HEART DISEASE PREDICTION

using 3 ML Algorithm in R

(Random Forest, KNN, Logistic Regression)

Group Project

Table of Contents

Team Contribution:

Group Members	**Contribution**
Manoj Paudel	*Abstract, Introduction, EDA*
Kushal Pandey	*EDA, Data Mining Techniques, Conclusion*
Sailendra Ghale	*Introduction, Data Mining Results*
Kabin Shrestha	*Abstract, Conclusion*
Rojal Shrestha	*Data Overview, Data Mining Techniques*

Abstract

In today's era, heart disease is a major cause of death around the world, taking an estimated 17.9 million each year. Heart disease acts as a silent killer, early prediction and treatment is crucial where machine learning and data mining can help us to process the data and provide insight which cannot be visible with the naked eye. The goal of our project is to develop a model that can accurately predict whether a person is at high risk of heart disease or not based on a variety of medical features/variables, allowing for early treatment and potentially saving lives. For our study, we utilize a built-in dataset comprising 302 rows and 14 columns with various risk factors for heart disease like age, sex, cholesterol levels(cp), blood pressure, fasting blood sugar (fbs), resting blood sugar, and so on. Our model was trained on this dataset, with 70% train and 30 % test data using three machine learning algorithms like Logistic Regression, Random Forest and KNN. The result demonstrates that the Random Forest algorithm achieved the highest accuracy compared to other 2 models, with accuracy of 81 %.

Introduction

Heart disease is a serious health threat that kills millions of people globally. Cardiovascular disease (CVD) refers to a variety of heart and blood vessels related problems, including coronary artery disease, heart failure, stroke, and peripheral artery disease. According to the World Heart Federation (WHF) report, deaths from CVD jumped globally from 12.1 million in 1990 to 20.5 million in 2021. Despite advancement in medical technology, early detection and prediction of heart disease remain challenging. Since early detection can be aided by machine learning techniques, these opportunities and challenges form the foundation of our study.

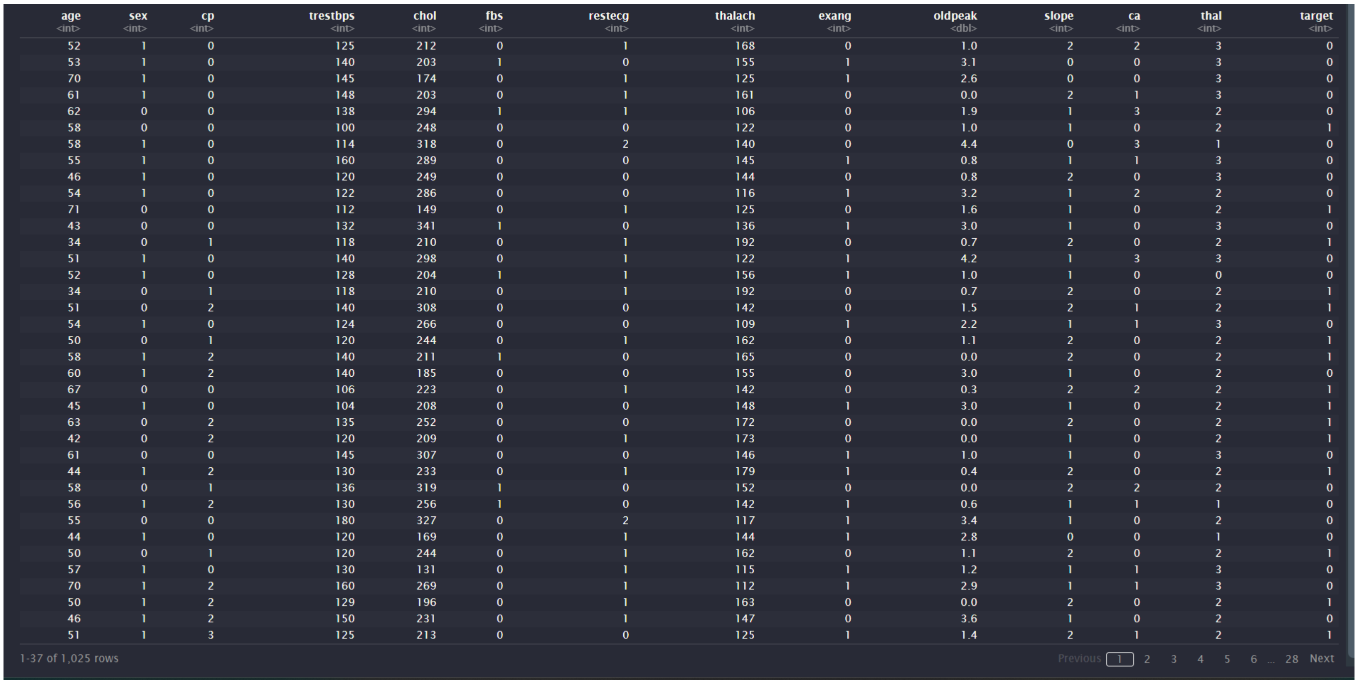
The current methods for predicting heart disease often rely on traditional statistical methods and medical examinations. These techniques can be costly, time-consuming, and might not always correctly estimate a person's risk of developing heart disease because some people may not show symptoms until the condition is advanced. This project's main goal is to find a more effective and precise way to forecast heart disease by using machine learning (ML) algorithms to mine massive databases and find recurring patterns that might potentially result in heart disease. The project's objective is to explore and develop a machine learning algorithm that can accurately predict the risk a person might have related to heart disease based on various health parameters.

Overview

The dataset used in our project was sourced from multiple independent datasets, combined based on common column attributes. These datasets were collected from various prestigious institutions, including the University Hospital in Basel, Switzerland, the V.A. Medical Center in Long Beach and Cleveland Clinic Foundation, the University Hospital in Zurich, Switzerland, and the Hungarian Institute of Cardiology in Budapest. The combined dataset is publicly available on the UCI Machine Learning Repository.

The dataset has 1025 records, each representing a patient, with 14 features related to their health and lifestyle, *age* range from 29 - 77, *sex* [male: 0 and female: 1], *chest pain (cp)* with 4 types: typical angina, atypical angina, non-anginal pain, asymptomatic, *trestbps* related to resting blood pressure ranging from 94 to 200 mm Hg, *chol* ranging from 126 to 564 mg/dl, *fasting blood sugar (fbs)* [1 = true, 0 = false], *restecg (Resting electrocardiographic)* [1 presence, 0: absence], *slope* [0: Upsloping, 1: Flat, 2: Downsloping], *ca* number of major vessels [0-4], *thal (thallium stress test)* , *oldpeak, thalach (maximum heart rate)* range from 71 to 202 and lastly *target* class presence (1) or absence (0) of heart disease.

Here, the sample of dataset:



age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
<int>	<int>	<int>	<int>	<int>	<int>	<int>	<int>	<int>	<dbl>	<int>	<int>	<int>	<int>
52	1	0	125	212	0	1	168	0	1.0	2	2	3	0
53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
61	1	0	148	203	0	1	161	0	0.0	2	1	3	0
62	0	0	138	294	1	1	106	0	1.9	1	3	2	0
58	0	0	100	248	0	0	122	0	1.0	1	0	2	1
58	1	0	114	318	0	2	140	0	4.4	0	3	1	0
55	1	0	160	289	0	0	145	1	0.8	1	1	3	0
46	1	0	120	249	0	0	144	0	0.8	2	0	3	0
54	1	0	122	286	0	0	116	1	3.2	1	2	2	0
71	0	0	112	149	0	1	125	0	1.6	1	0	2	1
43	0	0	132	341	1	0	136	1	3.0	1	0	3	0
34	0	1	118	210	0	1	192	0	0.7	2	0	2	1
51	1	0	140	298	0	1	122	1	4.2	1	3	3	0
52	1	0	128	204	1	1	156	1	1.0	1	0	0	0
34	0	1	118	210	0	1	192	0	0.7	2	0	2	1
51	0	2	140	308	0	0	142	0	1.5	2	1	2	1
54	1	0	124	266	0	0	109	1	2.2	1	1	3	0
50	0	1	120	244	0	1	162	0	1.1	2	0	2	1
58	1	2	140	211	1	0	165	0	0.0	2	0	2	1
60	1	2	140	185	0	0	155	0	3.0	1	0	2	0
67	0	0	106	223	0	1	142	0	0.3	2	2	2	1
45	1	0	104	208	0	0	148	1	3.0	1	0	2	1
63	0	2	135	252	0	0	172	0	0.0	2	0	2	1
42	0	2	120	209	0	1	173	0	0.0	1	0	2	1
61	0	0	145	307	0	0	146	1	1.0	1	0	3	0
44	1	2	130	233	0	1	179	1	0.4	2	0	2	1
58	0	1	136	319	1	0	152	0	0.0	2	2	2	0
56	1	2	130	256	1	0	142	1	0.6	1	1	1	0
55	0	0	180	327	0	2	117	1	3.4	1	0	2	0
44	1	0	120	169	0	1	144	1	2.8	0	0	1	0
50	0	1	120	244	0	1	162	0	1.1	2	0	2	1
57	1	0	130	131	0	1	115	1	1.2	1	1	3	0
70	1	2	160	269	0	1	112	1	2.9	1	1	3	0
50	1	2	129	196	0	1	163	0	0.0	2	0	2	1
46	1	2	150	231	0	1	147	0	3.6	1	0	2	0
51	1	3	125	213	0	0	125	1	1.4	2	1	2	1

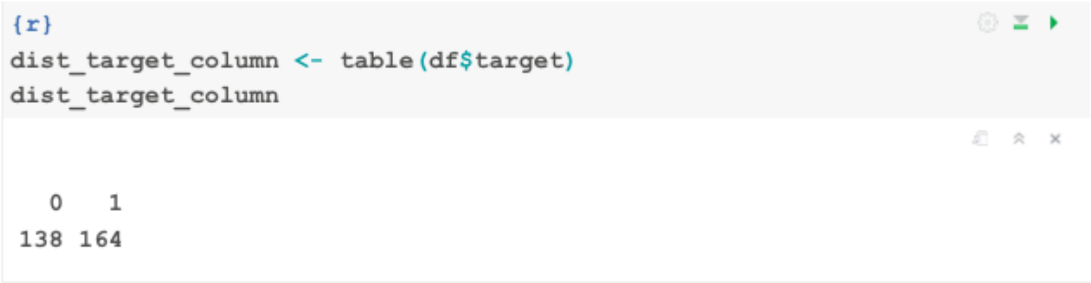
1-37 of 1,025 rows

Dataset Preprocessing:

When initially examining the dataset, several potential issues were detected that may lead to complications in the future. These issues encompassed the presence of duplicate rows, incongruent data, and anomalous values. Such issues possess the capacity to compromise the accuracy of our analysis by producing inaccurate outcomes.

The importance of preprocessing: To ensure the cleanliness, consistency, and analytical readiness of our data, it is crucial that we engage in preprocessing procedures prior to employing any sophisticated modeling techniques. The following actions were undertaken:

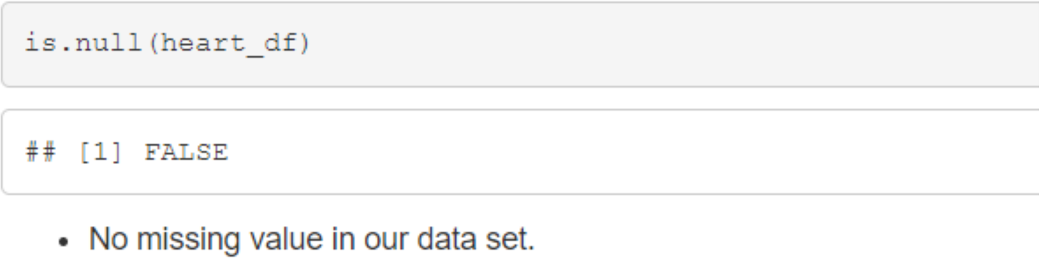
Imbalance Classes : The target class 'Target' is relatively balanced, with slightly more instances of 'Yes' than 'No' with proportion of 45.6 % of 'No' and 54.3 % of 'Yes'.



```
{r}
dist_target_column <- table(df$target)
dist_target_column
```

```
  0   1 
138 164 
```

Addressing missing values: Before we proceed with our analysis, it is imperative to develop a strategy for dealing with any missing data in our dataset. This can be achieved either by completely eliminating them or by filling the gaps with estimated values.



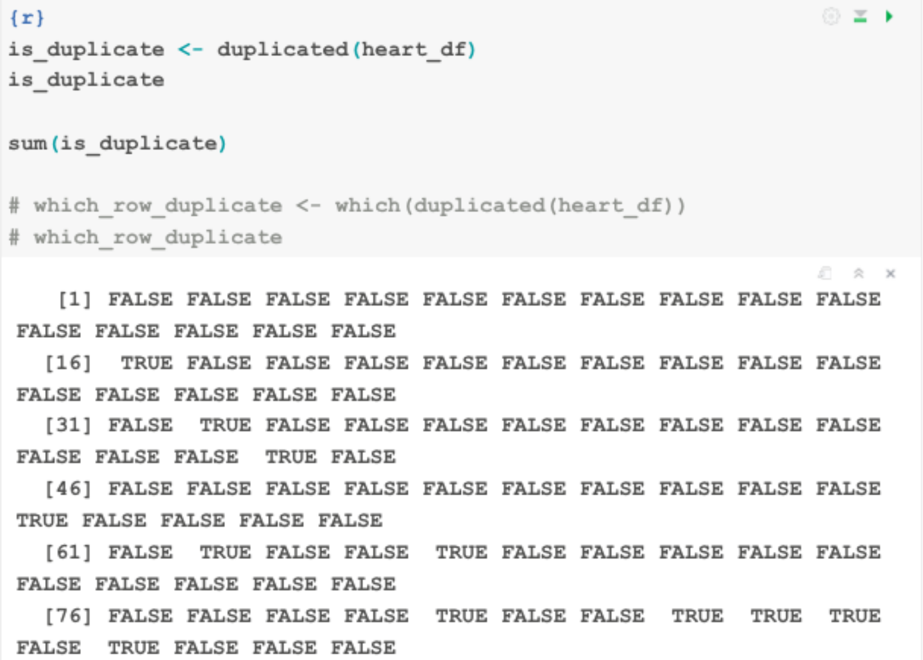
```
is.null(heart_df)
```

```
## [1] FALSE
```

- No missing value in our data set.

In the case of this dataset, there were no missing values.

Recognizing and resolving duplicates: It is also essential to stay vigilant for any duplicate records, which refer to data records that occur in more than one row. We identified the duplicate rows, which was a significant part of the dataset (723) , and removed them thus having only 302 records remaining.

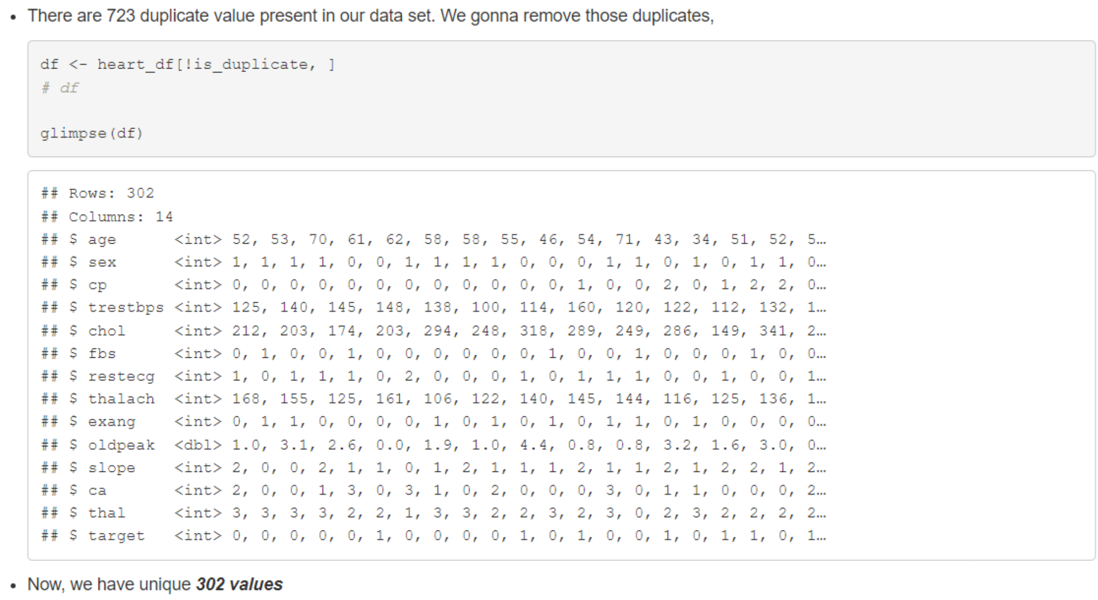


```
{r}
is_duplicate <- duplicated(heart_df)
is_duplicate

sum(is_duplicate)

# which_row_duplicate <- which(duplicated(heart_df))
# which_row_duplicate
```

```
    [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE
   [16]  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE
   [31] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE  TRUE FALSE
   [46] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 TRUE FALSE FALSE FALSE FALSE
   [61] FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE
   [76] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE
 FALSE  TRUE FALSE FALSE FALSE
```



- There are 723 duplicate value present in our data set. We gonna remove those duplicates,

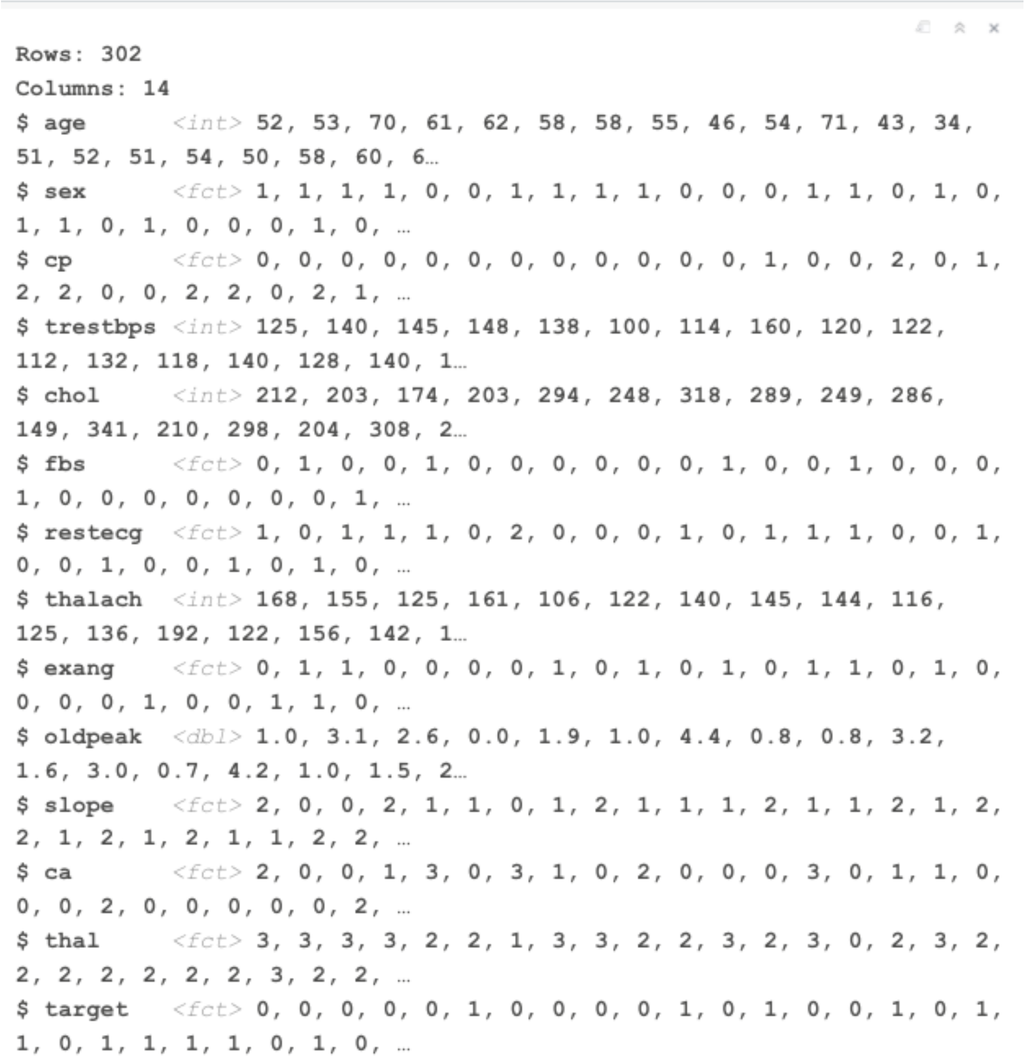
```
df <- heart_df[!is_duplicate, ]
# df

glimpse(df)
```

```
## Rows: 302
## Columns: 14
## $ age      <int> 52, 53, 70, 61, 62, 58, 58, 55, 46, 54, 71, 43, 34, 51, 52, 5…
## $ sex      <int> 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0…
## $ cp       <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 2, 0, 1, 2, 2, 0…
## $ trestbps <int> 125, 140, 145, 148, 138, 100, 114, 160, 120, 122, 112, 132, 1…
## $ chol     <int> 212, 203, 174, 203, 294, 248, 318, 289, 249, 286, 149, 341, 2…
## $ fbs      <int> 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0…
## $ restecg  <int> 1, 0, 1, 1, 1, 0, 2, 0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0, 1…
## $ thalach  <int> 168, 155, 125, 161, 106, 122, 140, 145, 144, 116, 125, 136, 1…
## $ exang    <int> 0, 1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0…
## $ oldpeak  <dbl> 1.0, 3.1, 2.6, 0.0, 1.9, 1.0, 4.4, 0.8, 0.8, 3.2, 1.6, 3.0, 0…
## $ slope    <int> 2, 0, 0, 2, 1, 1, 0, 1, 2, 1, 1, 1, 2, 1, 1, 2, 1, 2, 2, 1, 2…
## $ ca       <int> 2, 0, 0, 1, 3, 0, 3, 1, 0, 2, 0, 0, 0, 3, 0, 1, 1, 0, 0, 0, 2…
## $ thal     <int> 3, 3, 3, 3, 2, 2, 1, 3, 3, 2, 2, 3, 2, 3, 0, 2, 3, 2, 2, 2, 2…
## $ target   <int> 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1…
```

- Now, we have unique ***302 values***

Data Transformation : Even though all the columns of the dataset are numerical and with the right data types, 9 among the 14 columns represent categorical data. The columns *(cp, fbs, restecg,* *slope, ca, thal,sex,, exang, oldpeak and target)* represent categorical data. So it would be better if we change the data type of the columns from numeric to factor data type.

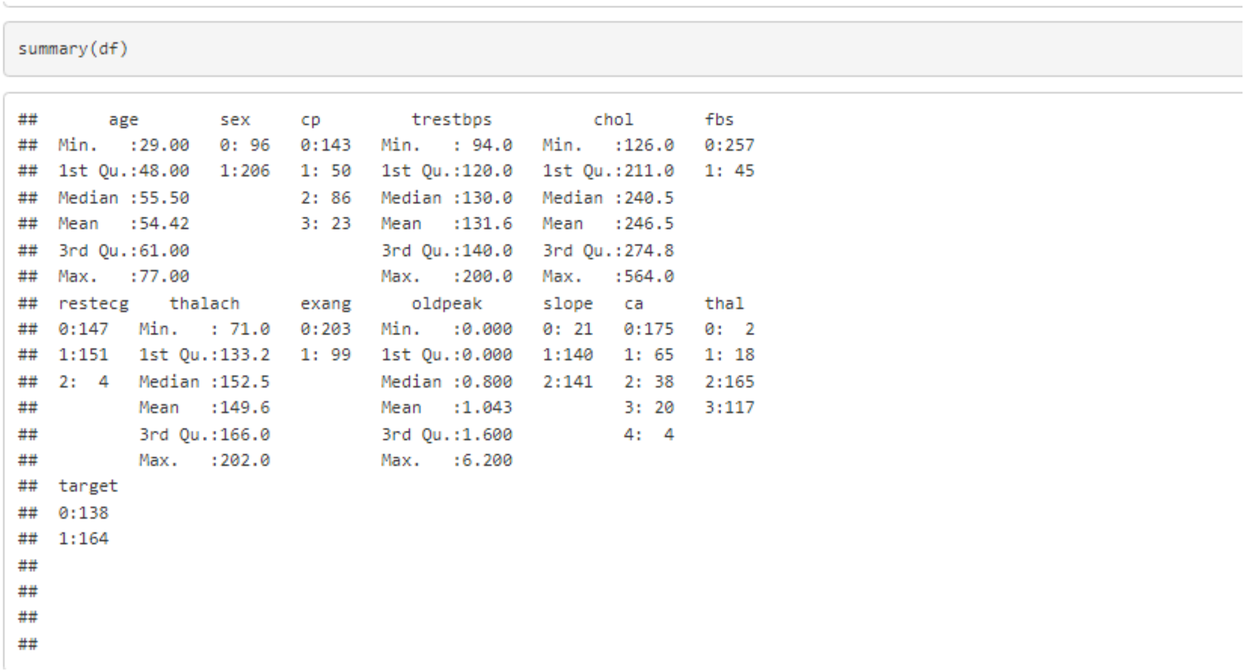


```
Rows: 302
Columns: 14
$ age      <int> 52, 53, 70, 61, 62, 58, 58, 55, 46, 54, 71, 43, 34,
51, 52, 51, 54, 50, 58, 60, 6…
$ sex      <fct> 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0,
1, 1, 0, 1, 0, 0, 0, 1, 0, …
$ cp       <fct> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 2, 0, 1,
2, 2, 0, 0, 2, 2, 0, 2, 1, …
$ trestbps <int> 125, 140, 145, 148, 138, 100, 114, 160, 120, 122,
112, 132, 118, 140, 128, 140, 1…
$ chol     <int> 212, 203, 174, 203, 294, 248, 318, 289, 249, 286,
149, 341, 210, 298, 204, 308, 2…
$ fbs      <fct> 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0,
1, 0, 0, 0, 0, 0, 0, 0, 1, …
$ restecg  <fct> 1, 0, 1, 1, 1, 0, 2, 0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 1,
0, 0, 1, 0, 0, 1, 0, 1, 0, …
$ thalach  <int> 168, 155, 125, 161, 106, 122, 140, 145, 144, 116,
125, 136, 192, 122, 156, 142, 1…
$ exang    <fct> 0, 1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0,
0, 0, 0, 1, 0, 0, 1, 1, 0, …
$ oldpeak  <dbl> 1.0, 3.1, 2.6, 0.0, 1.9, 1.0, 4.4, 0.8, 0.8, 3.2,
1.6, 3.0, 0.7, 4.2, 1.0, 1.5, 2…
$ slope    <fct> 2, 0, 0, 2, 1, 1, 0, 1, 2, 1, 1, 1, 2, 1, 1, 2, 1, 2,
2, 1, 2, 1, 2, 1, 1, 2, 2, …
$ ca       <fct> 2, 0, 0, 1, 3, 0, 3, 1, 0, 2, 0, 0, 0, 3, 0, 1, 1, 0,
0, 0, 2, 0, 0, 0, 0, 0, 2, …
$ thal     <fct> 3, 3, 3, 3, 2, 2, 1, 3, 3, 2, 2, 3, 2, 3, 0, 2, 3, 2,
2, 2, 2, 2, 2, 2, 3, 2, 2, …
$ target   <fct> 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1,
1, 0, 1, 1, 1, 1, 0, 1, 0, …
```

Exploratory Data Analysis (EDA):

In the EDA stage, we performed a thorough analysis of the dataset. We did visualization of each feature with the target class, such as visualizing the distribution of variables and checking the correlation between variables.

Summary Statistics

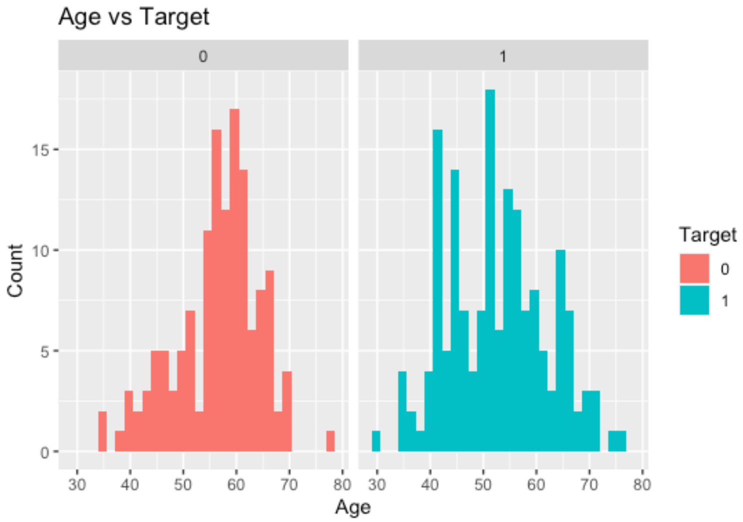


```
summary(df)
```

```
##       age         sex      cp         trestbps          chol         fbs    
##  Min.   :29.00   0: 96   0:143   Min.   : 94.0   Min.   :126.0   0:257  
##  1st Qu.:48.00   1:206   1: 50   1st Qu.:120.0   1st Qu.:211.0   1: 45  
##  Median :55.50           2: 86   Median :130.0   Median :240.5          
##  Mean   :54.42           3: 23   Mean   :131.6   Mean   :246.5          
##  3rd Qu.:61.00                   3rd Qu.:140.0   3rd Qu.:274.8          
##  Max.   :77.00                   Max.   :200.0   Max.   :564.0          
##  restecg    thalach      exang      oldpeak      slope   ca      thal   
##  0:147   Min.   : 71.0   0:203   Min.   :0.000   0: 21   0:175   0:  2  
##  1:151   1st Qu.:133.2   1: 99   1st Qu.:0.000   1:140   1: 65   1: 18  
##  2:  4   Median :152.5           Median :0.800   2:141   2: 38   2:165  
##          Mean   :149.6           Mean   :1.043           3: 20   3:117  
##          3rd Qu.:166.0           3rd Qu.:1.600           4:  4          
##          Max.   :202.0           Max.   :6.200                          
##  target 
##  0:138  
##  1:164  
## 
## 
## 
## 
```

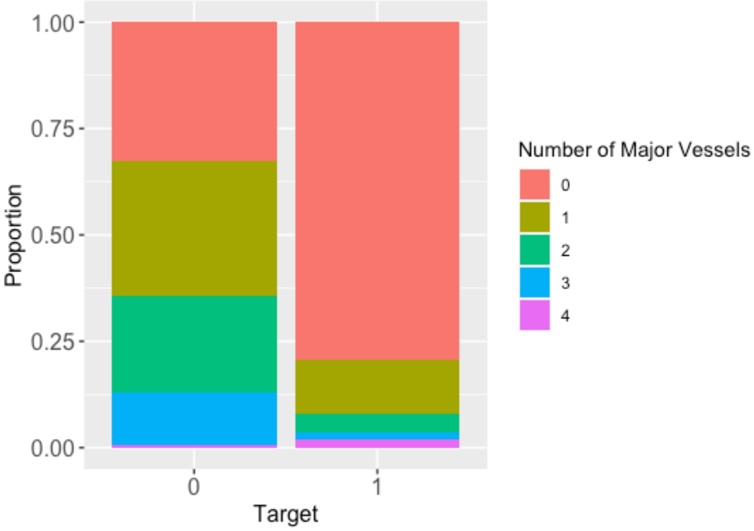
Data Visualization:

1. **age vs target**



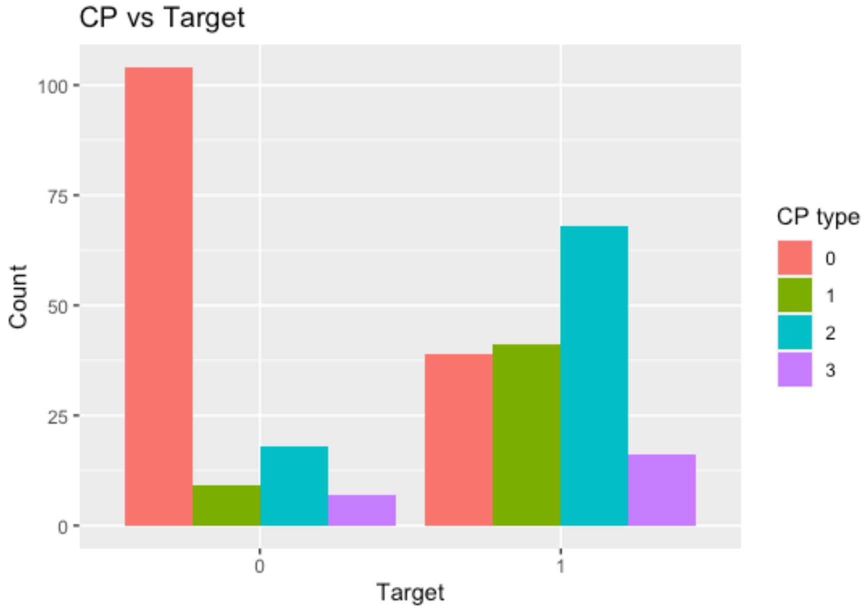
Above figure shows, age between 40 to 60 have a higher probability of having a heart disease as compared to younger people.

2. **ca vs target**



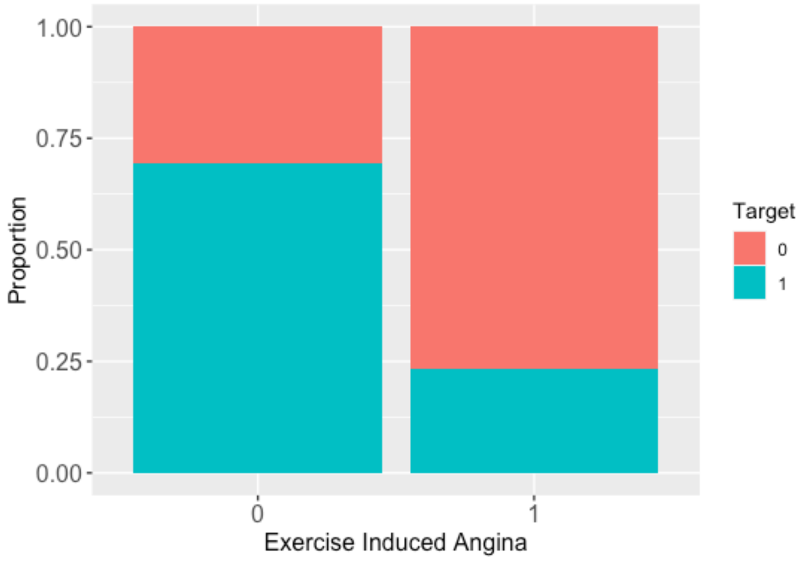
Here, patients with 0 major vessels have the highest probability of heart disease and as the major vessels increase, the probability of having heart disease decreases.

3. **cp vs target**



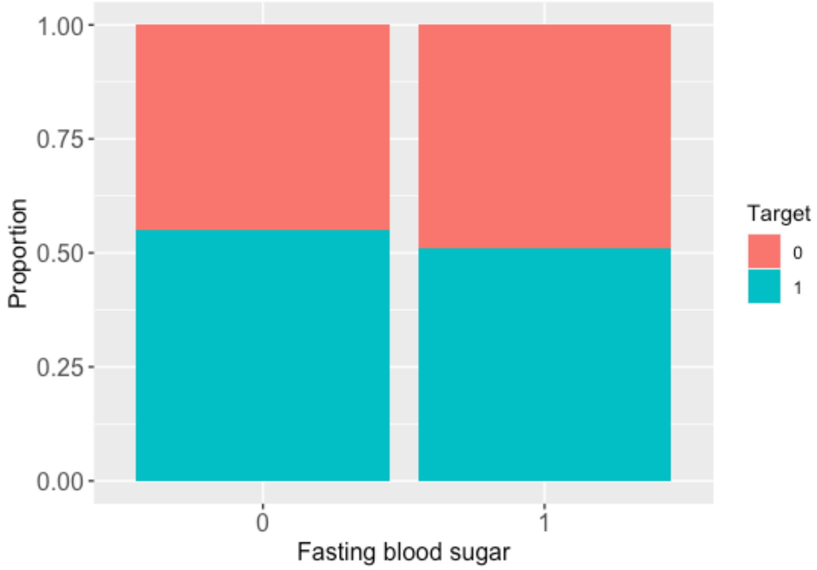
Here, non-anginal chest pain (2) has the highest count for a person to have a heart disease, while asymptomatic (3) has the lowest count.

4. exang vs target



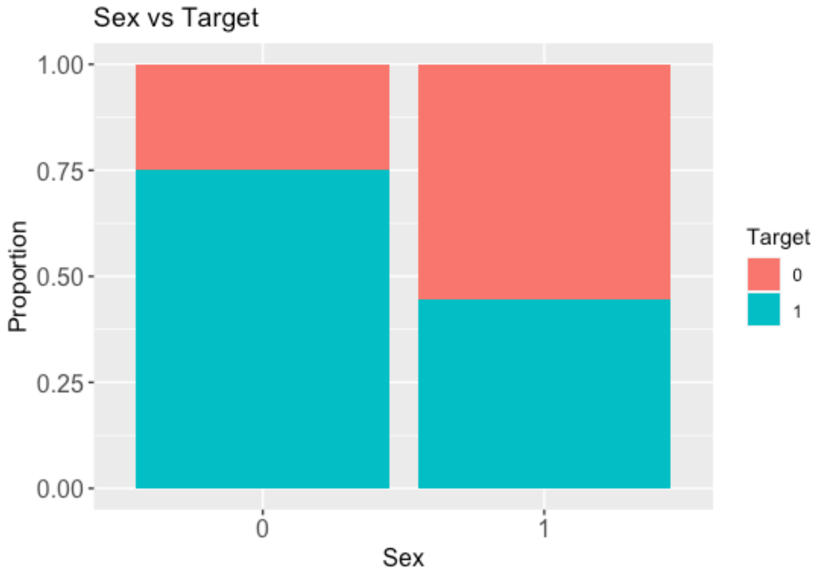
Here, patients who did not have exercise induced angina 0 (no) show high probability of having heart disease compared to those who experience (1). There seems to be a correlation between target classes.

5. **fbs vs target**



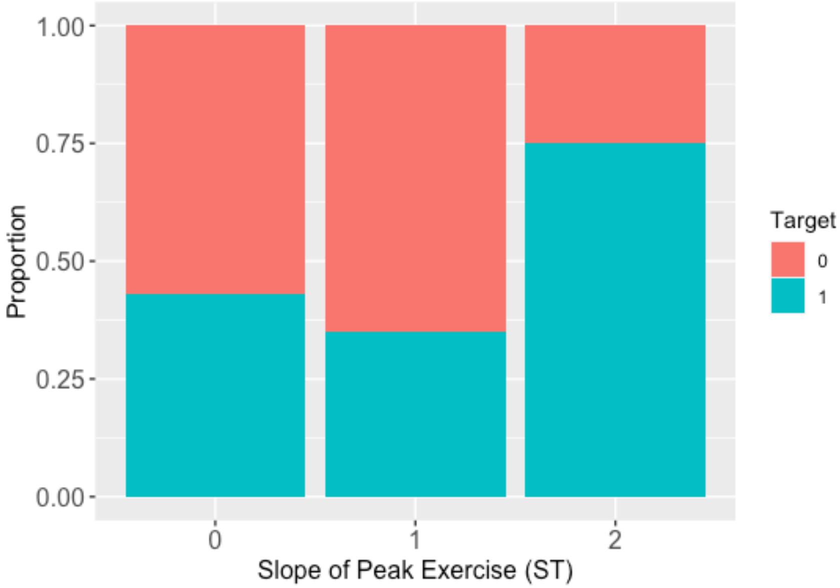
There is a similar distribution between (1) high blood sugar > 120 mg/dl and (0) is relatively similar, so fbs columns have no correlation with target class.

6. **sex vs target**



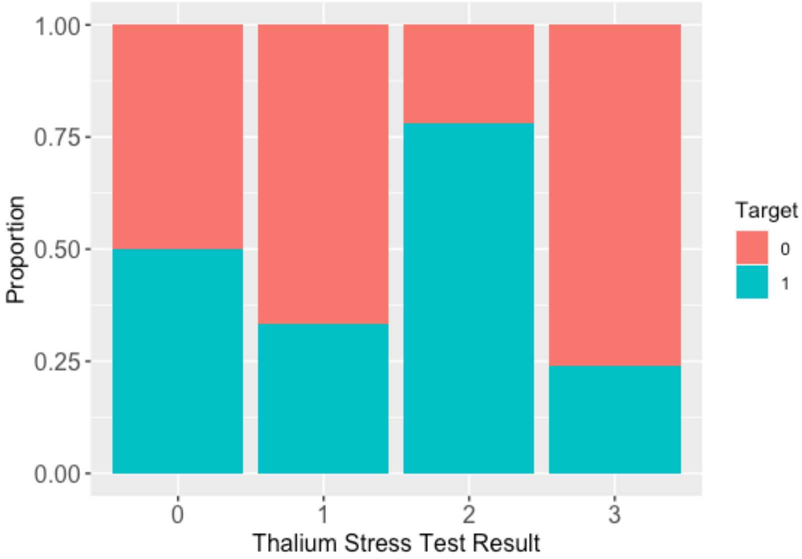
In the above bar plot, male(0) have a high probability of heart disease in comparison to females (1).

7. **slope vs target**



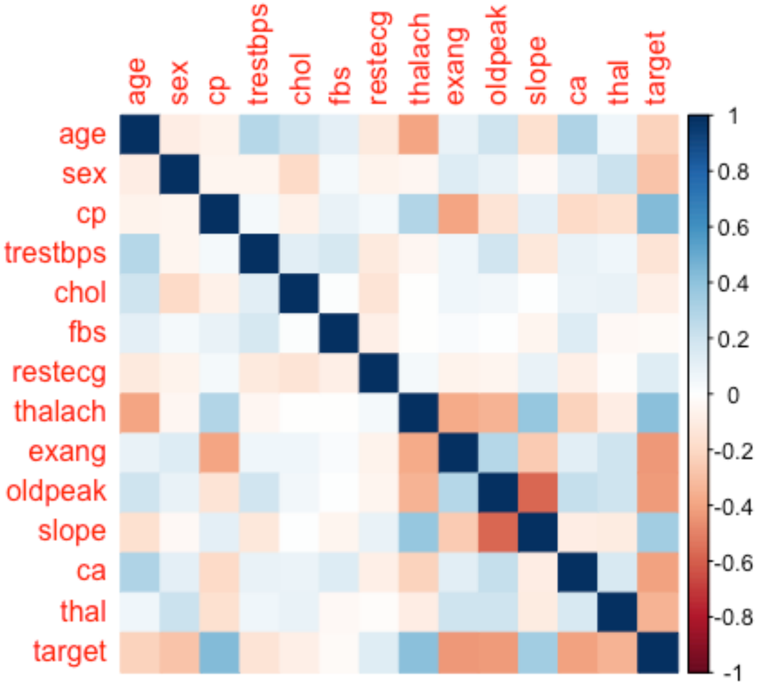
Here, 2 (Down sloping) has a probability of having heart disease with low probability (flat), showing potential to predict target class.

8. thal vs target



Above bar plot shows 2 type (thal) have a high proportion of heart disease compared to other types.

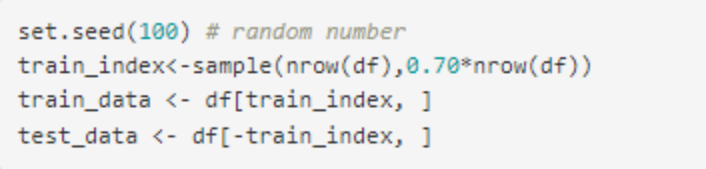
Correlation Analysis:



From the above correlation matrix, we can say, cp, thalach, slope are highly correlated to target column whereas restecg have moderate impact on target class and fbs have low impact on target class. Other features like exang, thal, sex, age ca, have negative correlation with target class.

Data Mining Techniques

Before feeding the dataset into the algorithms, we split the dataset into training and test dataset in a 70:30 split.

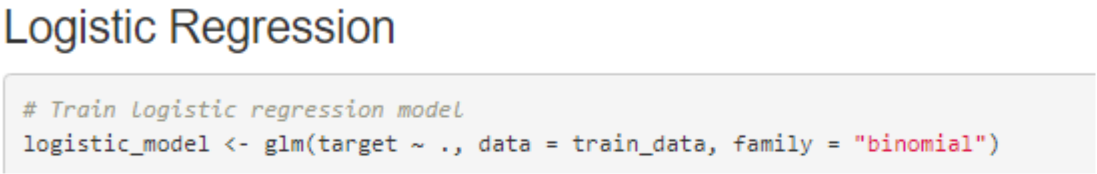


```
set.seed(100) # random number
train_index<-sample(nrow(df),0.70*nrow(df))
train_data <- df[train_index, ]
test_data <- df[-train_index, ]
```

We have used 3 different machine learning algorithms for our predictive model.

1. Logistic Regression

Logistic Regression is a one of the statistical methods used for binary classification tasks. Despite its name, it is primarily used specifically for classification problems, especially when the dependent variable is categorical. It estimates the probability that a given instance belongs to a particular class divided using a sigmoid function.



Logistic Regression

```
# Train logistic regression model
logistic_model <- glm(target ~ ., data = train_data, family = "binomial")
```

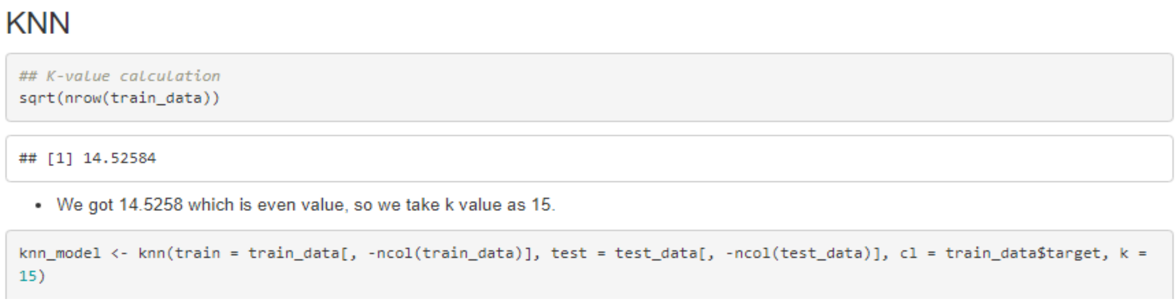
It is suitable for the classification of heart disease risk because of its simplicity, interpretability and because it works well with binary classification (classifying into 0 and 1). It also allows us to understand the relationship between each feature and probability of heart disease risk.

2. K- Nearest Neighbor

KNN, known as K-Nearest Neighbor, is a non-parametric and lazy learning algorithm that is used for both classification and regression tasks. It classifies instances based on similarity measures between data points. K defines the number of nearest neighbors to consider.

KNN finds the K nearest neighbors or groups in the training dataset based on distance metrics like Euclidean Distance, Manhattan distance etc. It then assigns the majority class among these neighbors to the test instance.

KNN is suitable for this case because it doesn't make strong assumptions about the underlying data distribution, which can be useful when the relationship between predictors and outcome is complex or non-linear. It can capture local patterns in the data and adapt well to changes in the data distribution.



KNN

```
## K-value calculation
sqrt(nrow(train_data))
```

```
## [1] 14.52584
```

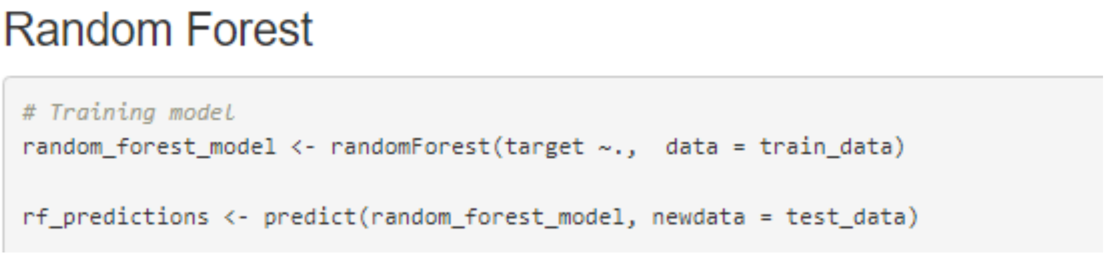
- We got 14.5258 which is even value, so we take k value as 15.

```
knn_model <- knn(train = train_data[, -ncol(train_data)], test = test_data[, -ncol(test_data)], cl = train_data$target, k =
15)
```

3. Random Forest

Random Forest, an ensemble learning method is a tree structure based algorithm that obtains a multitude of decision trees during training where each tree is trained independently, and the ultimate prediction is made by combining the predictions of all trees (e.g., taking a majority vote).

Random Forest is suitable for this case because of the mix between numerical and categorical values in our dataset. It can handle high-dimensional datasets with categorical and numerical features



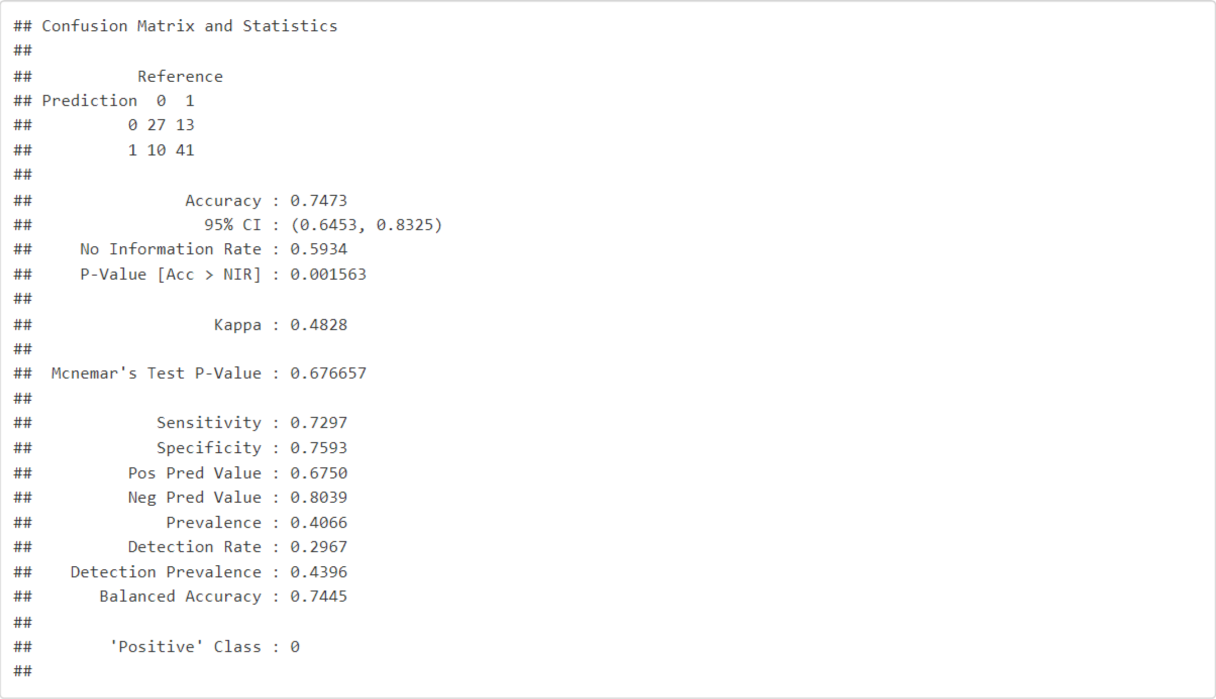
Random Forest

```
# Training model
random_forest_model <- randomForest(target ~.,  data = train_data)

rf_predictions <- predict(random_forest_model, newdata = test_data)
```

Data Mining Results

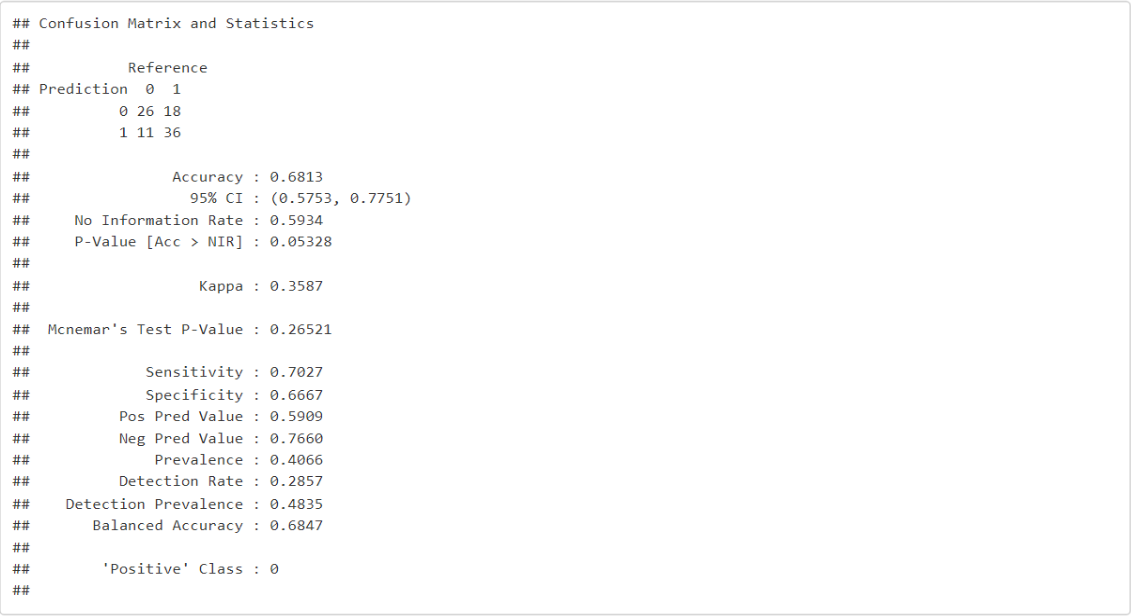
1. Logistic Regression



```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 27 13
##          1 10 41
##
##                Accuracy : 0.7473
##                  95% CI : (0.6453, 0.8325)
##     No Information Rate : 0.5934
##     P-Value [Acc > NIR] : 0.001563
##
##                   Kappa : 0.4828
##
##  Mcnemar's Test P-Value : 0.676657
##
##             Sensitivity : 0.7297
##             Specificity : 0.7593
##          Pos Pred Value : 0.6750
##          Neg Pred Value : 0.8039
##              Prevalence : 0.4066
##          Detection Rate : 0.2967
##    Detection Prevalence : 0.4396
##       Balanced Accuracy : 0.7445
##
##        'Positive' Class : 0
##
```

- Using **logistic regression** we achieved accuracy with 74 %

2. K-Nearest Neighbor

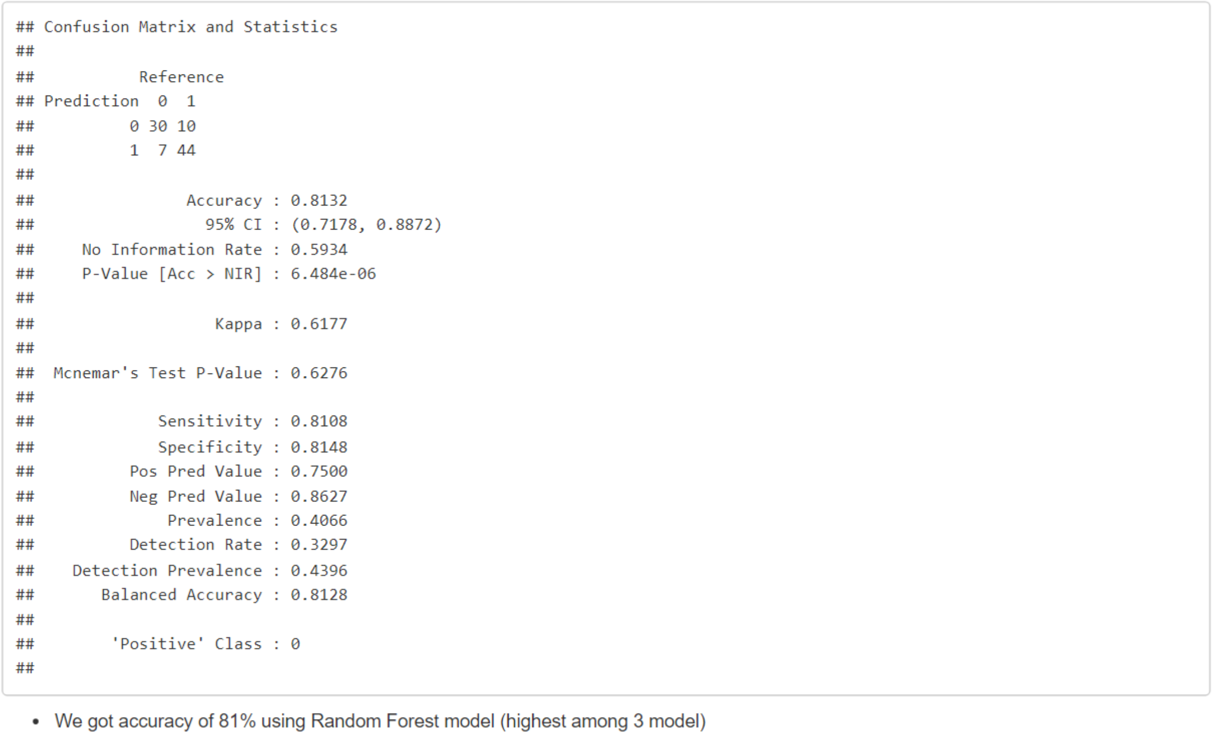


```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 26 18
##          1 11 36
##
##                Accuracy : 0.6813
##                  95% CI : (0.5753, 0.7751)
##     No Information Rate : 0.5934
##     P-Value [Acc > NIR] : 0.05328
##
##                   Kappa : 0.3587
##
##  Mcnemar's Test P-Value : 0.26521
##
##             Sensitivity : 0.7027
##             Specificity : 0.6667
##          Pos Pred Value : 0.5909
##          Neg Pred Value : 0.7660
##              Prevalence : 0.4066
##          Detection Rate : 0.2857
##    Detection Prevalence : 0.4835
##       Balanced Accuracy : 0.6847
##
##        'Positive' Class : 0
##
```

- Using **K-Nearest Neighbor** we achieved accuracy with 68 % which less than **logistic regression** model.

The low accuracy by the KNN Model is subject to the Feature Scaling Descrepancy in the dataset, where some numerical attributes are in the range of 0-1, 0-3 while some features being in range of 100-200, 200-300 and so on. Proper feature scaling techniques can be applied to the dataset, while not affecting the categorical natures of the features.

3. Random Forest



```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 30 10
##          1  7 44
##
##                Accuracy : 0.8132
##                  95% CI : (0.7178, 0.8872)
##     No Information Rate : 0.5934
##     P-Value [Acc > NIR] : 6.484e-06
##
##                   Kappa : 0.6177
##
##  Mcnemar's Test P-Value : 0.6276
##
##             Sensitivity : 0.8108
##             Specificity : 0.8148
##          Pos Pred Value : 0.7500
##          Neg Pred Value : 0.8627
##              Prevalence : 0.4066
##          Detection Rate : 0.3297
##    Detection Prevalence : 0.4396
##       Balanced Accuracy : 0.8128
##
##        'Positive' Class : 0
##
```

- We got accuracy of 81% using Random Forest model (highest among 3 model)

Random forest has given us the highest accuracy of prediction which is 81%. It is likely because it works well than others with datasets that has a mix of categorical features and numerical features, both of which are equally significant for the analysis.

Conclusion

Millions of individuals worldwide suffer from heart disease each year, which has the potential to be deadly. Therefore, by giving patients and medical providers the knowledge they need to limit death and save expenses, early detection of cardiac disease can be beneficial. In our project, we used 3 different algorithms: Logistic Regression, Random Forest and K-Nearest Neighbor out of which Random Forest got the highest accuracy 81 % compared to the other two models.

We can conclude that data mining and machine learning have a significant place in the healthcare system. While traditional disease diagnosis relies on doctor intuition and standard procedures, which have drawbacks and are expensive, machine learning models enable cost-effective diagnosis on large datasets. The major limitation of our study is we did our model prediction on a small dataset only with 302 rows and 14 columns which restricted the models for large dataset prediction.

In the future, our supervised machine learning model—which we created using Random Forest, will be able to be tested and trained on a larger dataset with additional features which have a major impact on heart disease like smoking, family genetics and so on.

Appendixes

References

World Heart Federation. (2023, August 9). Deaths from cardiovascular disease surged 60% globally over the last 30 years: Report - World Heart Federation. https://world-heart-federation.org/news/deaths-from-cardiovascular-disease-surged-60-globally-over-the-last-30-years-report/

Akhtar, N. (2021). Heart disease prediction. ResearchGate. https://www.researchgate.net/publication/349140147_Heart_Disease_Prediction

Jindal, H., Agrawal, S., Khera, R., Jain, R., & Nagrath, P. (2021). Heart disease prediction using machine learning algorithms. IOP Conference Series: Materials Science and Engineering, 1022(1), 012072. https://doi.org/10.1088/1757-899x/1022/1/012072

Ahmed, Intisar, "A STUDY OF HEART DISEASE DIAGNOSIS USING MACHINE LEARNING AND DATA MINING" (2022). *Electronic Theses, Projects, and Dissertations*. 1591. https://scholarworks.lib.csusb.edu/etd/1591

Devare, Virendra Sunil, "Heart Disease Prediction Using Binary Classification" (2023). Electronic Theses,Projects, and Dissertations. 1747. https://scholarworks.lib.csusb.edu/etd/1747

Heart Disease Prediction Using Feature Selection and Machine Learning Techniques. (n.d.). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/10076767

Indrakumari, R., Poongodi, T., & Jena, S. R. (2020). Heart Disease Prediction using Exploratory Data Analysis. Procedia Computer Science, 173, 130–139. https://doi.org/10.1016/j.procs.2020.06.017