Benjamin, E. J., Bittencourt, M. S., Callaway, C. W., ... & American Heart Association Council on Epidemiology and Prevention Statistics Committee and Stroke Statistics Subcommittee. (2021). Heart disease and stroke statistics—2021 update: a report from the American Heart Association. *Circulation*, *143*(8), e254-e743.
2. (2020, September 8). Heart Disease Facts | cdc.gov. Retrieved April 6, 2021, from https://www.cdc.gov/heartdisease/facts.htm
3. (n.d.). EKGs and Exercise Stress Tests | Choosing Wisely. Retrieved April 6, 2021, from https://www.choosingwisely.org/patient-resources/ekgs-and-exercise-stress-tests/
4. (2021, February 9). Heart disease - Diagnosis and treatment .... Retrieved April 6, 2021, from https://www.mayoclinic.org/diseases-conditions/heart-disease/diagnosis-treatment/drc-20353124