# An ML Tool to Detect Heart Disease

CIND 820: Big Data Analytics Project

Robert M. Pineau

941-049-371

Supervisor: Dr. Ceni Babaoglu

Toronto Metropolitan University

# An ML Tool to Detect Heart Disease: Final Report

Robert M. Pineau

941-049-371

Toronto Metropolitan University

CIND 820: Big Data Analytics Project

Supervisor: Dr. Ceni Babaoglu

December 5, 2022

# Contents

## Abstract

Heart Disease is a serious condition that impacts us all. The goal of project is to analyze, and provide the basis for a Machine Learning(ML) based tool to detect Heart Disease. This tool will be created through analysis and training with a popular data-set containing patient records, and tests, related to heart Disease.

As part of this project, the data-set was analyzed, transformed, and re-organized, to make it suitable for use in training and testing of various ML algorithms. Once the data was ready for use, seven different supervised ML algorithms were trained and tested against each other to determine which provided the best initial results. Accuracy's ranging from a low of 68% to just under 88% were achieved in this initial training and testing.

After the initial results, three ML models were chosen for further testing, Logistic Regression(LR), Random Forest(RF), and XGBoost(XGB), as these models repeatedly produce the highest levels of accuracy, and the lowest values for False Negative Rate(FNR), which is important in a medical analysis tool. In order to improve performance of the ML models, a hyper-tuning algorithm was used to fine-tune each model.

Secondary training and testing yielded the highest accuracy, of just over 88%, using a tuned version of RF, with an FNR just under 12%. However, both RF and XGB produced similar results, that statistically show no difference in performance between each other.

Finally some deployment logistics, and other practical concerns are discussed in the conclusion.

## Introduction

Heart disease is the second leading cause of death in Canada. An estimated 2.6 million Canadian adults are living with diagnosed heart disease. Every hour, 14 adults in Canada die due to the disease. However, the Public Health Agency of Canada suggests early detection can help reduce one's risk of heart disease(Public-Health-Agency-of-Canada, 2022).

Heart disease can be detected through various diagnostic means, including blood tests, chest x-ray, Electrocardiogram(ECG), among others(Mayo-Clinic, 2022). However, as with many complex conditions, a skilled physician is required to use their judgment, based on these test results, to determine if heart disease is present or not. To help with the goal of early detection, I decided to explore Machine Learning(ML) and Artificial Intelligence(AI) approaches, specifically using classification to automate the diagnosis of the presence of heart disease.

## Literature Review

In the (Ambrish et al., 2022) study, the UCI[1] data-set (Aha, 1988) was used. The UCI data-set contains 13 main attributes(features), plus the target variable, for (Ambrish et al., 2022) correlation was used to reduce the list of features to six. Logistic Regression(LR) was used to create the ML Model. Of the studies reviewed, only (Ambrish et al., 2022) mentioned techniques for reducing the number of features. [2]

For (Nishadi, 2019), also based on LR, used one of the many subsets of the ongoing Framingham MA. USA, Heart Study (“Framingham Heart Study,” n.d.).

In (Dinesh et al., 2018) the UCI data-set was also used, but ML algorithms explored included LR, Naive Bayes(NB), Random Forest(RF), Support Vector Machine(SVM), and Gradient Boosting(GB). There was no mention of any technique used to reduce the number of attributes.

Similar to (Dinesh et al., 2018), (Khan et al., 2020) also used the UCI data-set , and multiple ML algorithms including LR, KNN, RF, DT, NB, and SVM.

Study (Uyar & İlhan, 2017), another using the UCI data-set proposes using a "genetic algorithm based trained recurrent fuzzy neural networks(GA RFNN)" ML approach.

Study (Latifah et al., 2020), also using a subset of the Framingham MA. USA, Heart Study, made a comparison between LR and RF.

In (Uyar & İlhan, 2017), using GA RFNN, a model accuracy of 97.78% was achieved. Whereas in (Latifah et al., 2020), a model accuracy of 84.4% was achieved using RF, and 85.04%

[1] The UCI Heart Disease data-set is also commonly known as the "Cleveland" data-set .

[2] Some of the studies that utilized the UCI data-set comment on the fact this data-set actually has 76 attributes, but commonly only 13 are used. However, they don’t explain how the 13 common attributes were chosen.

using LR. Study (Ambrish et al., 2022), using LR achieved an accuracy of 87.1%. Similarly (Nishadi, 2019), also using LR alone achieved 86.7%

Finally, in (Khan et al., 2020), accuracy achieved ranged from 79.1% using DT, and 89.0% using RF.

## Data-Set

The data used to build this ML heart disease detection tool is called the "Heart Disease Dataset(Comprehensive)" from the IEEEDataPort(Siddhartha, 2020). This dataset is actually a combination of five separate heart disease datasets, brought together for the purpose of advancing ML and data mining algorithms. This data comes from the following studies:

1. Cleveland.
2. Hungarian.
3. Switzerland.
4. Long Beach VA.
5. Statlog (Heart) Data Set.

(Siddhartha, 2020)

Contained in this dataset is 1190 records, each with 11 attributes, and one class variable. Since this data is a union of different sources, there was clearly some modifications to the data to accommodate the process of merging it. For example, the UCI Heart Disease Data Set(Aha, 1988), which itself is a combination of the data sets from multiple sources, contains 76 attributes.

Further, the process of combining data from multiple sources has its caveats due to differing test procedures, data units, and values used for nominal variables. For example, in the UCI data-set (Aha, 1988) for the attribute "slope" the following nominal values were used:

- Value 0: upsloping
- Value 1: flat

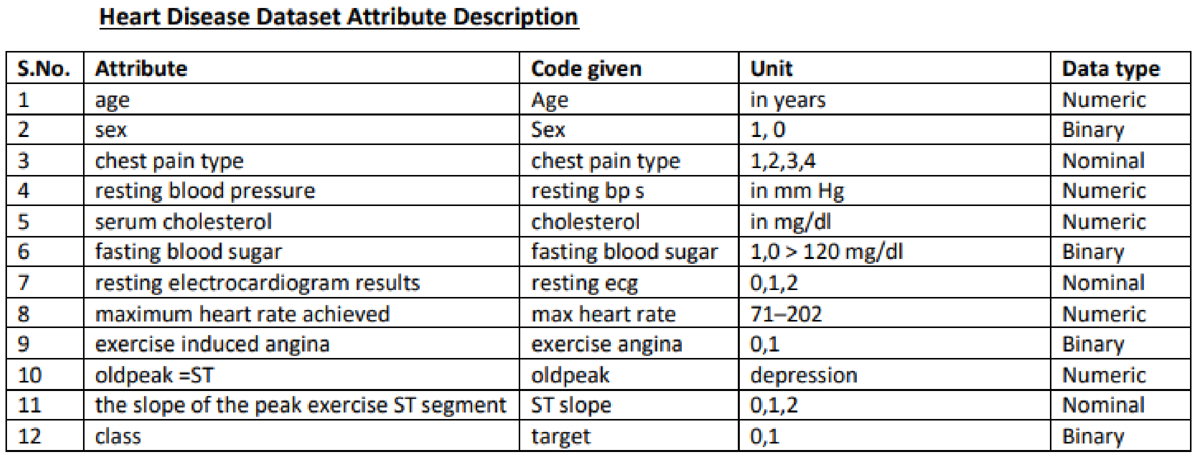
- Value 2: downsloping

However, in the combined "Heart Disease Dataset(Comprehensive)" from the IEEEDataPort(Siddhartha, 2020), this same attribute is as follows:

- Value 1: upsloping
- Value 2: flat
- Value 3: downsloping

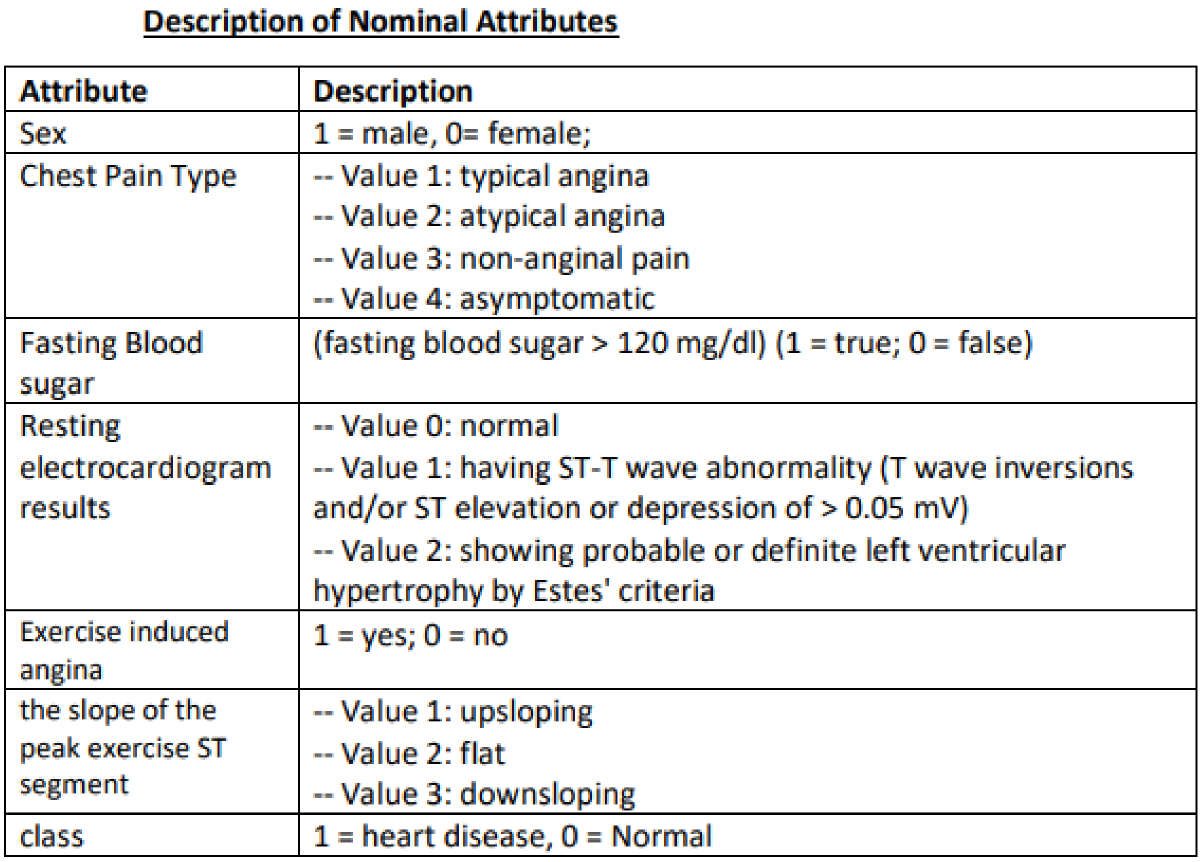
As one can see, the definitions of the nominal values are identical, except the UCI data-set starts at zero, and ends at two, whereas the Heart Disease Dataset(Comprehensive) starts at one and ends at three.

The following tables, from (Siddhartha, 2020) outline the names, units, and ranges, for all of the attributes used for this project:



**Heart Disease Dataset Attribute Description**

| S.No. | Attribute | Code given | Unit | Data type |
|---|---|---|---|---|
| 1 | age | Age | in years | Numeric |
| 2 | sex | Sex | 1, 0 | Binary |
| 3 | chest pain type | chest pain type | 1,2,3,4 | Nominal |
| 4 | resting blood pressure | resting bp s | in mm Hg | Numeric |
| 5 | serum cholesterol | cholesterol | in mg/dl | Numeric |
| 6 | fasting blood sugar | fasting blood sugar | 1,0 > 120 mg/dl | Binary |
| 7 | resting electrocardiogram results | resting ecg | 0,1,2 | Nominal |
| 8 | maximum heart rate achieved | max heart rate | 71–202 | Numeric |
| 9 | exercise induced angina | exercise angina | 0,1 | Binary |
| 10 | oldpeak =ST | oldpeak | depression | Numeric |
| 11 | the slope of the peak exercise ST segment | ST slope | 0,1,2 | Nominal |
| 12 | class | target | 0,1 | Binary |



**Description of Nominal Attributes**

| Attribute | Description |
|---|---|
| Sex | 1 = male, 0= female; |
| Chest Pain Type | -- Value 1: typical angina<br>-- Value 2: atypical angina<br>-- Value 3: non-anginal pain<br>-- Value 4: asymptomatic |
| Fasting Blood sugar | (fasting blood sugar > 120 mg/dl) (1 = true; 0 = false) |
| Resting electrocardiogram results | -- Value 0: normal<br>-- Value 1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV)<br>-- Value 2: showing probable or definite left ventricular hypertrophy by Estes' criteria |
| Exercise induced angina | 1 = yes; 0 = no |
| the slope of the peak exercise ST segment | -- Value 1: upsloping<br>-- Value 2: flat<br>-- Value 3: downsloping |
| class | 1 = heart disease, 0 = Normal |

## Approach

In order to accomplish the goals of this project, four main tasks were done: a) Data cleaning, verification, and characterization, b) Data transformation, c) Modeling (ML training), and d) Model performance comparisons, and final selection.

All programming for each step was done in Python, utilizing proven, and popular, modules where applicable, and custom code where needed. Data cleaning, verification, and characterization, with a combination of standard statistical analysis(mean, medium, quartiles, etc.), visual plots, and various forms of correlation.

Where appropriate, some data transformation was performed, to address outliers, missing values, and class imbalance.

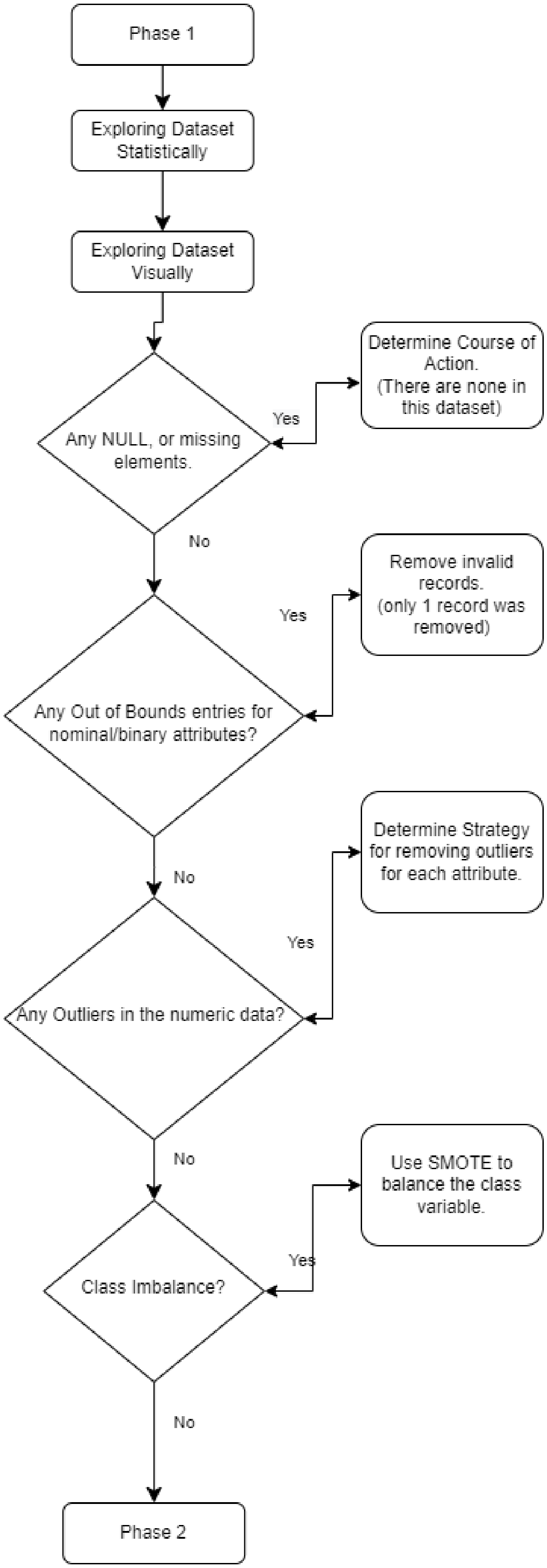
Multiple ML models were used, including Logistic Regression(LR), Decision Tree(DT), Naive Bayes Classification, Random Forest(RF), K-Nearest Neighbor(KNN), Support Vector Machine(SVM), and XGBoost(XGB).

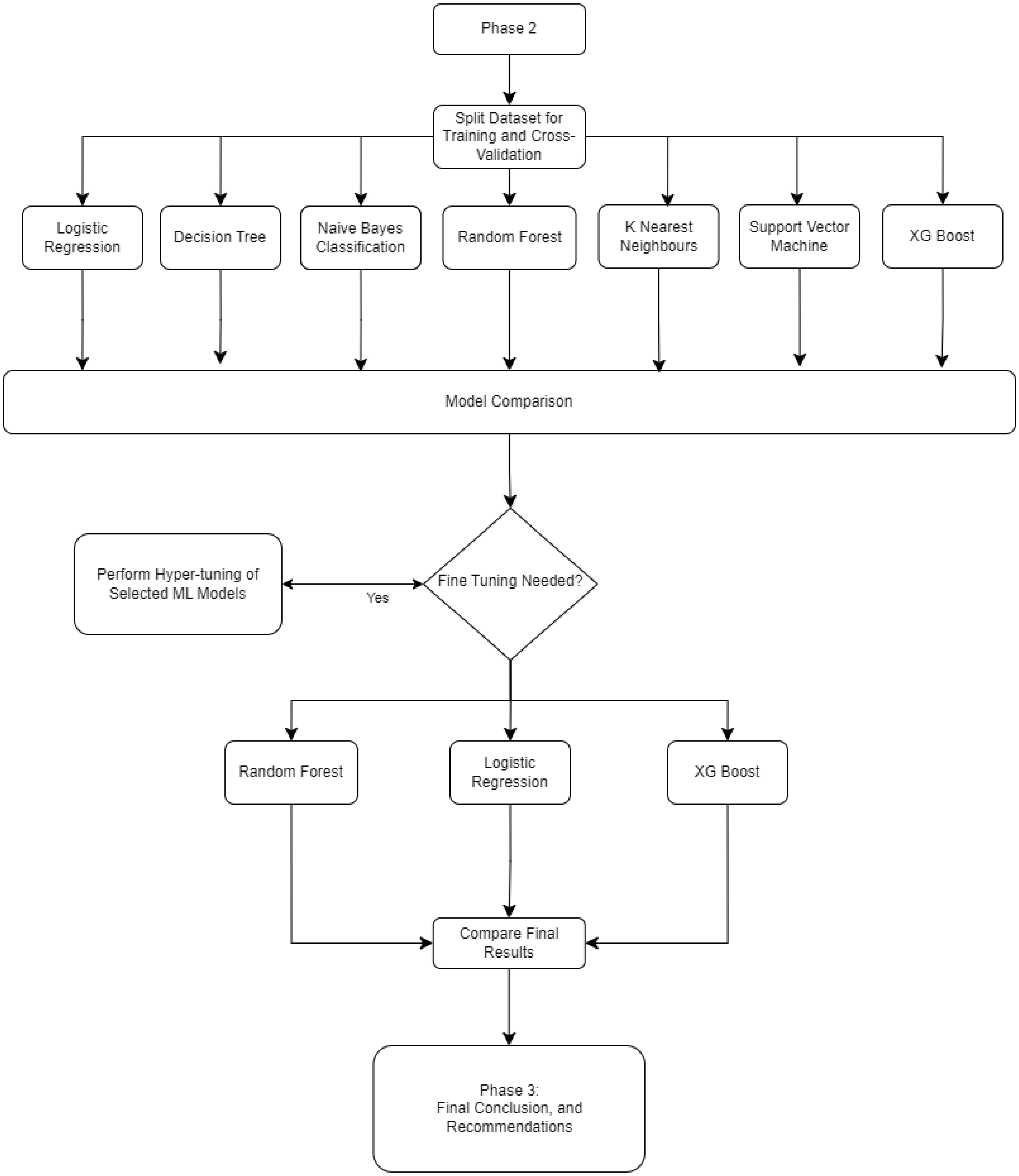
Finally, the results of all ML models used were compared to each other to determine the best model for this problem. Model comparisons were made using standard statistical means including confusion matrix, accuracy, false negative rate etc.

### Software & Tools

The data-set used for this project was downloaded from https://ieee-dataport.org/open-access/heart-disease-dataset-comprehensive

The Python code used for this project can be found at: https://github.com/robert-pineau/CIND-820-Capstone


Phase 1
Exploring Dataset Statistically
Exploring Dataset Visually
Any NULL, or missing elements.
Yes
Determine Course of Action. (There are none in this dataset)
No
Any Out of Bounds entries for nominal/binary attributes?
Yes
Remove invalid records. (only 1 record was removed)
No
Any Outliers in the numeric data?
Yes
Determine Strategy for removing outliers for each attribute.
No
Class Imbalance?
Yes
Use SMOTE to balance the class variable.
No
Phase 2


Phase 2
Split Dataset for Training and Cross-Validation
Logistic Regression
Decision Tree
Naive Bayes Classification
Random Forest
K Nearest Neighbours
Support Vector Machine
XG Boost
Model Comparison
Perform Hyper-tuning of Selected ML Models
Yes
Fine Tuning Needed?
Random Forest
Logistic Regression
XG Boost
Compare Final Results
Phase 3:
Final Conclusion, and Recommendations

# Data Analysis

## Missing, Null, or Duplicate Data

Since this data-set was curated from already existing data-sets it is not surprising that this data is fairly complete. However, to be sure the data was checked for missing, and NULL entries, to ensure there were no mistakes made in the process of combining sources. However, no missing, or NULL entries were found.

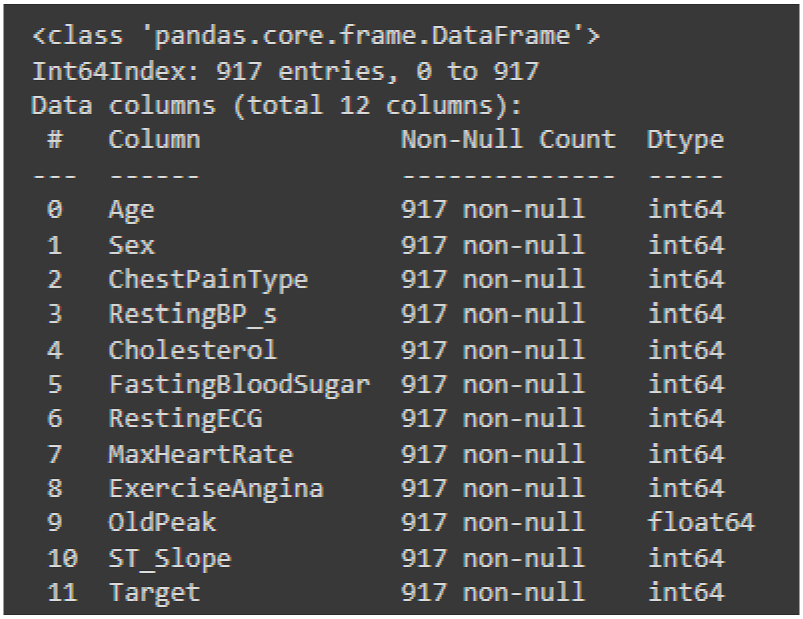
However, it was found that 272 rows were duplicate of others in the data. These were removed, as they were not needed, and would have had a detrimental impact to the modeling/training phase later.

## Out-Of-Bound Data for Nominal & Binary Attributes

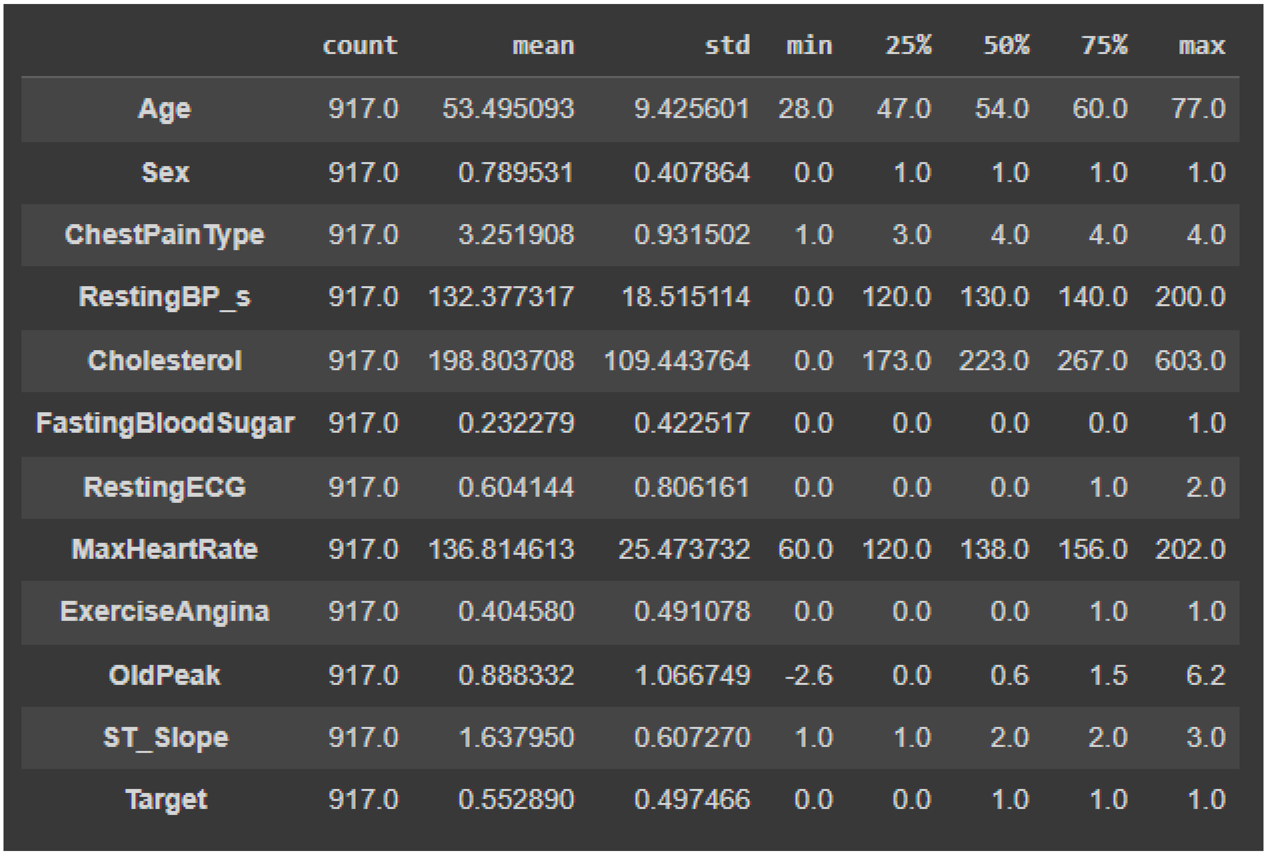
For the nominal, and binary attributes, there is a defined range of integer values the the data is allowed to contain. Since this range is known, every row of data was analyzed to ensure these ranges were adhered to. As a result of this analysis one row of data was found to have an out-of-bounds entry for the attribute ’ST_Slope’ Since only a single row of data was found to be out-of-bounds it was simply discarded from the data-set .

## Attribute Summary

This data-set originally contained 1190 rows. However, after removing 272 duplicate rows, and one entry containing an out-of-bounds nominal value, only 917 rows remained.



```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 917 entries, 0 to 917
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Age                917 non-null    int64
 1   Sex                917 non-null    int64
 2   ChestPainType      917 non-null    int64
 3   RestingBP_s        917 non-null    int64
 4   Cholesterol        917 non-null    int64
 5   FastingBloodSugar  917 non-null    int64
 6   RestingECG         917 non-null    int64
 7   MaxHeartRate       917 non-null    int64
 8   ExerciseAngina     917 non-null    int64
 9   OldPeak            917 non-null    float64
 10  ST_Slope           917 non-null    int64
 11  Target             917 non-null    int64
```



| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| **Age** | 917.0 | 53.495093 | 9.425601 | 28.0 | 47.0 | 54.0 | 60.0 | 77.0 |
| **Sex** | 917.0 | 0.789531 | 0.407864 | 0.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| **ChestPainType** | 917.0 | 3.251908 | 0.931502 | 1.0 | 3.0 | 4.0 | 4.0 | 4.0 |
| **RestingBP_s** | 917.0 | 132.377317 | 18.515114 | 0.0 | 120.0 | 130.0 | 140.0 | 200.0 |
| **Cholesterol** | 917.0 | 198.803708 | 109.443764 | 0.0 | 173.0 | 223.0 | 267.0 | 603.0 |
| **FastingBloodSugar** | 917.0 | 0.232279 | 0.422517 | 0.0 | 0.0 | 0.0 | 0.0 | 1.0 |
| **RestingECG** | 917.0 | 0.604144 | 0.806161 | 0.0 | 0.0 | 0.0 | 1.0 | 2.0 |
| **MaxHeartRate** | 917.0 | 136.814613 | 25.473732 | 60.0 | 120.0 | 138.0 | 156.0 | 202.0 |
| **ExerciseAngina** | 917.0 | 0.404580 | 0.491078 | 0.0 | 0.0 | 0.0 | 1.0 | 1.0 |
| **OldPeak** | 917.0 | 0.888332 | 1.066749 | -2.6 | 0.0 | 0.6 | 1.5 | 6.2 |
| **ST_Slope** | 917.0 | 1.637950 | 0.607270 | 1.0 | 1.0 | 2.0 | 2.0 | 3.0 |
| **Target** | 917.0 | 0.552890 | 0.497466 | 0.0 | 0.0 | 1.0 | 1.0 | 1.0 |

## Data Visualization

In order to visualize the data properly this task was divided up based on the data types: There are two main types of data in this dataset:

1. Numeric Data
2. Nominal & Binary Data

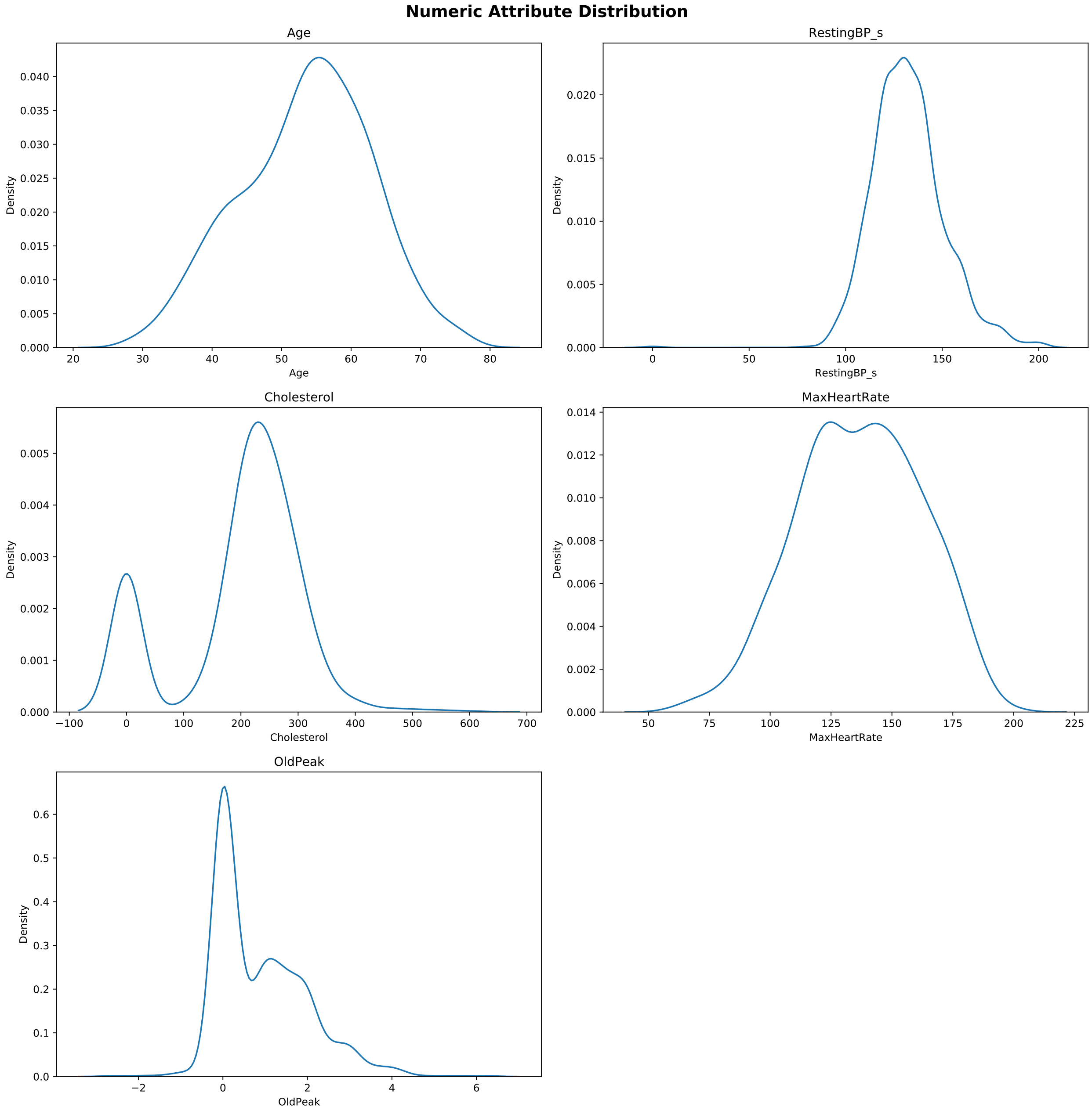
Further, the data was viewed in three different contexts:

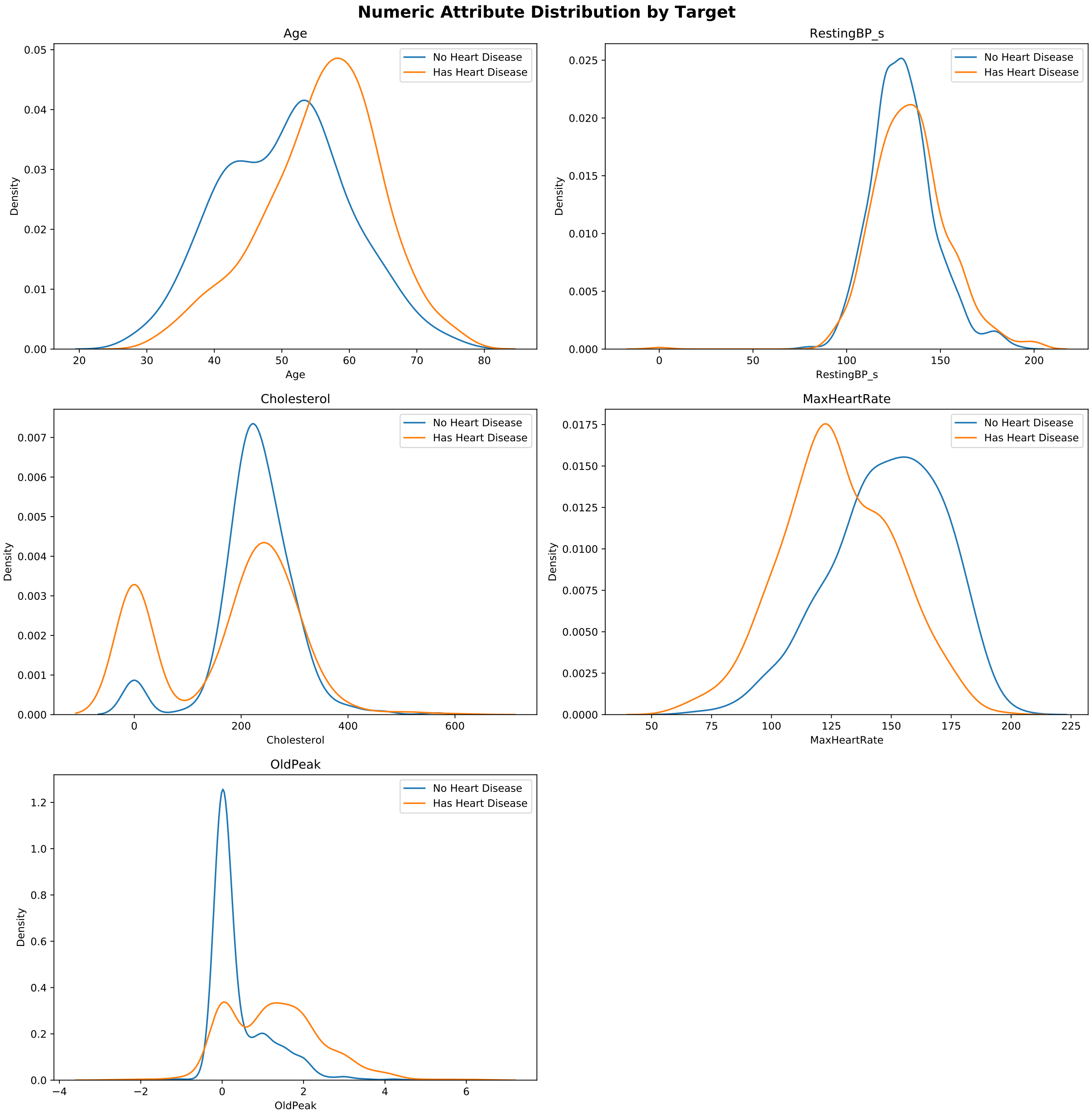
1. Each attribute for the whole dataset.
2. Each attribute subset based on the target variable.
3. Each attribute subset based on both the target variable and the sex of the individual the data was obtained.

Numeric data is visualized using a basic line graph, whereas the nominal and binary data is visualized using a series of pie plots.

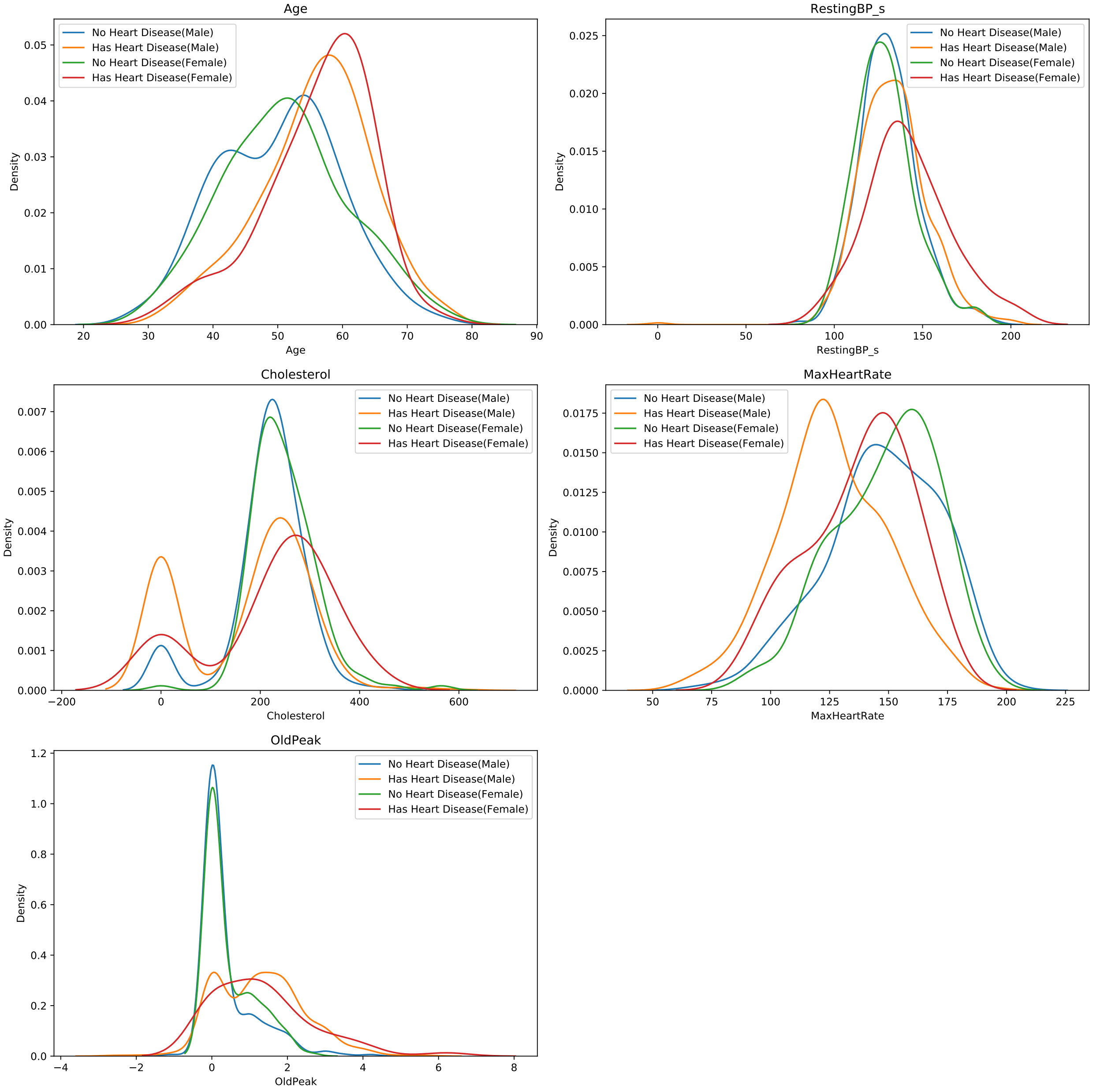
### *Numeric Data*

There are five numeric attributes Age, RestingBP_s, Cholesterol, MaxHeartRate, and OldPeak. From the graphs, the data mostly looks to be normally distributed, as in fairly mound shaped, with the exception of a fairly pronounced secondary peak for Cholesterol at value zero.




Numeric Attribute Distribution by Target
Age
No Heart Disease
Has Heart Disease
Density
0.05
0.04
0.03
0.02
0.01
0.00
20
30
40
50
60
70
80
Age
RestingBP_s
No Heart Disease
Has Heart Disease
Density
0.025
0.020
0.015
0.010
0.005
0.000
0
50
100
150
200
RestingBP_s
Cholesterol
No Heart Disease
Has Heart Disease
Density
0.007
0.006
0.005
0.004
0.003
0.002
0.001
0.000
0
200
400
600
Cholesterol
MaxHeartRate
No Heart Disease
Has Heart Disease
Density
0.0175
0.0150
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
50
75
100
125
150
175
200
225
MaxHeartRate
OldPeak
No Heart Disease
Has Heart Disease
Density
1.2
1.0
0.8
0.6
0.4
0.2
0.0
−4
−2
0
2
4
6
OldPeak

**Numeric Attribute Distribution, by Target and Sex**



### *Nominal & Binary Data*

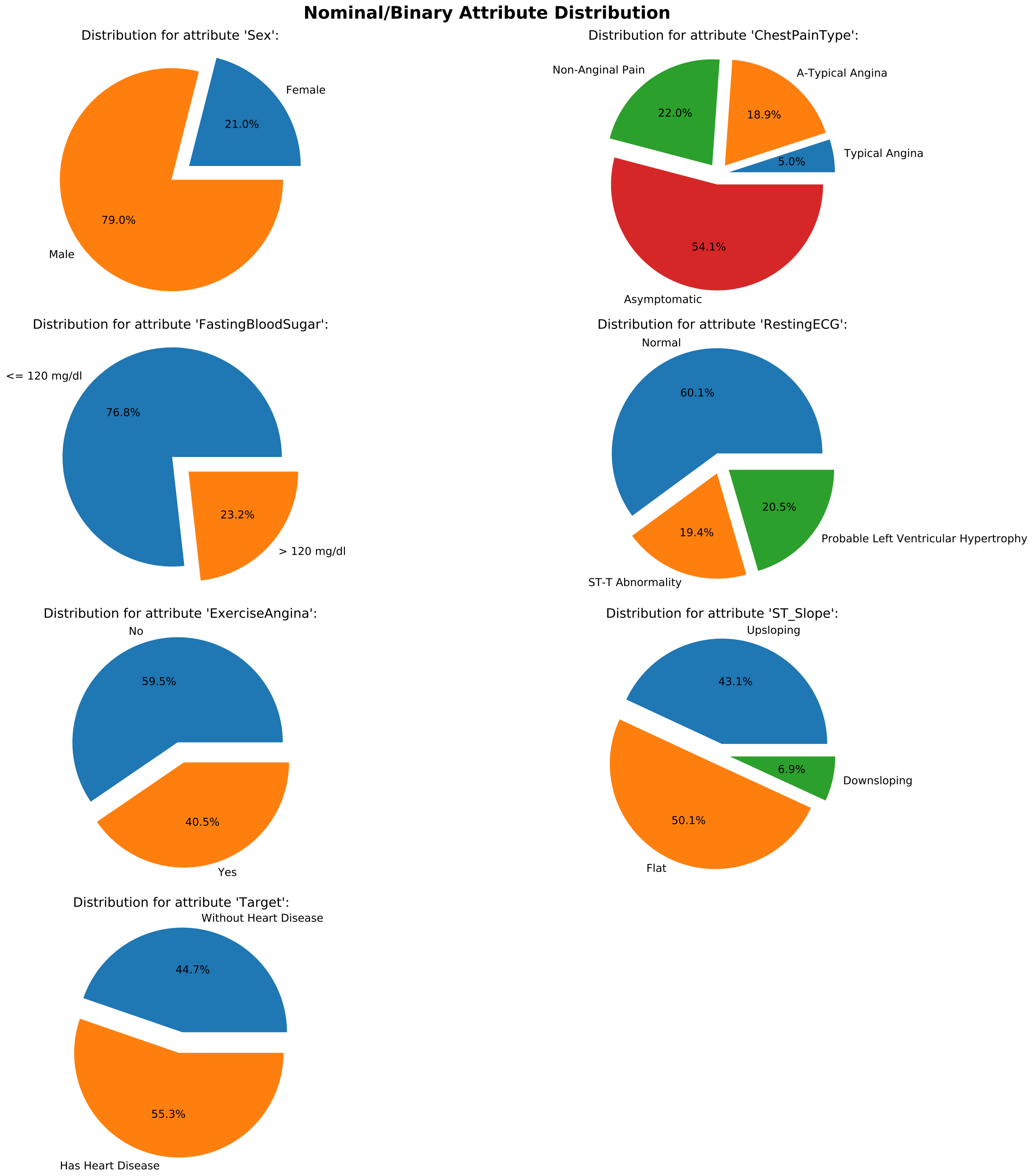
Among the Nominal and Binary data there were two significant findings that suggest this data-set is not representative of a true random population:

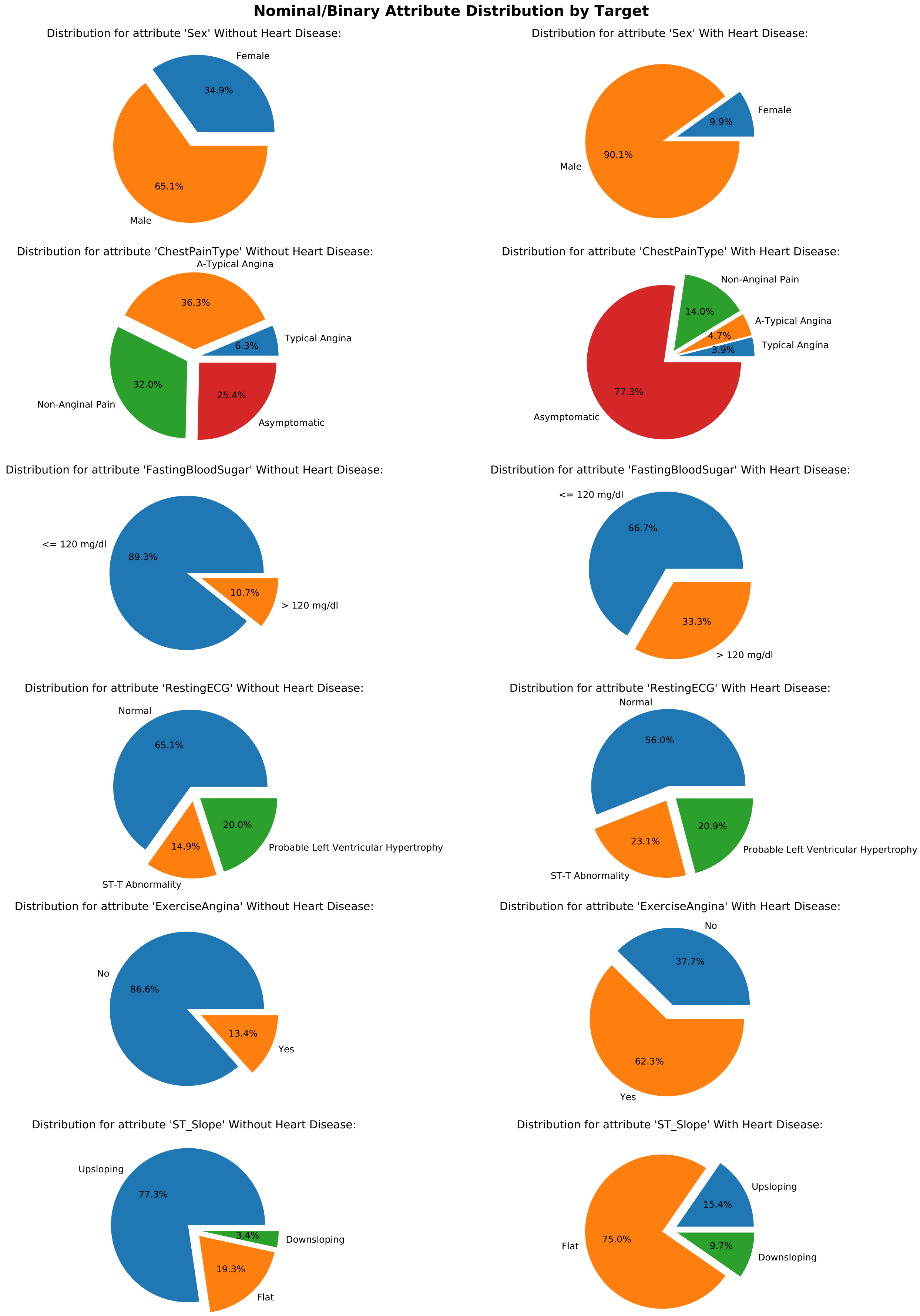
1. The Target variable, that is "Has Heart Disease" and its inverse "Without Heart Disease" is distributed 55% towards the "Has Heart Disease" category. While an evenly distributed class variable may be desirable in the training phase of this project, it does not accurately represent a true population distribution. According to the Center for Disease Control and Prevention(CDC), 7.2% of American's above age 20 are living with Heart Disease.(Centers-for-Disease-Control-and-Prevention, 2022b) While this equates to over 20 million people, it still is significantly lower than the 55% contained is this data-set .

2. This data-set contains records from 79% Males, and only 21% Female. This clearly is not representative of a true random population.

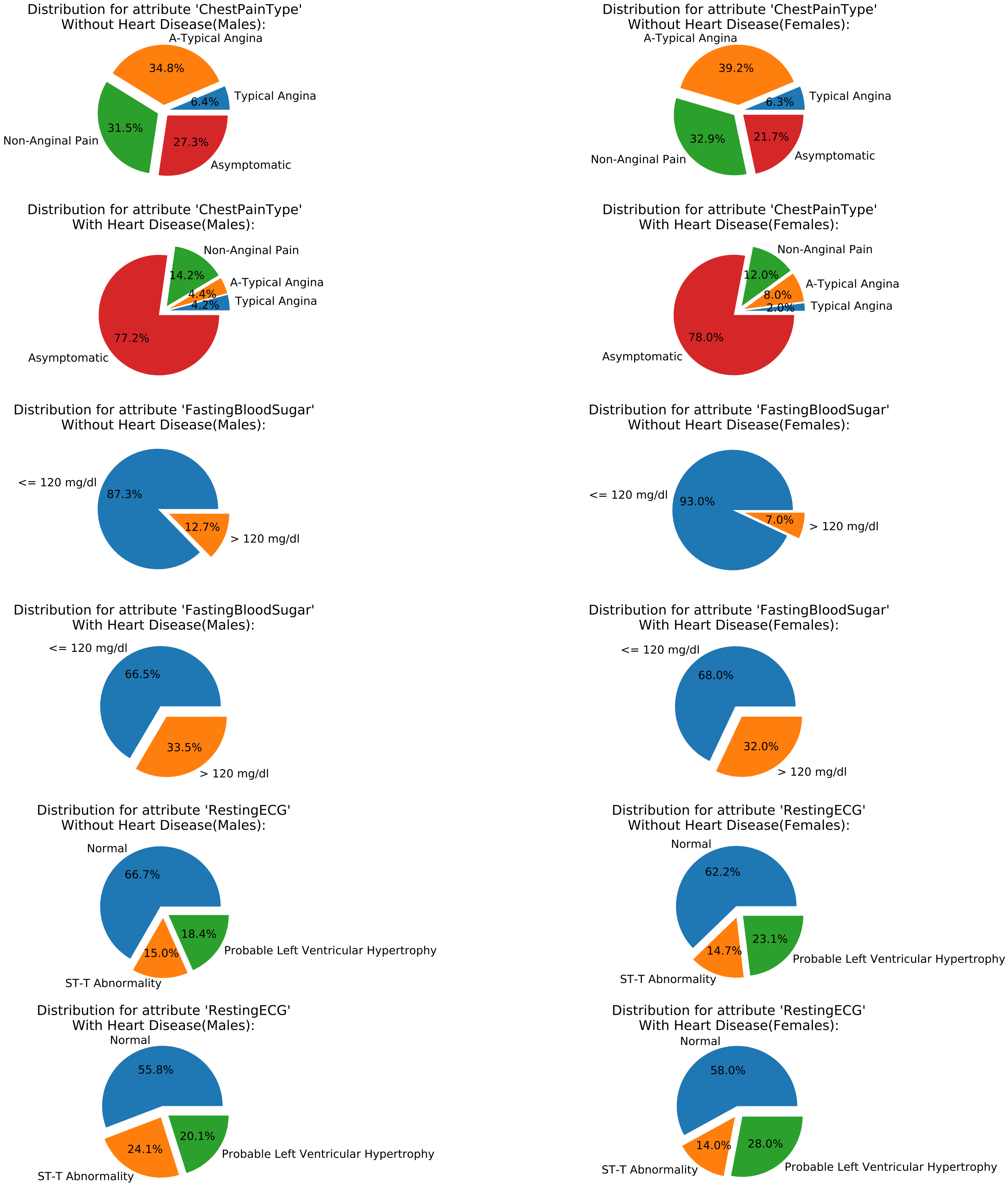
While it is clear this data-set does not truly represent a random population, this is not surprising. The type of attributes contained in this data-set are obtained from clinical tests that the average person would not have access to, or need. For example, how many of us have had an Electrocardiogram(ECG)? It is more likely that the majority of this data was obtained from patients that were already referred to Cardiologists(Heart Disease Specialists), over a heart health concern.

Further, also according to the CDC, males are nearly twice as likely as females to die of heart disease.(Centers-for-Disease-Control-and-Prevention, 2022a) This, along with thought that this data-set was obtained from those whom were already referred to a Cardiologist, explains the bias in the dataset regarding 79% males.

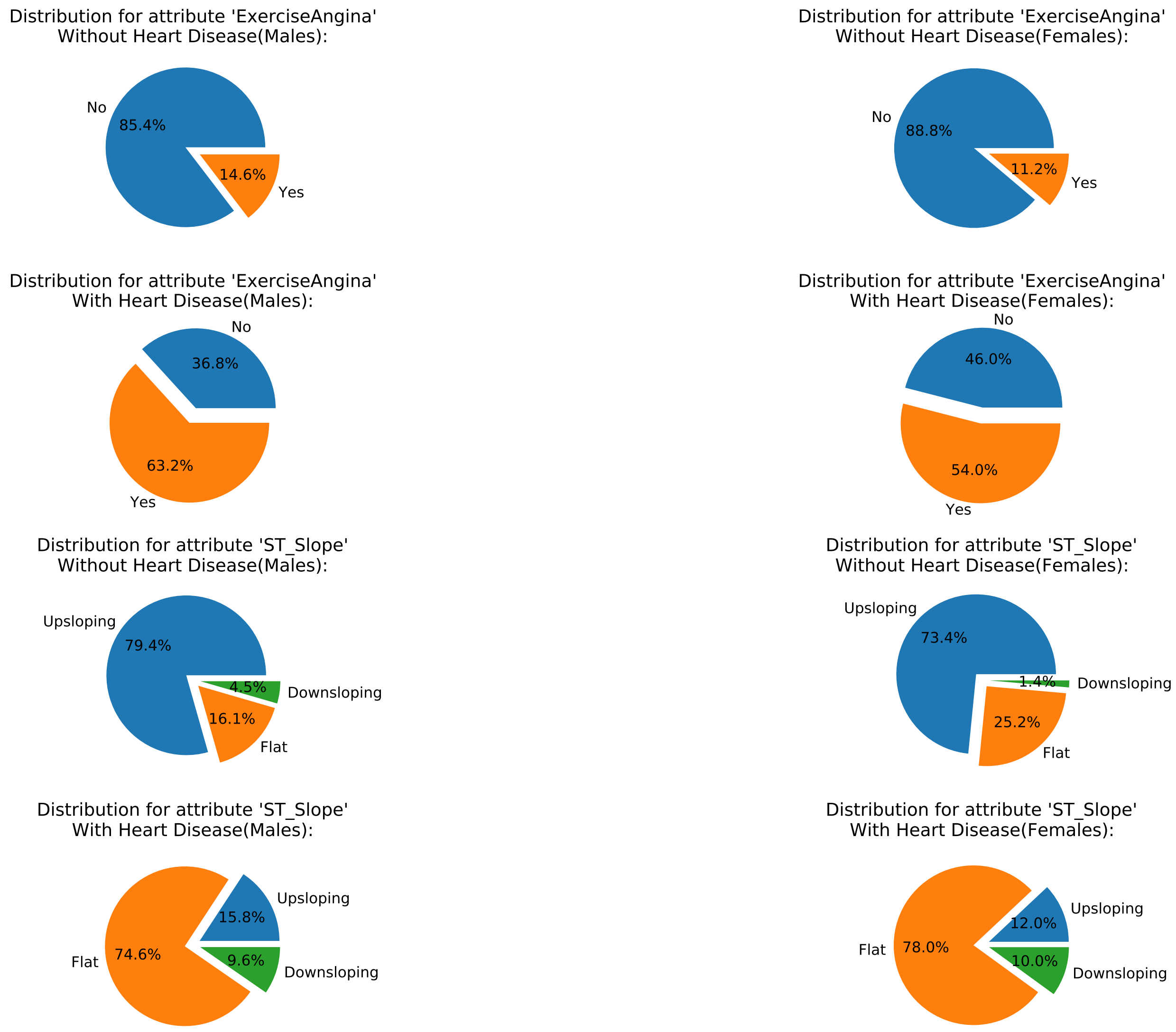

Nominal/Binary Attribute Distribution
Distribution for attribute 'Sex':
Female
21.0%
79.0%
Male
Distribution for attribute 'ChestPainType':
Non-Anginal Pain
A-Typical Angina
22.0%
18.9%
Typical Angina
5.0%
54.1%
Asymptomatic
Distribution for attribute 'FastingBloodSugar':
<= 120 mg/dl
76.8%
23.2%
> 120 mg/dl
Distribution for attribute 'RestingECG':
Normal
60.1%
20.5%
19.4%
Probable Left Ventricular Hypertrophy
ST-T Abnormality
Distribution for attribute 'ExerciseAngina':
No
59.5%
40.5%
Yes
Distribution for attribute 'ST_Slope':
Upsloping
43.1%
6.9%
Downsloping
50.1%
Flat
Distribution for attribute 'Target':
Without Heart Disease
44.7%
55.3%
Has Heart Disease


Nominal/Binary Attribute Distribution by Target
Distribution for attribute 'Sex' Without Heart Disease:
Female
34.9%
65.1%
Male
Distribution for attribute 'Sex' With Heart Disease:
Female
9.9%
90.1%
Male
Distribution for attribute 'ChestPainType' Without Heart Disease:
A-Typical Angina
36.3%
6.3%
Typical Angina
32.0%
Non-Anginal Pain
25.4%
Asymptomatic
Distribution for attribute 'ChestPainType' With Heart Disease:
Non-Anginal Pain
14.0%
A-Typical Angina
4.7%
3.9%
Typical Angina
77.3%
Asymptomatic
Distribution for attribute 'FastingBloodSugar' Without Heart Disease:
<= 120 mg/dl
89.3%
10.7%
> 120 mg/dl
Distribution for attribute 'FastingBloodSugar' With Heart Disease:
<= 120 mg/dl
66.7%
33.3%
> 120 mg/dl
Distribution for attribute 'RestingECG' Without Heart Disease:
Normal
65.1%
20.0%
14.9%
Probable Left Ventricular Hypertrophy
ST-T Abnormality
Distribution for attribute 'RestingECG' With Heart Disease:
Normal
56.0%
23.1%
20.9%
Probable Left Ventricular Hypertrophy
ST-T Abnormality
Distribution for attribute 'ExerciseAngina' Without Heart Disease:
No
86.6%
13.4%
Yes
Distribution for attribute 'ExerciseAngina' With Heart Disease:
No
37.7%
62.3%
Yes
Distribution for attribute 'ST_Slope' Without Heart Disease:
Upsloping
77.3%
3.4%
Downsloping
19.3%
Flat
Distribution for attribute 'ST_Slope' With Heart Disease:
Upsloping
15.4%
Flat
75.0%
9.7%
Downsloping

**Nominal/Binary Attribute Distribution by Target and by Sex**



**Nominal/Binary Attribute Distribution by Target and by Sex, Cont'd**



Distribution for attribute 'ExerciseAngina' Without Heart Disease(Males):

Distribution for attribute 'ExerciseAngina' Without Heart Disease(Females):

Distribution for attribute 'ExerciseAngina' With Heart Disease(Males):

Distribution for attribute 'ExerciseAngina' With Heart Disease(Females):

Distribution for attribute 'ST_Slope' Without Heart Disease(Males):

Distribution for attribute 'ST_Slope' Without Heart Disease(Females):

Distribution for attribute 'ST_Slope' With Heart Disease(Males):

Distribution for attribute 'ST_Slope' With Heart Disease(Females):

## Outliers

The detection of outliers in the data-set is done differently between numeric, and nominal(or binary) attributes.

1. Nominal & Binary Attributes: Since these use assigned numbers, they cannot be compared with each other, traditional methods that focus on deviations around the mean do not work. Two methods that add value are range out-of-bounds detection(performed earlier), and checking for sparsely selected values for a particular attribute. For example, with the

attribute "ST_Slope", among Females Without Heart Disease, only 1.4% of records have a value indicating a downsloping ST segment while exercising.

Aside from the previously mentioned single out-of-bounds detection within the data for "ST_Slope", no other attribute among the nominal and binary data within this data-set stood out.

2. Numeric Attributes: With numeric data is there exists many approaches to detecting and eliminating outliers from within a data-set . Three approaches for outlier detection were explored for this project:

    - Any values outside the 1.5 of the InterQuartileRange(1.5IQR)
    - Any values outside the range of Mean +/ 3 x Standard Deviation(3STDEV)
    - Any values outside a well known standard range for that specific attribute(for example, a negative number for Age.)

    As a result of these methods using different criteria for outlier classification, the list of outliers varied.

*Quantity of Outliers per Attribute*

| Method | Age | RestingBP_s | Cholesterol | MaxHeartRate | OldPeak |
|---|---|---|---|---|---|
| Outliers 1.5IQR | 0 | 28 | 183 | 2 | 16 |
| Outliers 3STDEV | 0 | 8 | 3 | 1 | 7 |

As noticed in the data-set visualization, there was a pronounced secondary peak for attribute "Cholesterol" at value zero. Since a value of zero seems unlikely for this attribute, it is more likely that the results for the "Cholesterol" test were not included, or the test was not performed for these patients. A total of 172 rows in this dataset have a "Cholesterol" value of zero. This represents a problem for the construction of ML models.

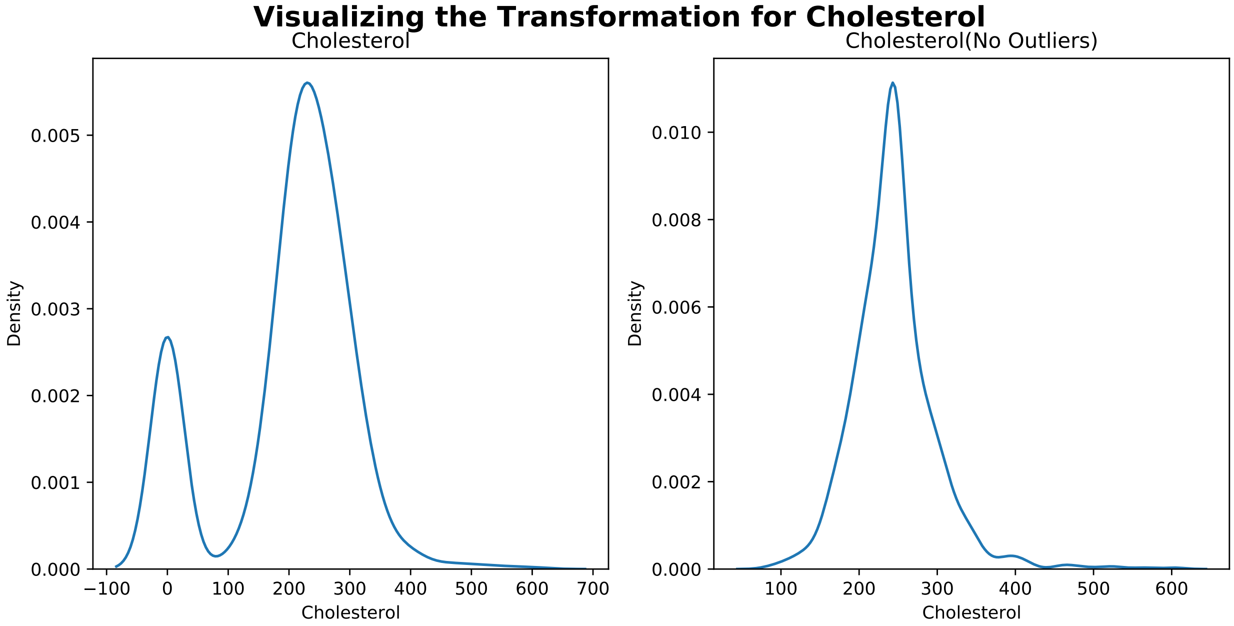
Aside from the attribute "Cholesterol", the amount of outliers in the data-set is relatively small compared to the size of the data-set . As such, only the attribute "Cholesterol" will be considered for any data-correction, all other attribute will remain untouched. Specifically for the attribute "Cholesterol", the zero values were replaced by the mean of all other values for "Cholesterol".

## Data Transformation

### *Cholesterol*

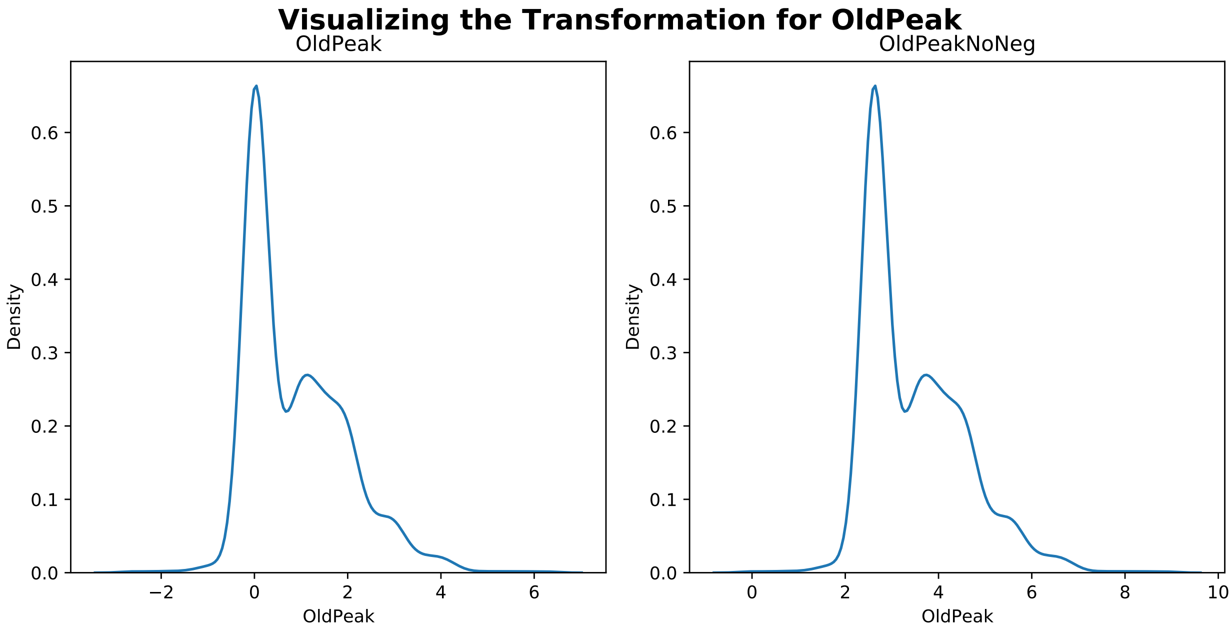
As noticed in the data-set visualization, there was a pronounced secondary peak for attribute "Cholesterol" at value zero. Since a value of zero seems unlikely for this attribute, it is more likely that the results for the "Cholesterol" test were not included, or the test was not performed for these patients. A total of 172 rows in this dataset have a "Cholesterol" value of zero. This represents a problem for the construction of ML models.

Aside from the attribute "Cholesterol", the amount of outliers in the data-set is relatively small compared to the size of the data-set . As such, only the attribute "Cholesterol" will be considered for any data-correction, all other attribute will remain untouched. Specifically for the attribute "Cholesterol", the zero values were replaced by the mean of all other values for "Cholesterol".



### *Oldpeak*

Of all the numeric attributes in this data-set , only the attribute "Oldpeak" contained negative values. One of the ML models explored for this project is Multinomial Naive Bayes Classification, while more suited to categorical features, also works on numeric data providing no negative values are present. In order to attempt the use a Multinomial Naive Bayes Classifier, the values for attribute "Oldpeak" needed to be adjusted to eliminate the negatives. To accomplish this a basic shift in the values was performed, where a fixed number was added to all values such that the lowest value remaining was now equal to zero, and all other values above that.

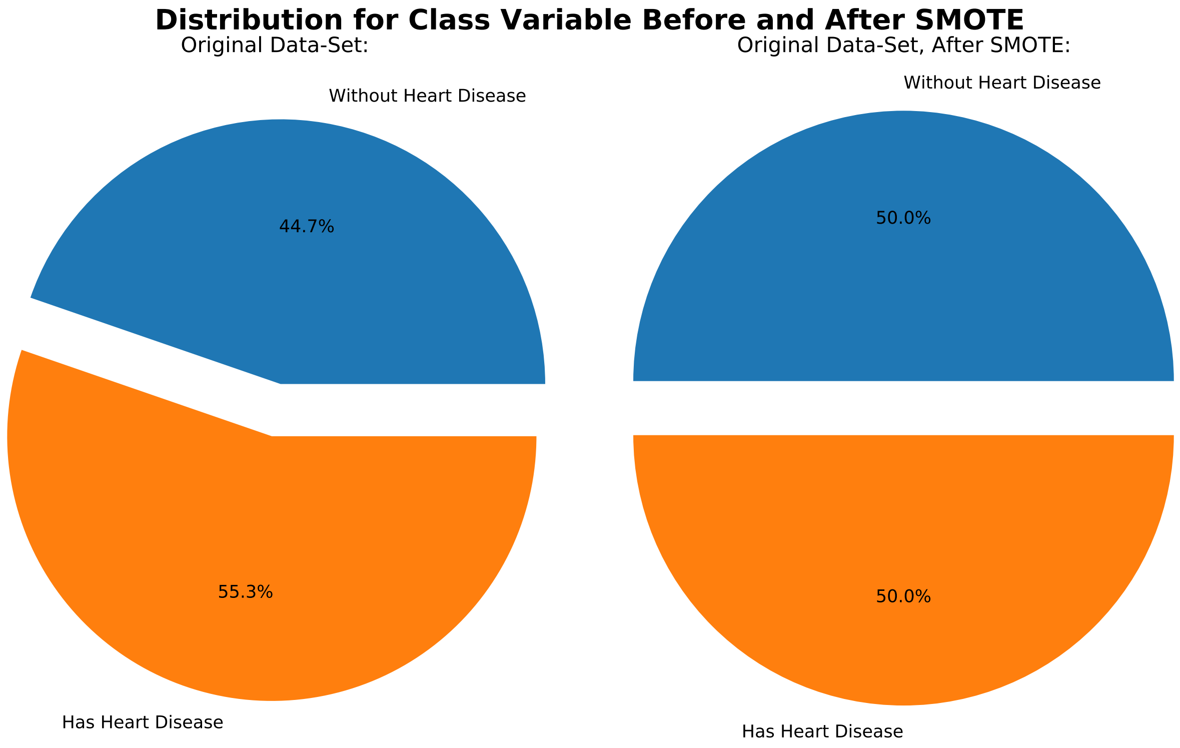


### *Class Imbalance*

As described earlier, there was a small imbalance in the distribution of the class variable, 55% towards the "Has Heart Disease" category. Since some ML models perform unfavorably in the presence of a class imbalance, it was desirable to attempt to balance the class variable for ML model training.

Under-sampling, that is selecting fewer records from the class with the larger representation, in this case "Has Heart Disease", could have been used to provide a balanced data-set for ML model training. Similarly, oversampling, duplicating records from the class with the smaller representation could have also balanced the data-set .

However, another technique, known as Synthetic Minority Over-sampling Technique(SMOTE) was used to balance the data-set . SMOTE creates synthetic(fake) samples by first random oversampling the smaller class, then modifying these samples using random generated values in a range that varies based on the nearest neighbor's to the over-sampled record in the data-set .(Chawla et al., 2002)


Distribution for Class Variable Before and After SMOTE
Original Data-Set:
Without Heart Disease
44.7%
55.3%
Has Heart Disease
Original Data-Set, After SMOTE:
Without Heart Disease
50.0%
50.0%
Has Heart Disease

## Results

The data-set used in this project is relatively small, under 1000 entries after removing duplicates, and invalid entries. As a result, training ML models is quite quick, enabling the ability to analyze multiple models, using multiple variations of the data-set with little concern for overall usage of time, or resources. Therefore instead of training the ML models using one variation of the dataset, multiple variations were tested:

- Original data-set , cleaned, duplicate and invalid entries removed.
- As above, with outliers for Cholesterol changed.
- Original with numerical values normalized.
- All variations above, with categorical attributes converted to single columns using one-hot replacement.
- All variations above, with class imbalance dealt with using SMOTE.

In total eight different variations of the data-set were used in the model training, and testing.

Each of the data-sets were used to train and test the following ML models:

- Decision Tree(DT), using levels three through eleven.
- Gaussian Naive Bayes(NB)
- Logistic Regression(LR)
- Support Vector Machines(SVM)
- K Nearest Neighbors(KNN)
- Random Forest(RF)
- XG Boost(XGB)

With eight variations of the data-set , and fifteen ML models used, this meant 120 models were trained and tested in total. This approach may not be suitable very large data-sets , but with the data-set used for this project it made sense to explore many combinations to determine the best performance.

Two metrics were used for model comparison:

- Over-all Accuracy
- False Negative Rate(FNR) (Miss-Rate)

Over-all accuracy needs no explanation as to why it would be used as a metric for model comparison. False Negative Rate(FNR) was used due to the end goal of this project, aid in the early detection of heart disease, for the benefit of the patients. Since the outcome of the tool is a simple binary, "Has Heart Disease" or "Does not have Heart Disease", a False Negative result would happen in cases where the tool declared "Does not have Heart Disease" yet the patient actually did have Heart Disease.

The opposite of a False Negative(FN) is a False Positive(FP), where the tool declared "Has Heart Disease" when the patient didn’t actually have heart disease. In both cases, FN, and FP, the tool failed to correctly identify Heart Disease or not. With a FP, the patient may get worried for no reason initially, but it will trigger more diagnostic tests, and eventually result in a good outcome. However, with a FN, the patient, and the medical team treating the patient, may falsely believe there is no concern, no reason for continued investigation, diagnostic tests, and the outcome for the patient could be very negative.

As a result of the importance of False Negative, the goal of the testing was to find the models, and data-set variations, that produced the high values for accuracy, while minimizing the values for False Negative Rate.[3]

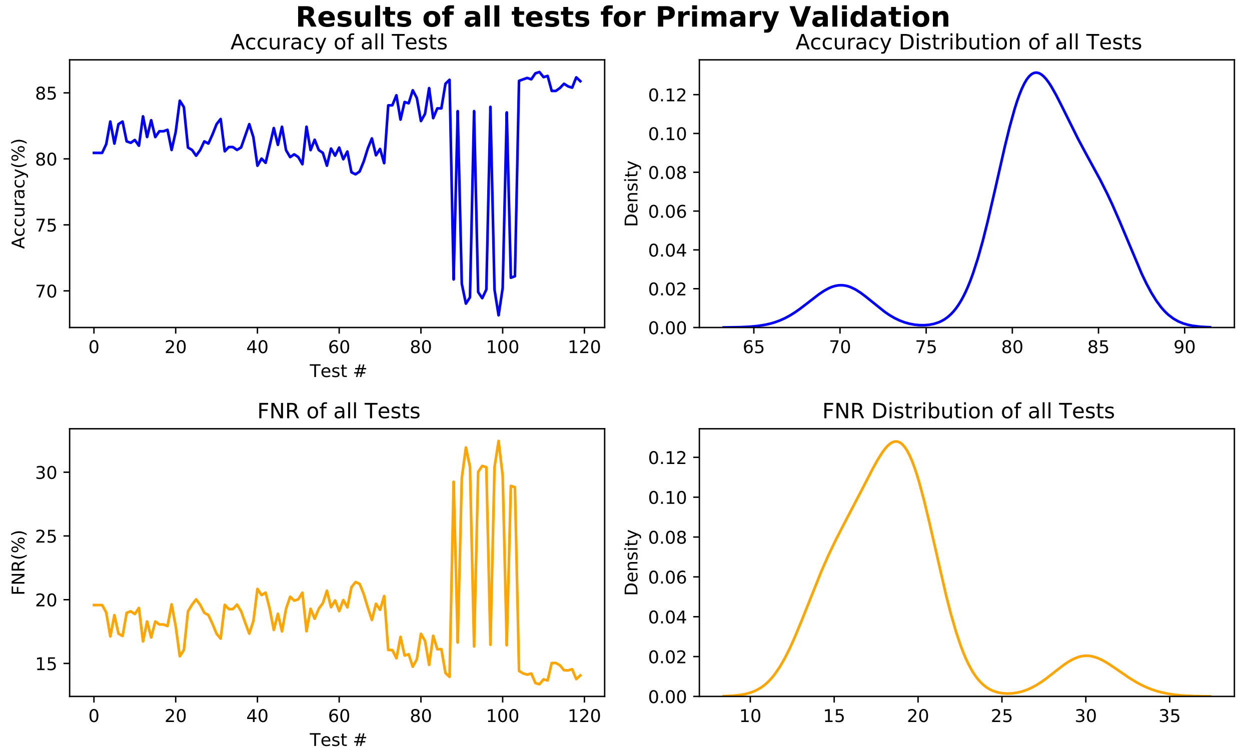
[3] There are competing goals depending on your viewpoint in terms of the importance of FNR versus FPR. The view of a patient, or the view of a hospital administrator. Un-needed tests due to an FP may add extra burden to the healthcare system, something that healthcare administration and government entities may care about more.

For initial testing and validation a fully random iterative approach was taken, where every model was trained using a random 80/20% split, and was repeated hundreds of times, averaging out the accuracy, and false negative rate for each. However, this method, while useful in this case, was replaced with a more conventional cross-fold validation scheme, where every data-set was divided up into twenty folds, and every combination was run twenty times for validation.

The original iterative version can be found here: https://github.com/robert-pineau/CIND-820-Capstone/blob/main/cind820_rpineau_final_iterative_version.ipynb

The revised final Cross-Validation version can be found here: https://github.com/robert-pineau/CIND-820-Capstone/blob/main/cind820_rpineau_final_cv_version.ipynb

## Primary Validation



*Results of Initial Tests sorted by Accuracy (Top 10)*

| ML Model | Data-Set | Accuracy | FNR |
|---|---|---|---|
| Random Forest(RF) | Normalized, After SMOTE | 86.58% | 13.37% |
| Random Forest(RF) | Original, After SMOTE | 86.48% | 13.46% |
| Random Forest(RF) | Outliers Addressed, After ONEHOT, After SMOTE | 86.29% | 13.66% |
| Random Forest(RF) | Original, After ONEHOT, After SMOTE | 86.19% | 13.76% |
| XG Boost(XGB) | Original, After ONEHOT, After SMOTE | 86.18% | 13.76% |
| Random Forest(RF) | Original, After ONEHOT | 86.13% | 14.12% |
| Random Forest(RF) | Outliers Addressed | 86.02% | 14.21% |
| Random Forest(RF) | Normalized | 86.02% | 14.22% |
| Logistic Regression(LR) | Outliers Addressed, After ONEHOT, After SMOTE | 85.99% | 13.95% |
| Random Forest(RF) | Original | 85.91% | 14.41% |

### Secondary Validation

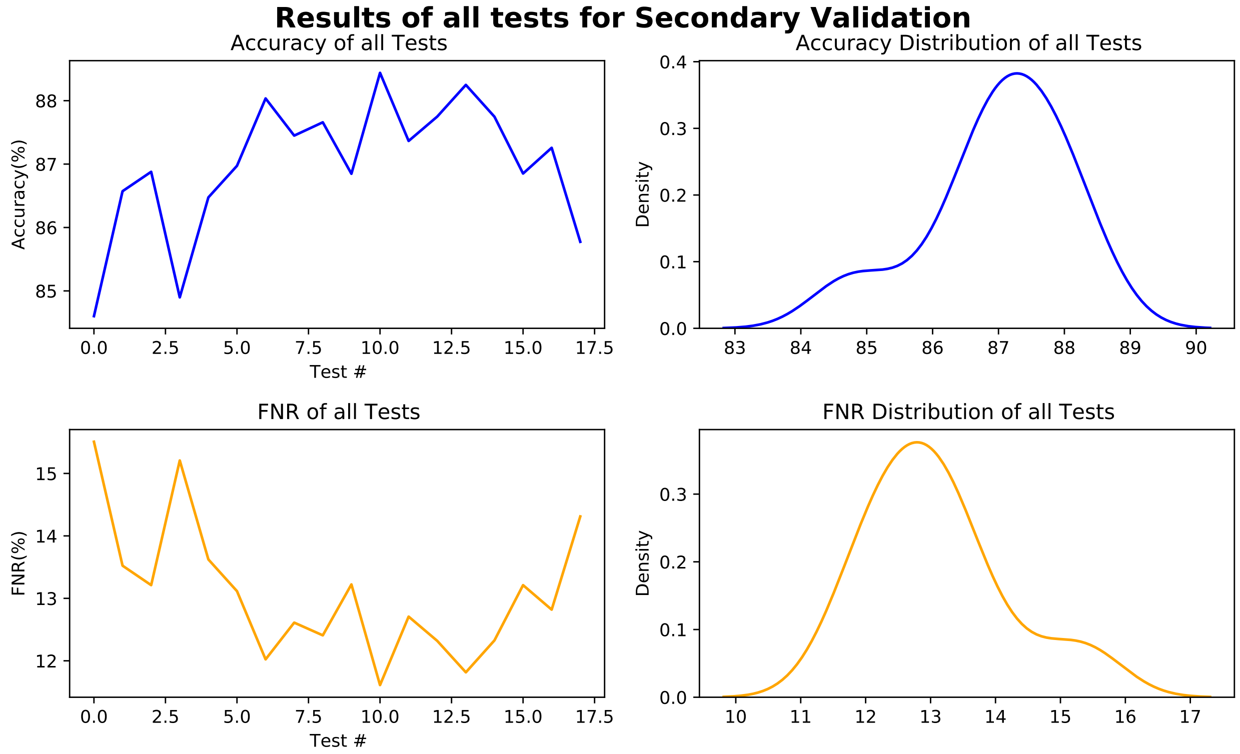
After the primary validation was performed, three ML models, and three data-set variations stood out from the others:

- Original data-set , with class imbalance addressed using SMOTE.
- As above, with class imbalance addressed using SMOTE.
- As above, with outliers for Cholesterol changed.

The ML models that performed better than the others were:

- Logistic Regression(LR)
- Random Forest(RF)
- XG Boost(XGB)

Using all three remaining data-set variations, and the three ML models, hyper-tuning was performed to adjust the models in order to deliver even better results.



*Results of Secondary Validation sorted by Accuracy (Top 10)*

| | | | |
|---|---|---|---|
| Random Forest(RF) | Original, After ONEHOT, After SMOTE | 89.03% | 11.02% |
| Random Forest(RF) - Tuned | Original, After ONEHOT, After SMOTE | 88.34% | 11.72% |
| XG Boost(XGB) | Original, After ONEHOT, After SMOTE | 88.24% | 11.83% |
| Random Forest(RF) | Original, After SMOTE | 88.03% | 12.03% |
| XG Boost(XGB) | Original, After SMOTE | 87.85% | 12.22% |
| Random Forest(RF) - Tuned | Outliers Addressed, After ONEHOT, After SMOTE | 87.56% | 12.52% |
| XG Boost(XGB) - Tuned | Original, After ONEHOT, After SMOTE | 87.55% | 12.52% |
| Random Forest(RF) | Outliers Addressed, After ONEHOT, After SMOTE | 87.26% | 12.81% |
| XG Boost(XGB) | Outliers Addressed, After ONEHOT, After SMOTE | 87.16% | 12.92% |
| Logistic Regression(LR) - Tuned | Outliers Addressed, After ONEHOT, After SMOTE | 87.07% | 12.98% |

The results show accuracy values over 89%, and FNR values under 12%. The best results were obtained using the Random Forest Classifier model, after it was tuned to the original data-set , modified using one-hot replacement for categorical attributes, and SMOTE to address the class imbalance. However, the XG Boost, and a non-tuned Random Forest results were close. In fact the top ten results only varied from 87.07% to 89.03% for accuracy, and 12.98% to 11.02% for FNR.

A dependent paired T-Test was performed on several results(pairs of results), to determine if there is actually any statistical difference between the accuracy of the top result, versus the tenth best result: With a 95% confidence level, Logistic Regression, after tuning, with outliers addressed, and one-hot replacement, and SMOTE for class imbalance, performed just as good as first place Random Forest test.

However, the same dependent paired T-Test showed that all test results outside of the top twelve, were statistically different from the top result.

## Deployment Logistics

Training, testing, and ultimately identifying the ideal ML model is only part of a full solution needed to deploy a workable tool. In order to deliver a tool into the hands of users, logistics regarding consumer models, product updates, financing, and even product support, need to be considered. While the goal of this project is only to deliver the basis for an ML tool to detect heart disease, with many logistic concerns being outside that scope, a few key considerations need to be discussed:

- Sustainability(Tool updates and improvements)
- Deployment models(on-premise vs. cloud)
- Data security and privacy concerns.

### Sustainability

The success, or failure, of any tool, software or otherwise, is based on many factors, including initial quality, initial performance, and ease of use. However, long term success is also based on continued performance upgrades, responsiveness of the vendor to fix issues, and adapt to changing use cases, and patterns. Since a medical a medical based tool operates against patient data this type of tool would be delivered as some form of a software application, versus an actual dedicated device(ie. a fixed single use electronic.)

While a software based solution is much easier to be updated, for fixes, or upgrades, than an actual device, the infrastructure for upgrades, and fixes, needs to exist. Further, the process for upgrades and fixes needs to be easy and trivial to the user, complex procedures will hinder any upgrades, and eventually render the tool obsolete.

### Deployment Models

Thirty years ago the debate within the software community was between micro computer(PC) based solutions, or mainframe solutions: eventually the PC prevailed. Twenty years ago, the discussion became PC based applications versus web based solutions: while a small part of that debate remains, for the most part web based solutions dominate the software landscape today. Ten to fifteen years ago, new devices for software to run on emerged, the smartphone, and handheld tablets. When the smartphone, and dedicated applications(apps), for the smartphone became mainstream, the new debate was web based solution vs. smartphone app: this debate ended up being a non-event, as both are required, any solution developed today needs to keep in mind both use on a PC, through a web based interface, and use on a smartphone, or tablet as an app.

While new technology, and new consumer models will always continue to evolve, in many ways the current discussion has come full circle, back to the days of the mainframe, with the question of Cloud, or On-Premise? Web based solutions dominate modern software deliverables, in the past all of the the servers where these web based solutions were executed existed in the data-centers of the corporations delivering these solutions, corporations large and small. Today the debate is in regards to should a web-server be managed by local corporations, or instead exist in large data-centers run by third party companies, like Microsoft, Google, Amazon, and others.

According to a recent survey by the publishing company O'Reilly, nearly 90% of IT respondents world wide reported their corporation had at least begun to move applications to a cloud model.(Loukides, 2021) Further, from the same survey, 67% of respondents reported this move was to a public based cloud, the majority with Amazon(AWS), Microsoft(Azure), and Google(Google Cloud), collectively known as the "big three" in the world of cloud.

While the concept of a private cloud does exist, that is a web based applications running on a corporation's own server, but using cloud specific APIs, it is the move to public based cloud providers that is of interest when discussing it in conjunction with medical based applications. In the same O'Reilly survey mentioned above, 85% of the respondents specific to the healthcare industry reported adoption of cloud based solutions, again the majority with the big three.

## Data Security and Privacy Concerns

An ML based tool to detect heart disease, by its own definition will need to take a patient's data as input, be computed against the model, and a result returned back to the user. Since the type of data required from the patient is medical in nature, this data falls under the category of private data in many jurisdictions. In Canada, healthcare data falls under the Personal Information Protection and Electronic Document Act(PIPEDA), and Bill-C11. In the USA, healthcare data falls under the Health Insurance Portability and Accountability Act(HIPAA) and the Health Information Technology for Economic and Clinical Health Act(HITECH). In Europe, healthcare data falls under General Data Protection Regulation(GDPR)(Lee & Johar, 2021)

The deployment of a medical tool, one that requires transfer of private patient data to a public cloud server, would require much attention to data & privacy laws, this model would not be trivial to implement. While details regarding the safeguarding of private data in every country is well beyond the scope of this project, it is safe to say there are many challenges in regards to the deployment of a tool that go well beyond coding software and testing algorithms. However, the challenges of navigating data security and privacy issues have already been worked out for many other medical based tools and applications, an ML tool for heart disease detection should be no different. For example, many health jurisdictions are contracting out management of their patient records and data to third party companies, which manage their Electronic Health Records(EHR). One such company EPIC Systems(“Epic Systems,” n.d.), while an American based company, but used in several Canadian hospitals, therefore has presumably worked out many of the logistics required to operate in Canada. Further, through EPIC’s Cosmos(“Epic Cosmos,” n.d.) division, and their "Best Care" program, they are already using data analytics, ML, and AI to improve health care, improve the patient experience.(Raths, 2022)

# Conclusions

The results show that any of Logistic Regression, Random Forest, or XG Boost, could be the basis for an ML tool to detect heart disease. Each of these three performed equally as well, with little difference in the results between them. All provided average accuracy values above 87%, and False Negative Rates below 13%

The results achieved for this project are in-line with past studies including (Latifah et al., 2020), where an accuracy of 84.4% was achieved using RF, and 85.04% using LR, or 87.1% in (Ambrish et al., 2022) using LR, or 86.7% in (Nishadi, 2019) also using LR. However, the results for these projects, and this one, fall short of (Uyar & İlhan, 2017), where using GA RFNN, a model accuracy of 97.78% was achieved.[4]

As mentioned earlier, the ML model is only one small part of delivering a working tool, many logistics regarding privacy, security, support, updates, and future improvements need to be worked out before any commercial product can be released. But the pursuit of potentially life saving tools is a worthwhile effort to say the least.

Finally, there is one question that needs to be asked in regards to accuracy, of any tool used for tasks as important as detecting heart disease: How accurate does it need to be? While this question is being posed, there is no intention to answer it, while it is an important question to ask, is anyone qualified to actually answer it? Perhaps Physicians, Scholars, or Theologians are qualified to provide part of the answer, the author of this paper is not.

[4] Decided Neural Networks were beyond the scope of this project, something to explore in CIND-860.

## References

Aha, D. W. (1988). *UCI Heart Disease Data Set*. https://archive.ics.uci.edu/ml/datasets/Heart+Disease

Ambrish, G., Bharathi, G., Anitha, G., Chetana, S., Dhanraj, & Mensinkal, K. (2022). Logistic regression technique for prediction of cardiovascular disease. *Global Transitions Proceedings*, *3*(1), 127–130. https://www.sciencedirect.com/science/article/pii/S2666285X22000449

Centers-for-Disease-Control-and-Prevention. (2022a). Heart Disease Deaths. *U.S. Department of Health & Human Services*. https://www.cdc.gov/nchs/hus/topics/heart-disease-deaths.htm

Centers-for-Disease-Control-and-Prevention. (2022b). Heart Disease Facts. *U.S. Department of Health & Human Services*. https://www.cdc.gov/heartdisease/facts.htm

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://www.jair.org/index.php/jair/article/download/10302/24590

Dinesh, K. G., Arumugaraj, K., Santhosh, K. D., & Mareeswari, V. (2018). Prediction of cardiovascular disease using machine learning algorithms. *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, 1–7. https://doi.org/10.1109/ICCTCT.2018.8550857

Epic Cosmos. (n.d.). https://cosmos.epic.com/

Epic Systems. (n.d.). https://en.wikipedia.org/wiki/Epic_Systems

*Framingham Heart Study*. (n.d.). https://www.framinghamheartstudy.org

Khan, Z., Mishra, D. K., Sharma, V., & Sharma, A. (2020). Empirical Study of Various Classification Techniques for Heart Disease Prediction. *2020 IEEE 5th International Conference on Computing Communication and Automation*. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9250852

Latifah, F. A., Slamet, I., & Sugiyanto. (2020). Comparison of heart disease classification with logistic regression algorithm and random forest algorithm. *AIP Conference Proceedings*, *2296*(1), 020021. https://aip.scitation.org/doi/abs/10.1063/5.0030579

Lee, L., & Johar, A. (2021). Cloud and the future of healthcare. https://www.itworldcanada.com/blog/cloud-and-the-future-of-healthcare/456259

Loukides, M. (2021). The Cloud in 2021: Adoption Continues. https://get.oreilly.com/ind_the-cloud-in-2021-adoption-continues.html

Mayo-Clinic. (2022). Heart Disease - Diagnosis. *Patient Care & Health Information: Diseases & Conditions*. https://www.mayoclinic.org/diseases-conditions/heart-disease/diagnosis-treatment/drc-20353124

Nishadi, A. S. T. (2019). Predicting Heart Diseases In Logistic Regression Of Machine Learning Algorithms By Python Jupyterlab. *International Journal of Advanced Research and Publications*, *3*(8), 69–74. http://www.ijarp.org/published-research-papers/aug2019/Predicting-Heart-Diseases-In-Logistic-Regression-Of-Machine-Learning-Algorithms-By-Python-Jupyterlab.pdf

Public-Health-Agency-of-Canada. (2022). Heart Disease in Canada. *Government of Canada: Publications - Health*. https://www.canada.ca/en/public-health/services/publications/diseases-conditions/heart-disease-canada.html

Raths, D. (2022). Epic to Pilot Decision Support Tools Using Real-World Data. https://www.hcinnovationgroup.com/clinical-it/clinical-decision-support/article/21285631/epic-to-pilot-decision-support-tools-using-realworld-data

Siddhartha, M. (2020). *Heart Disease Dataset (Comprehensive)*. https://ieee-dataport.org/open-access/heart-disease-dataset-comprehensive

Uyar, K., & İlhan, A. (2017). Diagnosis of heart disease using genetic algorithm based trained recurrent fuzzy neural networks [9th International Conference on Theory and Application of Soft Computing, Computing with Words and Perception, ICSCCW 2017, 22-23 August

2017, Budapest, Hungary]. *Procedia Computer Science*, *120*, 588–593. https://www.sciencedirect.com/science/article/pii/S187705091732495X