Project Report

Introduction:

Coronary heart disease (CHD) is an inescapable nemesis for those who live with bad lifestyle, but recent research suggests that CHD may follow certain patterns. This study aims to predict the 10-year risk of future CHD using an ongoing cardiovascular study of residents in Framingham, Massachusetts. The dataset, which includes over 4,000 records and 15 attributes, is publicly available on Kaggle.

The research question is to identify key health factors strongly associated with CHD and evaluate the accuracy of different classification models, including logistic regression, decision trees, random forests, and support vector machines. The project seeks to develop the most accurate model to help the healthcare system identify patients at risk for CHD, enabling early intervention or prevention methods.

Business Problem:

What are the key health factors strongly associated with coronary heart disease (CHD), and which classification model provides the most accurate prediction of the 10-year risk of future CHD? The goal of this project is to assist the healthcare system in identifying patients at risk for CHD, enabling early intervention or prevention methods.

Data source and description:

I have taken our data source from Kaggle (https://www.kaggle.com/datasets/dileep070/heart-disease-prediction-using-logistic-regression?resource=download). The dataset covers history. of 4000 residents in Framingham, Massachusetts. The below tables divide 15 variables of the dataset into 4 categories.

Models used:

1. Logistic regression
2. Decision trees
3. Random forests
4. SVM
5. PCA

Challenges Faced:

1. Unbalanced Data:

Our analysis aims to predict the risk of coronary heart disease using a dataset that presents a significant challenge. The dataset is imbalanced, with only 15% of individuals having a risk of coronary heart disease, while the remaining 85% have no such risk. This poses a challenge because standard machine learning models may be biased towards the majority class, leading to lower predictive performance for the minority class. Addressing this issue will require employing techniques such as data resampling or ensemble learning to improve the accuracy and robustness of our predictions.

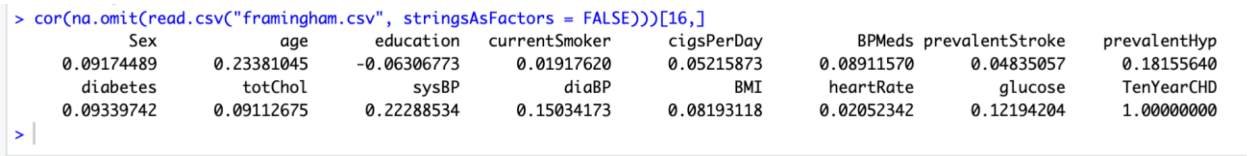
2. Biased Dataset:

A biased dataset is a significant challenge in building machine learning (ML) models, as it can lead to inaccurate models that may not perform well on new data. Biases in the dataset can occur due to various reasons, such as under-representation of certain groups, data collection errors, or intentional bias. Biases can cause the model to make incorrect predictions or generalize poorly to new data, resulting in unreliable and potentially harmful outcomes.

3. Feature Selection

Feature selection involves identifying the most relevant features from the dataset that can help the ML model accurately predict the target variable. It is challenging to select the right features that are relevant to the problem and transform them in a way that improves the model's performance. In addition, selecting too many or too few features can lead to overfitting or underfitting of the model.

4. Low Correlation



```
> cor(na.omit(read.csv("framingham.csv", stringsAsFactors = FALSE)))[16,]
           Sex            age      education  currentSmoker     cigsPerDay         BPMeds prevalentStroke   prevalentHyp
    0.09174489     0.23381045    -0.06306773     0.01917620     0.05215873     0.08911570     0.04835057     0.18155640
      diabetes        totChol          sysBP          diaBP            BMI      heartRate        glucose     TenYearCHD
    0.09339742     0.09112675     0.22288534     0.15034173     0.08193118     0.02052342     0.12194204     1.00000000
> 
```

(*low correlation between the variables*)

Understanding Data Nature:

1.Used combination of Supervised and Unsupervised Models:

In my project, I encountered a dataset that presented a challenge due to the low correlation between independent and response variables. To overcome this, I used a combination of supervised and unsupervised machine learning models to extract useful insights from the data.

I employed four different supervised models, namely logistic regression, decision tree, random forest, and support vector machine (SVM), to predict the outcome variable based on the independent variables. These models allowed me to detect patterns and relationships in the dataset, and to make accurate predictions based on labeled data.

However, a challenge I encountered was the presence of three highly correlated independent variables, which could have led to multicollinearity issues in the supervised models. To address this, I used Principal Component Analysis (PCA), an unsupervised machine learning technique, to reduce the dimensionality of the data and identify the underlying factors that explain the variance among these variables. This approach helped me to create a new set of independent variables that are linearly uncorrelated, allowing me to avoid the problem of multicollinearity and improve the performance of my supervised models.

In summary, by using a combination of supervised and unsupervised machine learning techniques, I was able to gain a better understanding of the complex relationships within the dataset, and to make more accurate predictions and informed decisions. My approach allowed me to overcome the challenge of low correlation and multicollinearity in the data, which is often encountered in real-world problems. I believe that my findings will be useful for future research in this field and can be used to inform decision-making in similar contexts.

2. Balancing Data:

In my project, I encountered an unbalanced dataset, which can lead to biased predictions and reduced model performance. To address this issue, I used two common techniques: oversampling and undersampling.

Oversampling involves increasing the number of instances in the minority class, while undersampling involves reducing the number of instances in the majority class. By using both techniques, I was able to balance the dataset and ensure that the model is trained on an equal number of instances for each class, thus avoiding bias towards the majority class.

After balancing the dataset, I applied the supervised models, namely logistic regression, decision tree, random forest, and support vector machine (SVM), to the dataset. By using the balanced dataset, I ensured that the models were trained on a representative sample of instances for each class, which improved the accuracy of the predictions and the overall performance of the models.

My approach allowed me to overcome the challenge of dealing with unbalanced datasets, which is often encountered in real-world problems. By employing over-sampling and under sampling techniques, I was able to ensure that the models could learn from all the available data while avoiding bias towards the majority class.

Data Preprocessing:

1. Data Cleaning:

Data cleaning is a crucial step in any data analysis project, as it ensures the accuracy and reliability of the results obtained. In my project, I encountered null values in the dataset, which can cause errors in the analysis and lead to incorrect conclusions.

To address this issue, I used R code to remove the null values from the dataset. This allowed me to ensure that the dataset was complete and that all the necessary information was available for analysis. By removing the null values, I was able to avoid potential errors in the analysis and obtain more accurate results.

2. Nature of Independent Variables:

In my project, I analyzed the nature of the independent variables and their relationship with the response variable. I found that each independent variable was almost normally distributed in relation to the response variable. I visualized this relationship through graphs that demonstrated the normal distribution of each independent variable in relation to the response variable.

By understanding the nature of the independent variables, I was able to gain insights into the underlying patterns and relationships within the dataset. The normal distribution of the independent variables in relation to the response variable suggests a non-linear relationship between the variables, which is useful information for modeling and prediction purposes.

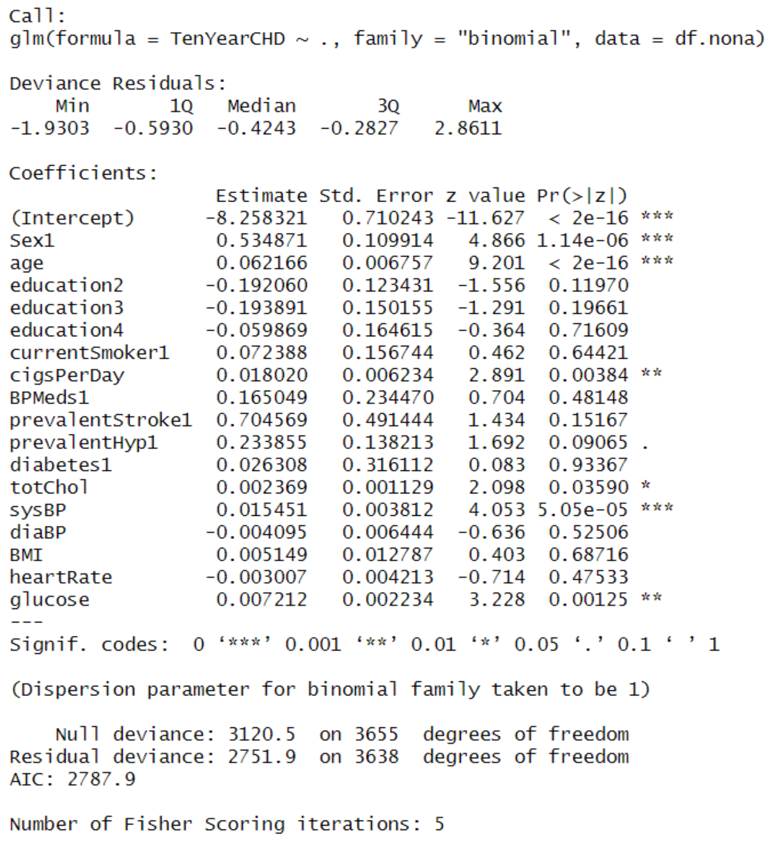
By visualizing this relationship through graphs, I was able to better understand the underlying patterns and relationships within the data, and to make more accurate predictions and informed decisions.

Model Building:

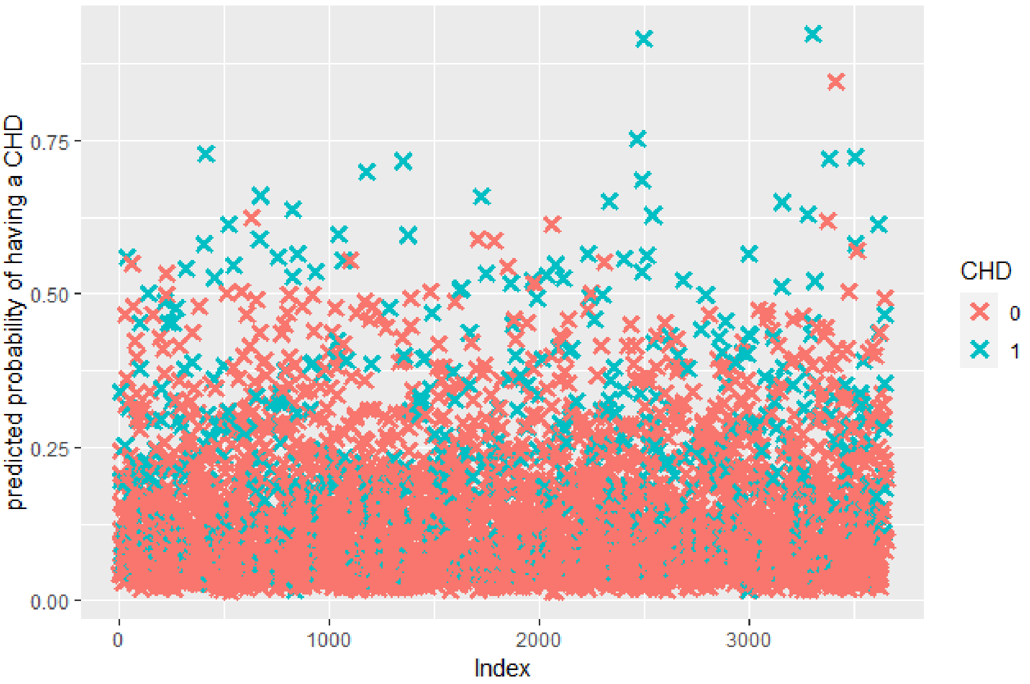
1. PCA

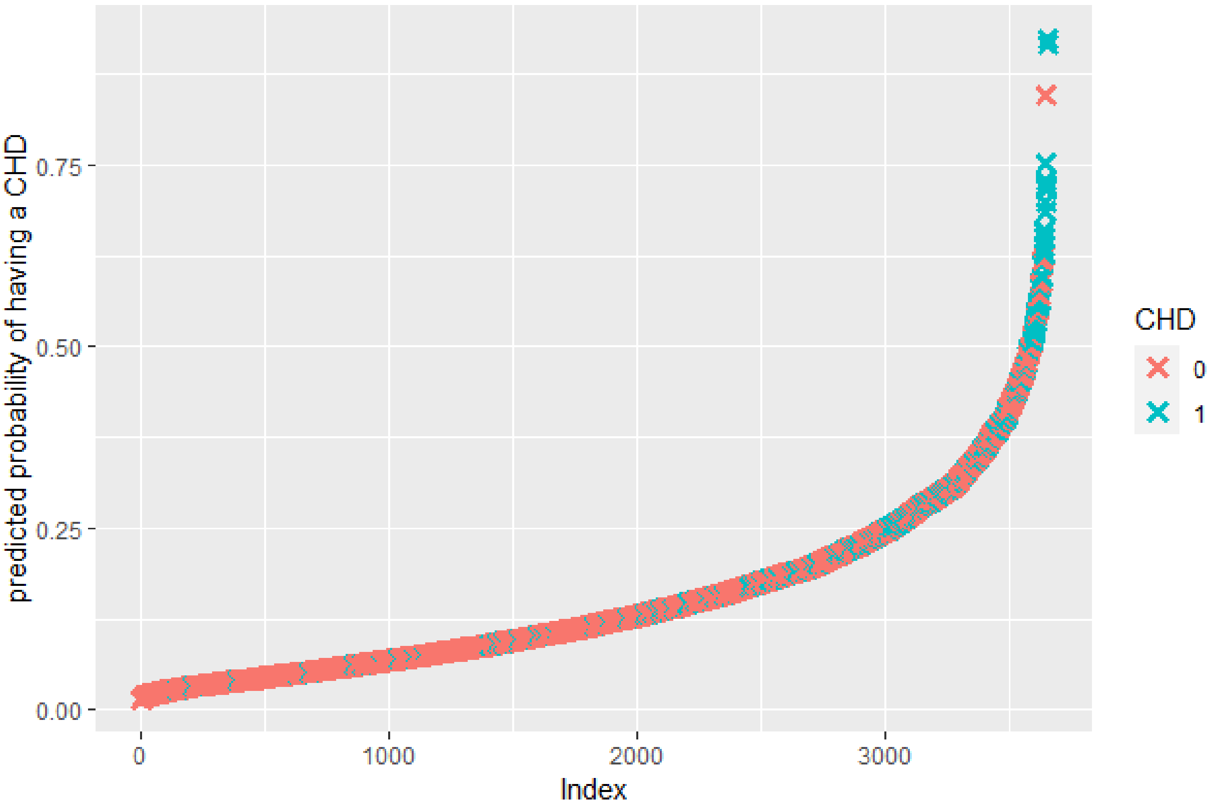
The correlation matrix analysis revealed that there is a possibility of reducing the dimensionality of the dataset. To achieve this, I employed the principal component analysis (PCA) technique to extract important features from the dataset. By using PCA, I was able to identify the underlying patterns and correlations within the dataset, and to reduce the number of variables while retaining the important information. PCA can be used to extract the major important predictor for the response out of the existing predictors. This can be used if there is a low correlation between response and independent predictors and the general analysis is unable to capture the general structure of the data to predict the response in Supervised learning.

Logistic regression:



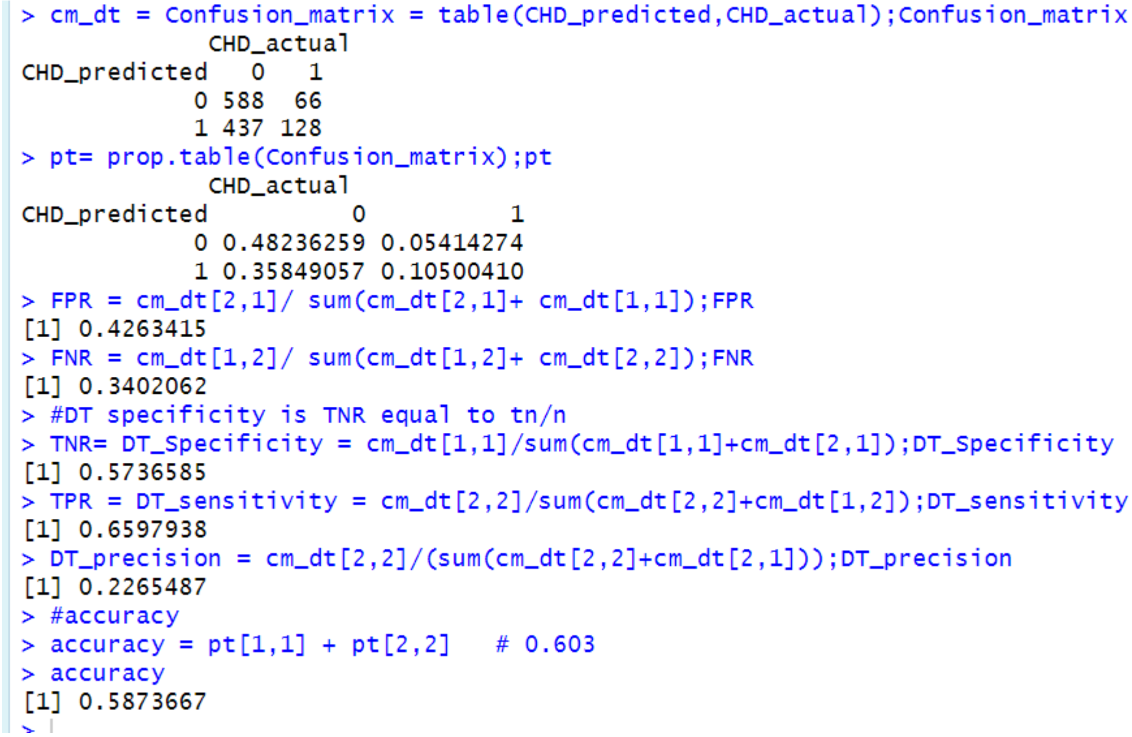
```
Call:
glm(formula = TenYearCHD ~ ., family = "binomial", data = df.nona)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-1.9303  -0.5930  -0.4243  -0.2827   2.8611  

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)    
(Intercept)      -8.258321   0.710243 -11.627  < 2e-16 ***
Sex1              0.534871   0.109914   4.866 1.14e-06 ***
age               0.062166   0.006757   9.201  < 2e-16 ***
education2       -0.192060   0.123431  -1.556  0.11970    
education3       -0.193891   0.150155  -1.291  0.19661    
education4       -0.059869   0.164615  -0.364  0.71609    
currentSmoker1    0.072388   0.156744   0.462  0.64421    
cigsPerDay        0.018020   0.006234   2.891  0.00384 ** 
BPMeds1           0.165049   0.234470   0.704  0.48148    
prevalentStroke1  0.704569   0.491444   1.434  0.15167    
prevalentHyp1     0.233855   0.138213   1.692  0.09065 .  
diabetes1         0.026308   0.316112   0.083  0.93367    
totChol           0.002369   0.001129   2.098  0.03590 *  
sysBP             0.015451   0.003812   4.053 5.05e-05 ***
diaBP            -0.004095   0.006444  -0.636  0.52506    
BMI               0.005149   0.012787   0.403  0.68716    
heartRate        -0.003007   0.004213  -0.714  0.47533    
glucose           0.007212   0.002234   3.228  0.00125 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3120.5  on 3655  degrees of freedom
Residual deviance: 2751.9  on 3638  degrees of freedom
AIC: 2787.9

Number of Fisher Scoring iterations: 5
```


predicted probability of having a CHD
0.00
0.25
0.50
0.75
0
1000
2000
3000
Index
CHD
0
1


predicted probability of having a CHD
0.00
0.25
0.50
0.75
0
1000
2000
3000
Index
CHD
0
1

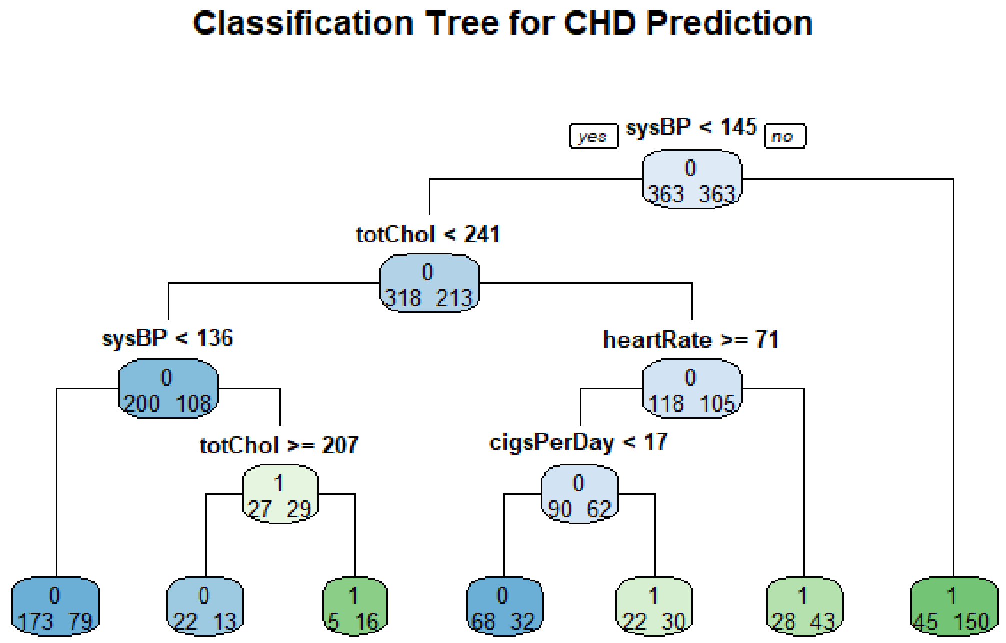
2. Decision trees:

Below are the results of Decision Tree with Under sampling and Feature Sampling.



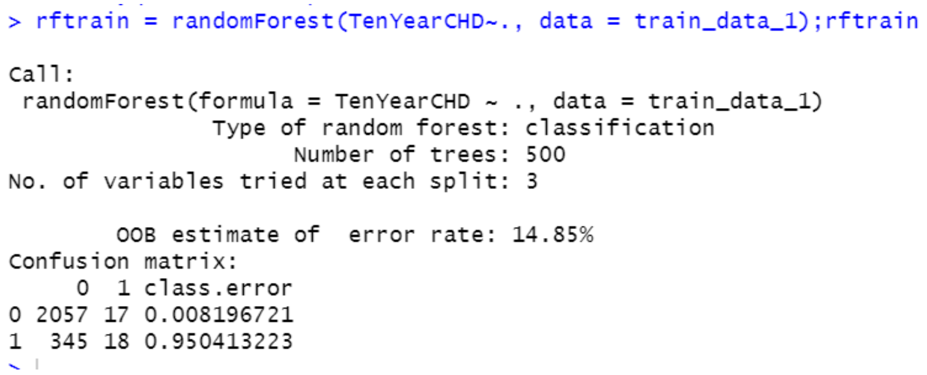
```
> cm_dt = Confusion_matrix = table(CHD_predicted,CHD_actual);Confusion_matrix
             CHD_actual
CHD_predicted   0   1
            0 588  66
            1 437 128
> pt= prop.table(Confusion_matrix);pt
             CHD_actual
CHD_predicted          0          1
            0 0.48236259 0.05414274
            1 0.35849057 0.10500410
> FPR = cm_dt[2,1]/ sum(cm_dt[2,1]+ cm_dt[1,1]);FPR
[1] 0.4263415
> FNR = cm_dt[1,2]/ sum(cm_dt[1,2]+ cm_dt[2,2]);FNR
[1] 0.3402062
> #DT specificity is TNR equal to tn/n
> TNR= DT_Specificity = cm_dt[1,1]/sum(cm_dt[1,1]+cm_dt[2,1]);DT_Specificity
[1] 0.5736585
> TPR = DT_sensitivity = cm_dt[2,2]/sum(cm_dt[2,2]+cm_dt[1,2]);DT_sensitivity
[1] 0.6597938
> DT_precision = cm_dt[2,2]/(sum(cm_dt[2,2]+cm_dt[2,1]));DT_precision
[1] 0.2265487
> #accuracy
> accuracy = pt[1,1] + pt[2,2]   # 0.603
> accuracy
[1] 0.5873667
> 
```

Below is the Classification Diagram of the above Decision Tree Prediction



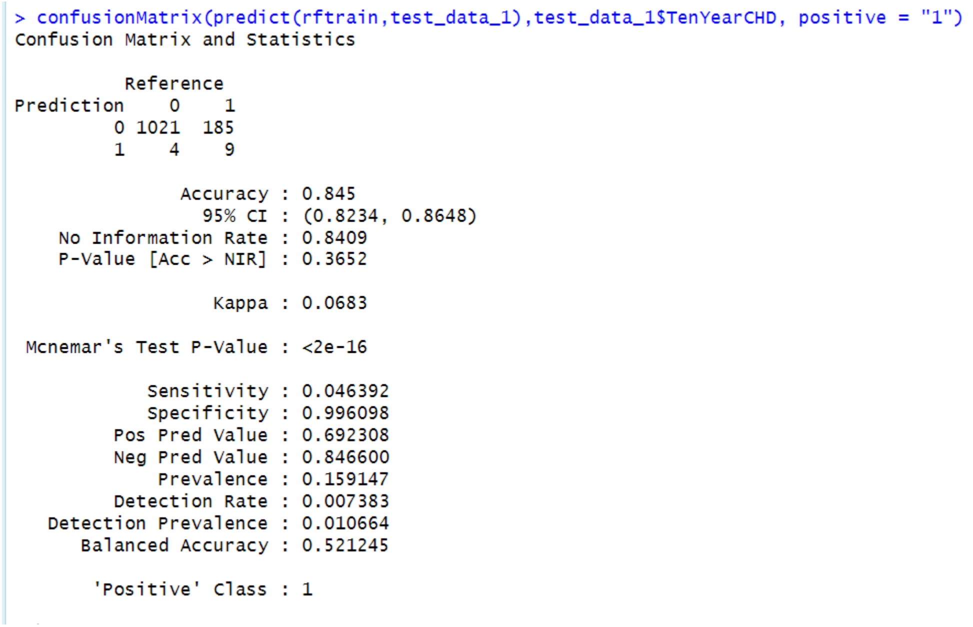
3. Random forests:

Below are the results of Random Forest technique performed over normal training data.



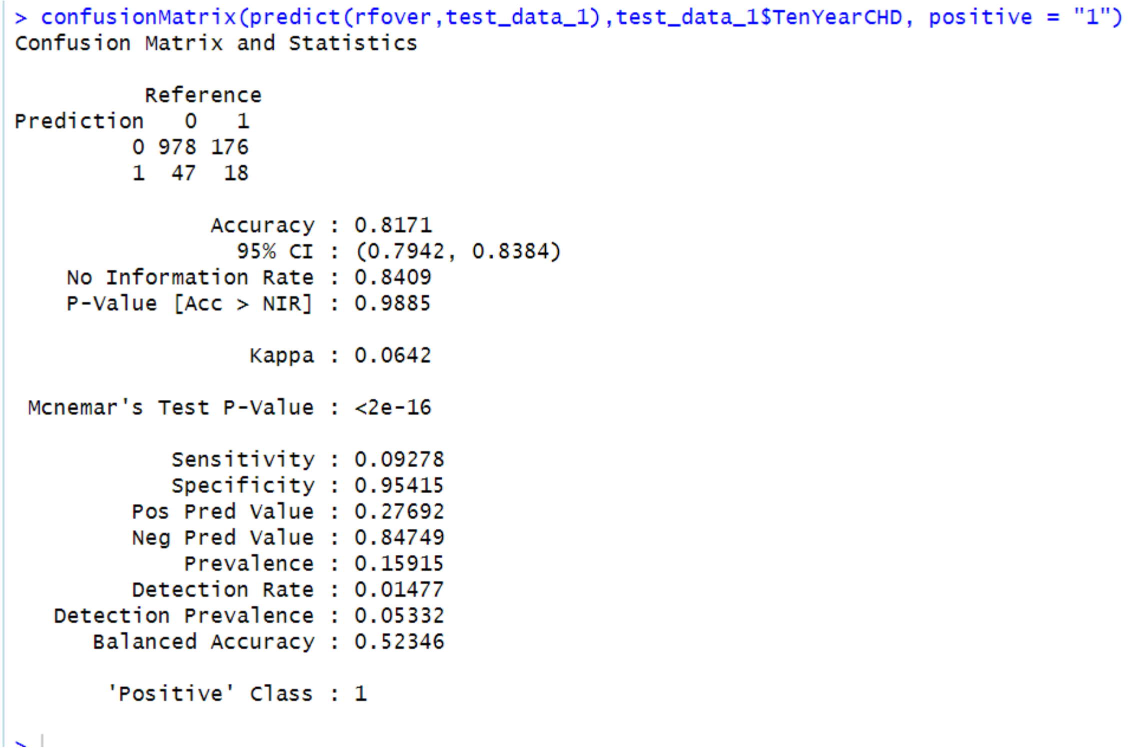
```
> rftrain = randomForest(TenYearCHD~., data = train_data_1);rftrain

Call:
 randomForest(formula = TenYearCHD ~ ., data = train_data_1)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 3

        OOB estimate of  error rate: 14.85%
Confusion matrix:
     0  1 class.error
0 2057 17 0.008196721
1  345 18 0.950413223
```

Results of Random Forest technique performed over normal training data.



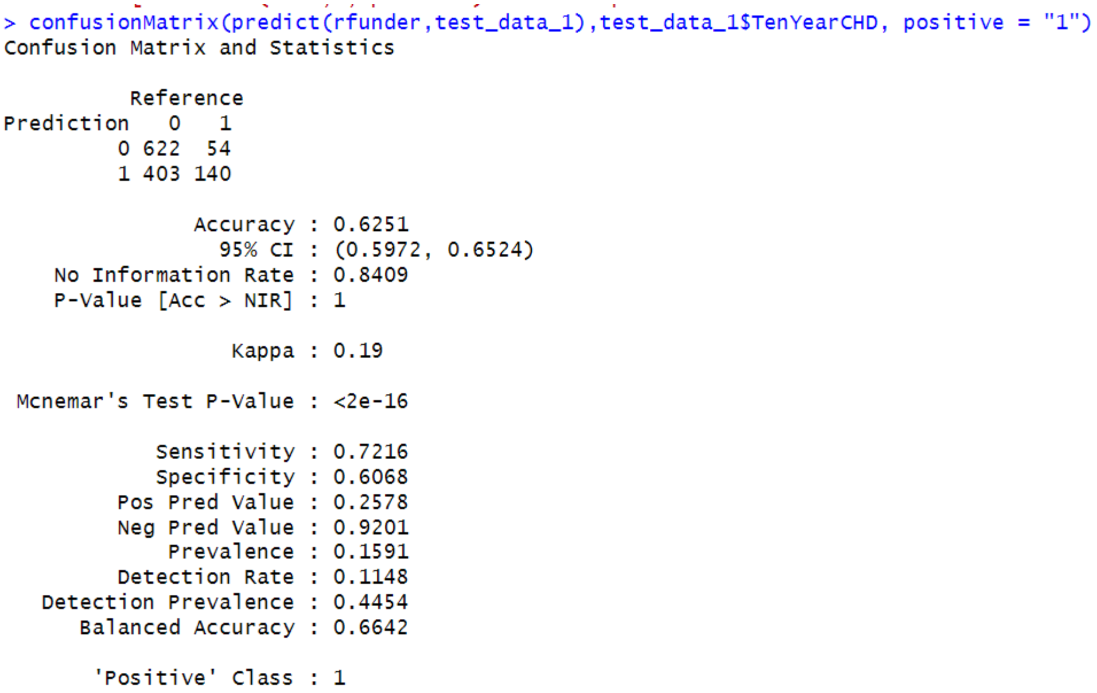
```
> confusionMatrix(predict(rftrain,test_data_1),test_data_1$TenYearCHD, positive = "1")
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1021  185
         1    4    9

               Accuracy : 0.845
                 95% CI : (0.8234, 0.8648)
    No Information Rate : 0.8409
    P-Value [Acc > NIR] : 0.3652

                  Kappa : 0.0683

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.046392
            Specificity : 0.996098
         Pos Pred Value : 0.692308
         Neg Pred Value : 0.846600
             Prevalence : 0.159147
         Detection Rate : 0.007383
   Detection Prevalence : 0.010664
      Balanced Accuracy : 0.521245

       'Positive' Class : 1
```

Results of Random Forest technique performed over Test data of Oversampled data.



```
> confusionMatrix(predict(rfover,test_data_1),test_data_1$TenYearCHD, positive = "1")
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 978 176
         1  47  18

               Accuracy : 0.8171
                 95% CI : (0.7942, 0.8384)
    No Information Rate : 0.8409
    P-Value [Acc > NIR] : 0.9885

                  Kappa : 0.0642

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.09278
            Specificity : 0.95415
         Pos Pred Value : 0.27692
         Neg Pred Value : 0.84749
             Prevalence : 0.15915
         Detection Rate : 0.01477
   Detection Prevalence : 0.05332
      Balanced Accuracy : 0.52346

       'Positive' Class : 1

>
```

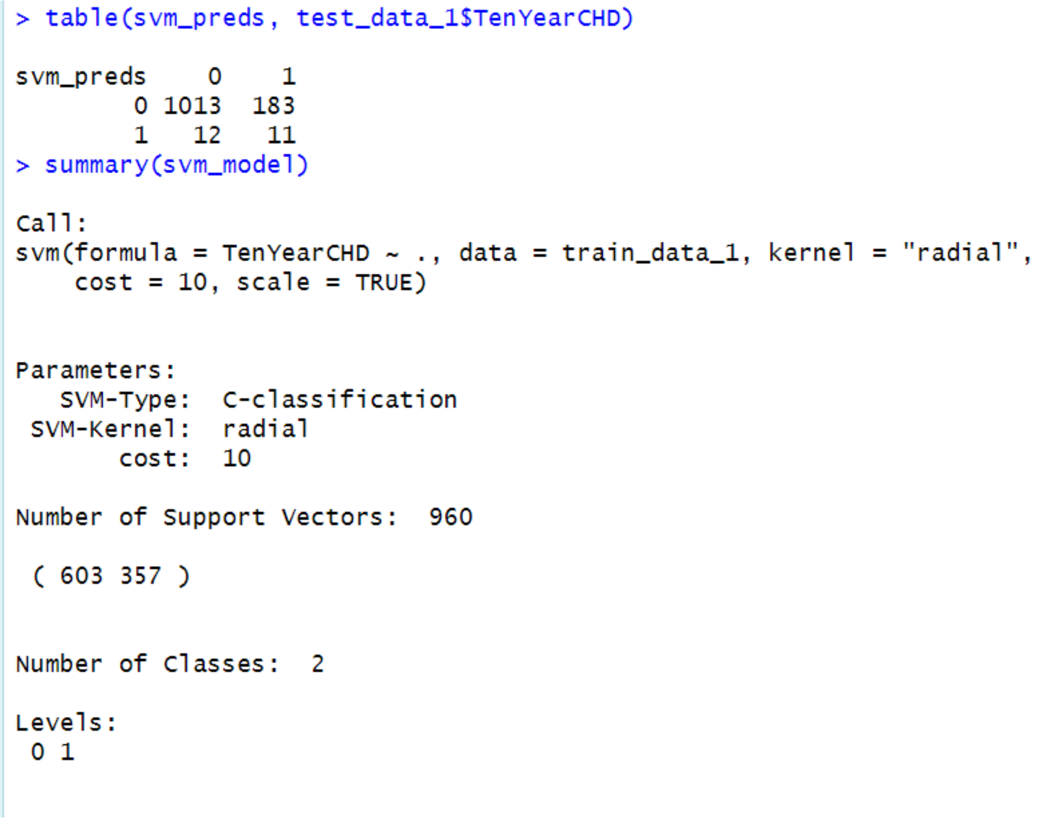
Results of Random Forest technique performed over Test data of Undersampled data



```
> confusionMatrix(predict(rfunder,test_data_1),test_data_1$TenYearCHD, positive = "1")
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 622  54
         1 403 140

               Accuracy : 0.6251
                 95% CI : (0.5972, 0.6524)
    No Information Rate : 0.8409
    P-Value [Acc > NIR] : 1

                  Kappa : 0.19

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.7216
            Specificity : 0.6068
         Pos Pred Value : 0.2578
         Neg Pred Value : 0.9201
             Prevalence : 0.1591
         Detection Rate : 0.1148
   Detection Prevalence : 0.4454
      Balanced Accuracy : 0.6642

       'Positive' Class : 1
```

4. SVM:

The following are the prediction results using Support Vector Machine technique. Of the all kernels - linear, polynomial, Sigmoid and radial, the radial kernel was better at predicting the risk of heart disease. But through SVM, I was not able to predict whether the people really had a risk of coronary heart disease or not.



```
> table(svm_preds, test_data_1$TenYearCHD)

svm_preds    0    1
        0 1013  183
        1   12   11
> summary(svm_model)

Call:
svm(formula = TenYearCHD ~ ., data = train_data_1, kernel = "radial",
    cost = 10, scale = TRUE)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  10

Number of Support Vectors:  960

 ( 603 357 )


Number of Classes:  2

Levels:
 0 1
```

Results and Analysis:

The sensitivity of Random Forest is 0.5928 which is best among the rest of the models taking recall and precision into consideration. The Random Forest model was improved after involving the feature selection and balancing techniques. So, I took under sampled data for random forest model as best Model.

Conclusion:

Coronary heart disease is a serious health concern that affects a significant portion of the population, but recent research has identified patterns and risk factors that can help to predict future risk.

Using machine learning algorithms and techniques such as dimensional reduction and under sampling, this study was able to identify key health factors strongly associated with CHD and develop an accurate classification model to predict 10-year risk.

The results showed that the random forest algorithm outperformed other classification models in terms of sensitivity, recall, and precision, suggesting that this model could be a valuable tool for identifying patients at risk for CHD.

By providing early intervention or prevention methods, this model has the potential to help improve patient outcomes and reduce the burden of CHD on the healthcare system.

Above all the Models used in random forest have given us better results in terms of sensitivity, recall and precision.