Heart Disease Classification Using Personal Key Indicators

(GR5291 Group Final Project)

Written By: Ningxin Xu (nx2160@columbia.edu);

Hui Li (hl3537@columbia.edu);

Yu Cheng (yc4026@columbia.edu);

Ziyu Liu (zl3060@columbia.edu);

Zizhou Wu (zw2676@columbia.edu)

Abstract

In this report, we used the data of “Personal Key Indicators of Heart Disease” found in Kaggle (https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease) in order to analyze how people could self-test whether they are potential to have heart disease by asking some personal questions such as “do you drink alcohol”, “do you ever have stroke”, “do you feel difficult in walking” etc.

There would be five parts in the report. Firstly, we would apply data visualization tools to have a direct look of the relationship between heart disease and each variable. Then, we mainly compare seven machine learning models: Logistic Regression, K-Nearest Neighbors, Support Vector Machines, Random Forest, Naive Bayes Classification, Gradient Boosting and Decision Tree. After comparison, we would like to find the most suitable model for this data and use this model to do further analysis about this data. In the Conclusion, we would state out the most important features in the model selected and propose some suggestions regarding how to self-test and how to reduce the potential of getting heart disease.

Part I Introduction

Heart disease is a collective term for various diseases that affect the structure of the heart and the way it works. Lloyd-Jones mentioned that although mortality from coronary heart disease has declined dramatically over the past 30 years, it remains the single leading cause of death among adults worldwide and is expected to remain the leading cause of death and disability in the Western world for a long time in the 21st century. (1999)

Therefore, we decide to examine what factors contribute to or have an impact on coronary heart disease (CHD) or myocardial infarction (MI), which may include many of the dominant factors associated with smoking, alcohol consumption, and stroke. We also want to analyze the association between heart disease and other recessive factors including body mass index, gender and age. After searching and analyzing, we finally chose the Personal Key Indicators of Heart Disease dataset from Kaggle, which includes data from 2020, for our study. The vast majority of the columns are questions asked to respondents about their health status.

Our goal is to examine the correlation between the factors and find a suitable model to describe the relationship between heart disease and the factors, to detect the factors that have a significant impact on heart disease, to draw conclusions based on the final results and to provide recommendations to reduce mortality from coronary heart disease (CHD) or myocardial infarction (MI).

Part II Exploratory Data Analysis

After we obtain the dataset from kaggle, we firstly take a look at the data itself, such as its variable type, the summary of each variable, then we check whether there are null values in the data. After summarizing the data, it consists of 319,795 rows and 18 columns. No N/A values are found in the data, so we could directly use it to the next data visualization part. Meanwhile, there are many binary variables for “have you ever” questions, just like “HeartDisease”, “Smoking” and “Stroke”, so in order for easier and better modeling, we convert them into logistic values (1/0). Also for categorical variables, such as “AgeCategory”, “GenHealth” and “Race”, we decide to apply the same logic that converts each category into a numerical group.

Name	Description
HeartDisease	Respondents that have ever reported having coronary heart disease (CHD) or myocardial infarction (MI). (Yes=1; No=0)
GenHealth	General evaluation of individual health: poor = 0, fair = 1, good = 2, very good = 3, excellent = 4
AgeCategory	Combine original 14 levels into 4 levels: 18 - 34 = 0;35 - 49 = 1; 50 - 64 = 2; 64 - older = 3
DiffWalking	Do you have serious difficulty walking or climbing stairs? (Yes=1; No=0)
Stroke	(Ever told) (you had) a stroke?(Yes=1; No=0)
PhysicalHealth	Physical health includes physical illness and injury,and for how many days during the past 30 days was your physical health not good? (range from 0 to 30)
KidneyDisease	Not including kidney stones, bladder infection or incontinence, were you ever told you had kidney disease?(Yes=1; No=0)
Smoking	Have you smoked at least 100 cigarettes in your entire life? (Yes=1; No=0)
PhysicalActivity	Adults who reported doing physical activity or exercise during the past 30 days other than their regular job.(Yes=1; No=0)
SkinCancer	(Ever told) (you had) skin cancer? (Yes=1; No=0)
Sex	male(1) or female(0)
BMI	Body Mass Index (range from 12.02 to 94.85)
Asthma	(Ever told) (you had) asthma? (Yes=1; No=0)
MentalHealth	How many days during the past 30 days was your mental health not good? (range from 0 to 30)
AlcoholDrinking	Heavy drinkers (adult men having more than 14 drinks per week and adult women having more than 7 drinks per week (Yes=1; No=0)
SleepTime	On average, how many hours of sleep do you get in a 24-hour period? (range from 1 to 24)

Table 1: Data Description

After we have numerical groups for all variables, we group each variable by whether they get heart disease and calculate the mean values. In this way, we are able to conclude that people with heart disease are always older and have a higher BMI, higher chance of smoking and stroke, and also more days of not feeling good. Also, they suffer more when walking or stair climbing; the probability of having kidney disease and skin cancer increases too.

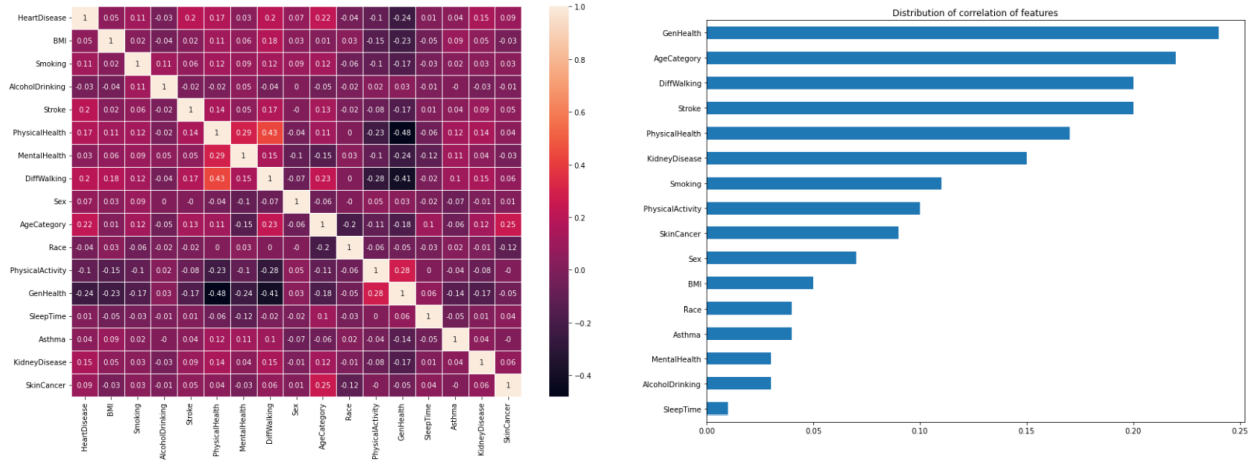

Figure 1: Correlation Matrix and Its Distribution

Moreover, we plot a correlation matrix to see the relationship between the predictors and the dependent variable ("HeartDisease") and whether there exists the multicollinearity problems in the predictors. We also sort the variables according to their correlations with "HeartDisease" descendingly. From the figure above, we cannot tell strong correlations between HeartDisease and other variables; "GenHealth" has the highest correlation of 0.24. Between predictors, some correlations are relatively high, such as the correlation between "DiffWalking" and "PhysicalAcitivity" is 0.28 and we think it is predictable since people with difficulty in waking can hardly have physical activities daily. After consideration, the variables that have less than 0.05 correlation with "heartDisease" are dropped and we will only contain remaining variables when fitting the model.

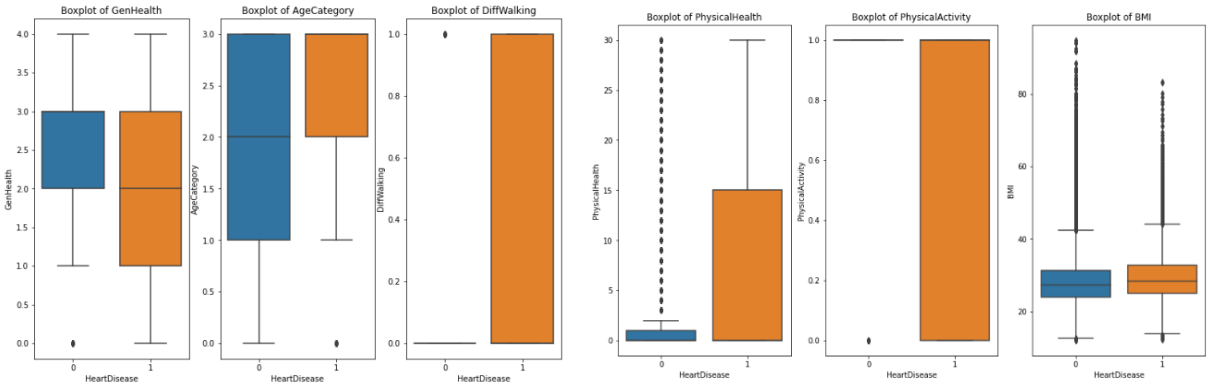

Figure 2: Boxplot of Each Variable Related to Heart Disease

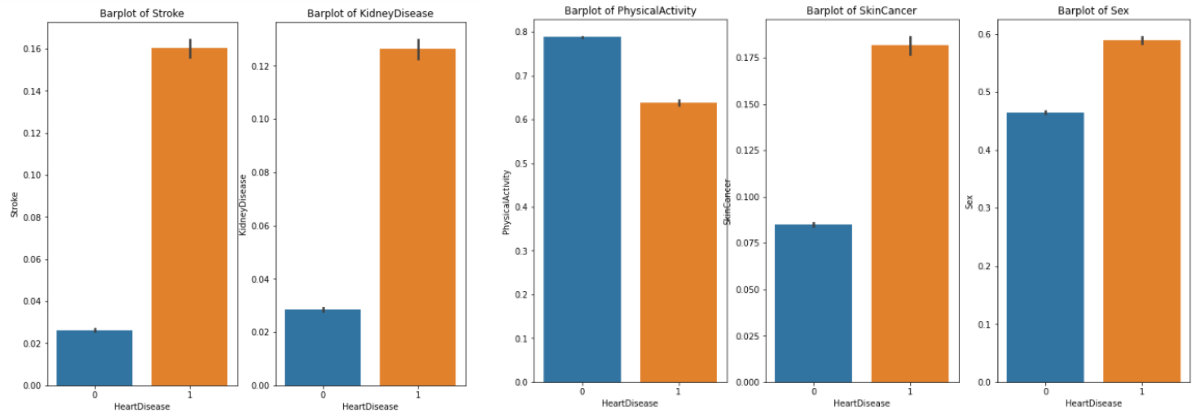

Figure 3: Barplot of Variables Related to Heart Disease

While visualizing the data, we firstly make some boxplots grouped by heart disease. From those plots, we could see that people with a higher score or answer yes on questions about the general health score, age, difficulty in waking, physical health and physical activity definitely have a higher chance to have heart disease. Besides, for some variables that cannot be telled from boxplots, we plot barplots to have a closer look at the relationship between heart disease and non heart disease. Thus, from the barplots, people with heart disease are more likely to have stroke, kidney disease and skin cancer, but exercise less than people who do not have heart disease.

What is also worth mentioning is that males have a higher probability to have heart disease than females. Comparing the two distributions below, though most of the distribution overlaps, the center of the distribution of heart disease is larger.

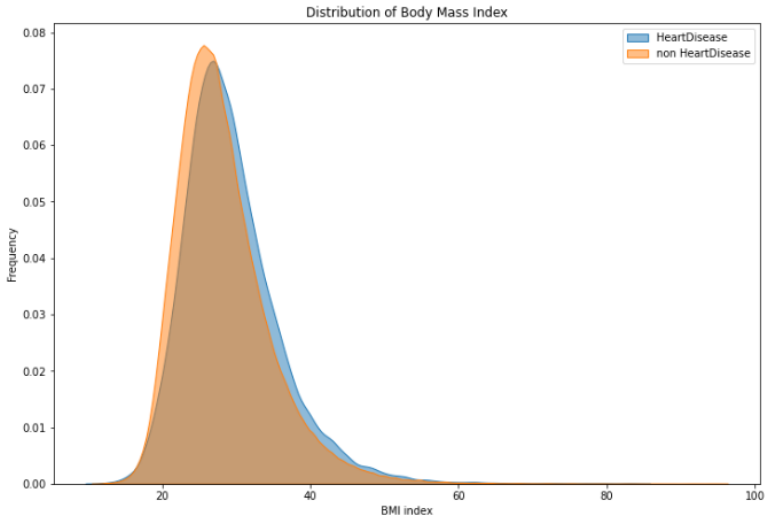

Figure 4: Distribution of BMI

At last, we calculate the proportion of heart disease in the data, which is only 0.09 and the proportion of non heart disease cases has reached 0.9, which means this data set is highly unbalanced. Thus, we may want the ratio of heart disease and non heart disease to be 1:1. We sampled the same number of heart disease cases in non heart disease cases and combined these two groups into one new balanced data set. In the following parts, we are consistently using the new data set.

Part III model comparison

In this part, we compared seven machine learning models to see how each model works for the data using statistics like accuracy and AUC.

1. Logistic Regression

Logistic regression allows the analysis of dichotomous or binary outcomes with 2 mutually exclusive levels(LaValley, 2008). Logistic regression allows the use of continuous or categorical predictors and provides the ability to adjust for multiple predictors. And after we filtered and adjusted the balance, we selected twelve variables, all of which belong to numerical variables. This makes logistic regression particularly useful for our analysis of observational data, especially when adjustments are needed to reduce the potential bias caused by differences between the groups being compared. Therefore, we decided to try to use logistic regression to speculate on the presence or absence of an effect of these variables on heart disease based on the values of the predictor variables.

Then we split our data set into a training set and a Test set. The training set is 2/3 of the total data set, and the test set is 1/3. We randomly generated partitions with the dependent variable HeartDisease as the condition to build a logistic regression. We next fit this model:

```
Call:
glm(formula = HeartDisease ~ GenHealth + AgeCategory + DiffWalking +
    Stroke + PhysicalHealth + KidneyDisease + Smoking + PhysicalActivity +
    SkinCancer + Sex + BMI, family = "binomial", data = Train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2015  -0.8151  -0.0390   0.8249   2.8416

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)      -1.865257   0.093497 -19.950  < 2e-16 ***
GenHealth        -0.581678   0.015072 -38.594  < 2e-16 ***
AgeCategory       0.904869   0.015810  57.234  < 2e-16 ***
DiffWalking       0.335049   0.036141   9.271  < 2e-16 ***
Stroke            1.289966   0.055982  23.042  < 2e-16 ***
PhysicalHealth    0.004697   0.001668   2.815  0.00488 **
KidneyDisease     0.759939   0.057016  13.329  < 2e-16 ***
Smoking           0.404070   0.025949  15.572  < 2e-16 ***
PhysicalActivity -0.047993   0.030634  -1.567  0.11719
SkinCancer        0.280050   0.038756   7.226 4.97e-13 ***
Sex               0.690955   0.026317  26.255  < 2e-16 ***
BMI               0.011775   0.002120   5.554 2.79e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 50849  on 36679  degrees of freedom
Residual deviance: 36789  on 36668  degrees of freedom
AIC: 36813
```

Table 2: Logistic Regression Model

The p-value for the predictor variable “PhysicalActivity” is 0.20739. Since this value is not less than .05, it does not have a statistically significant relationship with the response variable "HeartDisease" in the model. However, the p-value of other variables are all higher than 0.5, which means they have statistically significant relationships with the response variable "HeartDisease" in the model.

Finally, we performed an accuracy test on this logistic regression:

```
Test <- Test  %>% mutate(model_pred = 1*(model_prob > .53) + 0,
                                 HeartDisease_Binary = 1*(HeartDisease == 1) + 0)

Test <- Test %>% mutate(accurate = 1*(model_pred == HeartDisease_Binary))
sum(Test$accurate)/nrow(Test)
```

```
## [1] 0.7534595
```

Table 3: Code For Accuracy Calculation

The accuracy of logistic regression is 75.35% and we also calculated the AUC, which is 0.83.

```
#calculate AUC
library(pROC)
auc(Test$HeartDisease, predicted)

```
```

```
Setting levels: control = 0, case = 1
Setting direction: controls < cases
Area under the curve: 0.8301
```

Table 4: Code For AUC Calculation

2. K-Nearest Neighbors

The k-nearest neighbors (KNN) algorithm is a data classification method for estimating the likelihood that a data point will become a member of one group or another based on what group the data points nearest to it belong to (Joby, 2021). To be more specific, KNN is based on the
```

similarity of features to check how similar a data point is to its neighbors and eventually classify the data point to its most similar class.

After we have collected and balanced the dataset, we start with the import dataset. According to the structure of the dataset shown below we note the following points: the dataset consists of 54746 observations and 12 variables; 'HeartDisease' variable is our output variable, its value represents whether the observer has heart disease or not. Also, we note that the “PhysicalHealth” variable and “BMI” variable have a 2-digit numerical scale, while the other variables are single-digit. If the data is not normalized, it can lead to a biased outcome, so we normalize the dataset to ensure that the output remains unbiased.

```
## 'data.frame':    54746 obs. of  12 variables:
##  $ HeartDisease    : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ GenHealth       : int  1 0 0 2 1 0 2 0 1 1 ...
##  $ AgeCategory     : int  3 2 3 3 3 3 2 3 3 3 ...
##  $ DiffWalking     : int  1 1 1 1 1 0 0 0 1 1 ...
##  $ Stroke          : int  0 0 1 0 0 0 0 0 0 0 ...
##  $ PhysicalHealth  : num  6 30 10 0 6 3 0 30 30 2 ...
##  $ KidneyDisease   : int  0 0 0 0 1 0 0 1 0 0 ...
##  $ Smoking         : int  1 1 1 0 1 1 0 1 1 0 ...
##  $ PhysicalActivity: int  0 0 1 0 1 0 1 1 1 0 ...
##  $ SkinCancer      : int  0 0 1 1 0 0 0 0 1 0 ...
##  $ Sex             : int  0 1 1 0 0 0 0 0 1 0 ...
##  $ BMI             : num  28.9 34.3 33 25.1 30.2 ...
```

Table 5: Normalized Variables

Next, data splicing is performed after the dataset is normalized. We need to split the dataset into training and testing datasets. We randomly select 70% data from the dataset, where 70% training data, remaining 30% is test data. After deriving the training and testing datasets, a separate data frame is created for the "HeartDisease" variable so that our final results can be compared with the actual values. Then, we have to initialize the 'K' values in the KNN model. One way to find the best K value is to calculate the square root of the total number of observations in the dataset.

This square root will give the 'K' value. We have 38322 observations in our training dataset and the square root of 38322 is about 195.76, so we will create two models. One with a 'K' value of 195 and the other model with a 'K' value of 196.

After building the model, we can calculate the accuracy of the KNN model with K we choose:

```
#Calculate the proportion of correct classification for K = 195 and K= 196
ACC.195 <- 100 * sum(test.HD_labels == knn.195)/NROW(test.HD_labels)
ACC.195
```

```
## [1] 74.26936
```

```
ACC.196 <- 100 * sum(test.HD_labels == knn.196)/NROW(test.HD_labels)
ACC.196
```

```
## [1] 74.33025
```

Table 6: Code For Accuracy with K-value

As shown above, the accuracy for K = 195 is 74.27% and for K = 196 is 74.33%.

In general, the KNN algorithm is highly unbiased in nature, it is simple and effective. However, it does not create a model because there is no abstraction process involved. Therefore, the accuracy of k-nearest neighbors(KNN) is 74.33% with K = 196.

3. Support Vector Machines

support-vector machines (SVMs, also support-vector network) are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier (although methods such as Platt scaling exist to use SVM in a probabilistic classification setting).

Under SVM model and based on our data, a for loop function is used to find the best model to fit our dataset, i.e. to find a model with the lowest error rate and the highest accuracy score.In the process of tuning the hyperparameter, we found that the error rate is minimized when C=1 which equals to 0.2446 approximately.

```
C=1,Error Rate = 0.24456621004566215
C=2,Error Rate = 0.24502283105022826
C=3,Error Rate = 0.24511415525114155
C=4,Error Rate = 0.24511415525114155
C=5,Error Rate = 0.2466666666666667
C=6,Error Rate = 0.24566210045662096
C=7,Error Rate = 0.24611872146118718
C=8,Error Rate = 0.246027397260274
C=9,Error Rate = 0.24611872146118718
C=10,Error Rate = 0.246027397260274
```

Table 7: Error Rate of Each C-value

Also, based on the K-fold cross validation, we can get the accuracy measured using 10-Fold Cross Validation is 75.55% with std.deviation 0.64% using SVC model with RBF kernel. What's more, using the confusion matrix also can help us see more clearly how well the data fit the model. Using an ROC curve to understand the diagnostic value of the test. And we also know that AUC = 0.86.

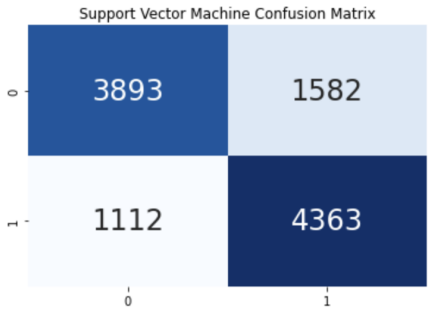

Figure 5: SVM Confusion Matrix

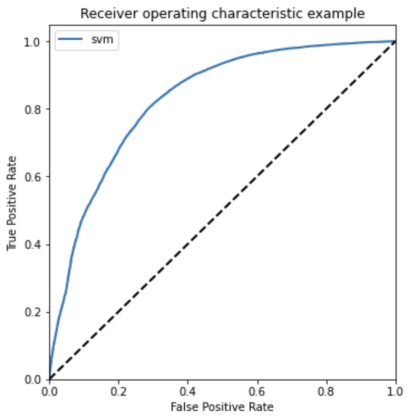

Figure 6: SVM ROC Curve

4. Gradient Boosting

Gradient Boosting, as known as a supervised machine learning algorithm, could be used for both classification and regression problems. As shown in the figure below, the idea of boosting is whether a weak learner could be modified to be better in the next level, while remaining the existing model unchanged and minimizing the overall errors after adding the new features (Kumar, 2020).

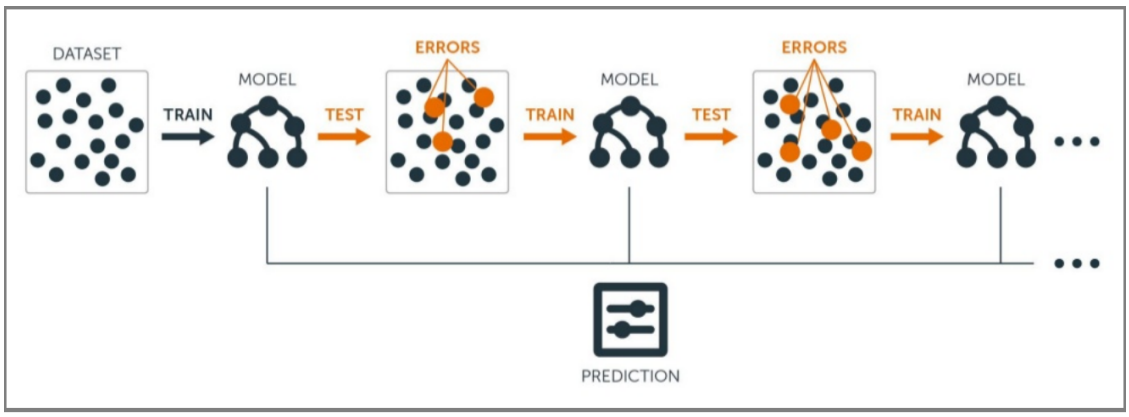

Figure 7: GBC Working Process

In application, Gradient Boosting involves three elements: 1. a loss function; 2. a weak learner to make predictions; 3. an addictive model to add weak learners to minimize the loss function (Kumar, 2020). The loss function is basically an algorithm depending on the problem data needed to be solved; the weak learners, in most cases, are decision trees; and the addictive model means adding one tree at one time and the existing trees remain the same.

Here we decided to use Gradient Boosting as one potential model, since it could work with both categorical and numerical data (we have a combination of those two types of data). Moreover, this model always provides a higher predictive accuracy compared with some other classifier

models, like Random Forest. However, it may also raise the problem of overfitting, due to its ability to minimize all errors, which needs us to use cross-validation to neutralize.

max_depth	1	2	3	5	10
mean_train_scores	0.76	0.76	0.76	0.77	0.87
mean_test_scores	0.76	0.76	0.76	0.76	0.74

Table 8: Scores with Different Depth

In our model, we first used a grid search to choose a depth that could produce the highest accuracy scores. As shown in the table, if we choose the maximum depth to be 10, we could gain the highest train score, though the test score decreases with regard to the test score when maximum depth is 5. While fitting the GradientBoosting model, we also set the number of boosting stages to perform (n_estimators) as 100, since it is a fairly robust number, which is not too large to cause the overfitting, but is large enough to result in better performance.

The mean cv score related to our model is 74.67% with its standard deviation 0.013. Besides the score, we also apply the confusion matrix to both the train data and test data to see how well our model performs.

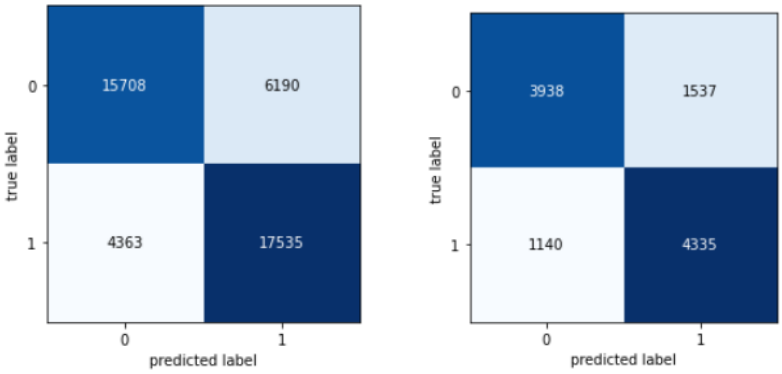

Figure 8: GB Confusion Matrix For Train and Test Data

The matrix on the left is for the train data: the true positive rate (TPR) is the true positive divided by positive, which is 97.43%; the false positive rate (FPR) is the false positive divided by negative, which is 28.27%. The matrix on the right is for the test data: the TPR is 79.18% and its FPR is 28.07%. As we can see, the true positive rate is very high, but also the false positive rate is around 28%, so this model may not have a good performance on non-heart disease. Here is the receiver operator characteristic (ROC) curve, which is an evaluation metric for classification problems. Also, the model has an area under curve (AUC) of 0.816, which is close to one. Thus, overall, the model works pretty well.

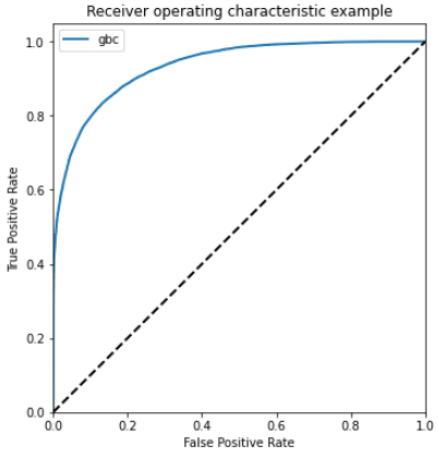

Figure 9: GB ROC Curve

5. Decision Tree

Decision Tree, as similar to Gradient Boosting, is also a type of supervised machine learning algorithm, and it is known as its tree-like structure, which contains decision nodes and leaves: beginning with the root node, the decision nodes are where the data splits and the leaves are the final outcomes, just as the figure below represents (Kurama, 2021).

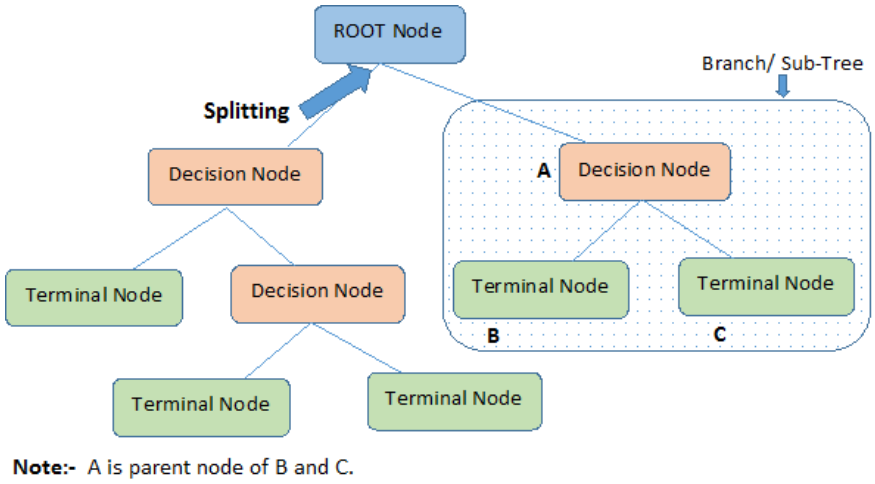

Figure 10: Decision Tree Working Process

Decision Trees are also good for both categorical variables and continuous variables and according to the data type, it could be divided into two types: 1. classification tree and 2. regression tree. For our data, we decided to use Decision Tree Classifier, since we have a categorical decision variable.

Compared to other machine learning models, Decision Tree models are relatively simple to understand and interpret and even though there are nonlinear relationships between parameters, it will not affect the tree performance. Whereas, this model could also lead to the overfitting problem and it will create biased trees if some variables in the data have a really strong relationship with the dependent variable.

max_depth	1	2	3	5	10
mean_train_scores	0.67	0.72	0.72	0.75	0.78
mean_test_scores	0.67	0.72	0.72	0.75	0.75

Table 9: Scores with Different Depth

In this model, we also used a grid search to choose the best depth that could produce the highest accuracy scores. From the table, we decided to use depth equals to 10, which produces the highest train and test score. After fitting the model with other default inputs, we get a mean cv accuracy of 75.03% with standard deviation of 0.010.

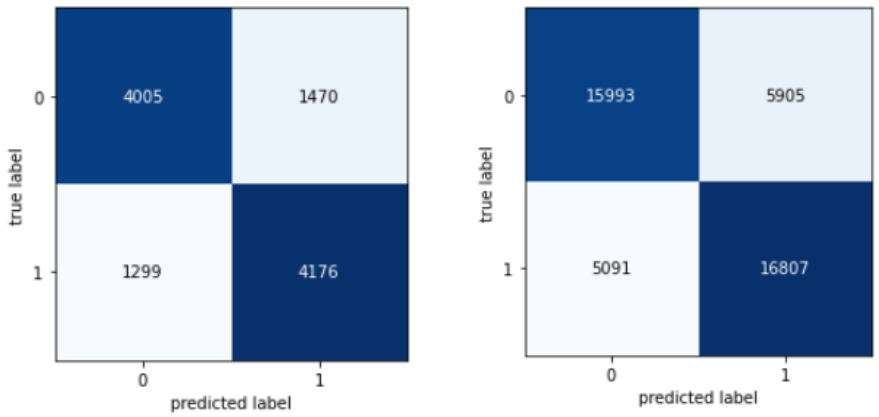

Figure 11: Decision Tree Confusion Matrix

Moreover, we will also check the confusion matrix for both train and test data. The matrix on the left is for the train data: the true positive rate (TPR) is the true positive divided by positive, which is 76.75%; the false positive rate (FPR) is the false positive divided by negative, which is 26.97%. The matrix on the right is for the test data: the TPR is 76.27% and its FPR is 26.85%. Beside the confusion matrix, here is the receiver operator characteristic (ROC) curve, which is an evaluation metric for classification problems. Also, the model also has an area under curve (AUC) of 0.816.

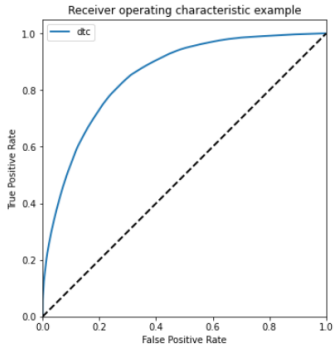

Figure 12: Decision Tree ROC Curve

6. Random Forest

Random Forest is a modeling method for classification and regression by constructing a number of decision trees using training data and averaging the result to make predictions.

The difference from Decision Trees and Random Forest is that Decision Tree is built on the entire data, however Random Forest Model randomly chooses the observations and variables to build up many distinct decision trees. And the output of each tree votes for the class. The most-voted class can be interpreted as the most influential factor for the model.

We resample the balanced data into 2 groups,train and test, with the proportions in 80% and 20% and specify the number of decision trees to be 100.

```
Call:
 randomForest(formula = HeartDisease ~ ., data = train, ntree = 100,      mtry = 4, importance = TR
               Type of random forest: classification
                     Number of trees: 100
No. of variables tried at each split: 4

        OOB estimate of  error rate: 24.34%
Confusion matrix:
      0     1 class.error
0 15739  6159   0.2812586
1  4500 17398   0.2054982
```

Table 10: Random Forest Model Summary

The above detail is the summary of the random forest model which shows error rates for calculated using predictions from trees that do contain in bootstrap sample. And the OOB error rate is 24.34%.

And in the next step, we look forward to obtaining the details of importance under the random forest model.

	0	1
GenHealth	59.419558	44.349664
AgeCategory	76.492810	136.307647
DiffWalking	31.425305	4.997605
Stroke	87.670853	15.332797
PhysicalHealth	31.255587	4.578031
KidneyDisease	32.734125	-4.096018
Smoking	14.499777	18.499576
PhysicalActivity	8.613355	6.426290
SkinCancer	17.976928	-1.398646
Sex	35.587625	45.112054
BMI	6.815127	11.493145

Table 11: Importance of Each Variable In RF

By the output, we could see that the most significant variable for the indication of non-heart disease is Stroke, and that for indication of heart disease is age.

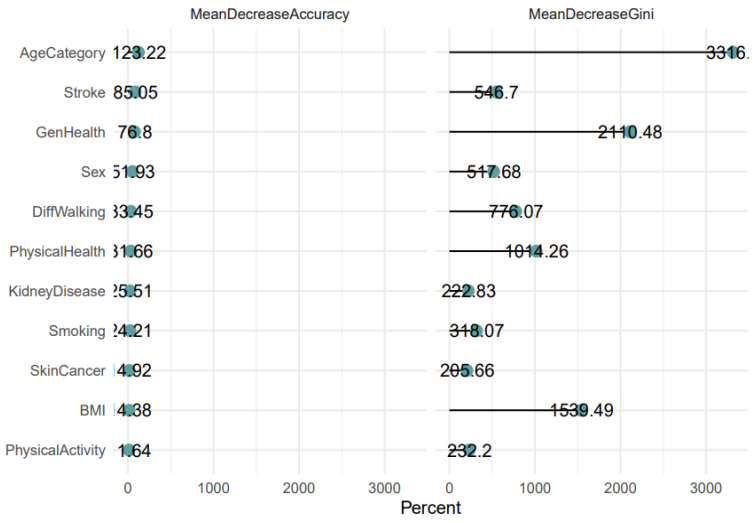

Figure 13: The Plot of Mean Decrease Accuracy and Mean Decrease Gini

The above plot shows the Mean Decrease Accuracy and Mean Decrease Gini. Mean Decrease Accuracy is the value that measures how the accuracy of a model will decrease as removing a specific variable. By the results we could know that the most important variable is AgeCategory.

Given the table and statistical detail of the confusion matrix below, the accuracy is 75.25% for the random forest model. Apart from the confusion matrix, we also take the ROC curve as a criterion to check the accuracy, and AUC is 0.8007.

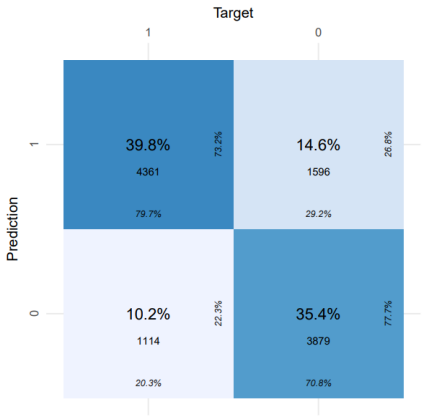

Figure 14: Random Forest Confusion Matrix

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 3879 1114
         1 1596 4361

               Accuracy : 0.7525
                 95% CI : (0.7443, 0.7606)
    No Information Rate : 0.5
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.505

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.7085
            Specificity : 0.7965
         Pos Pred Value : 0.7769
         Neg Pred Value : 0.7321
             Prevalence : 0.5000
         Detection Rate : 0.3542
   Detection Prevalence : 0.4560
```

Table 12: Confusion Matrix Statistics

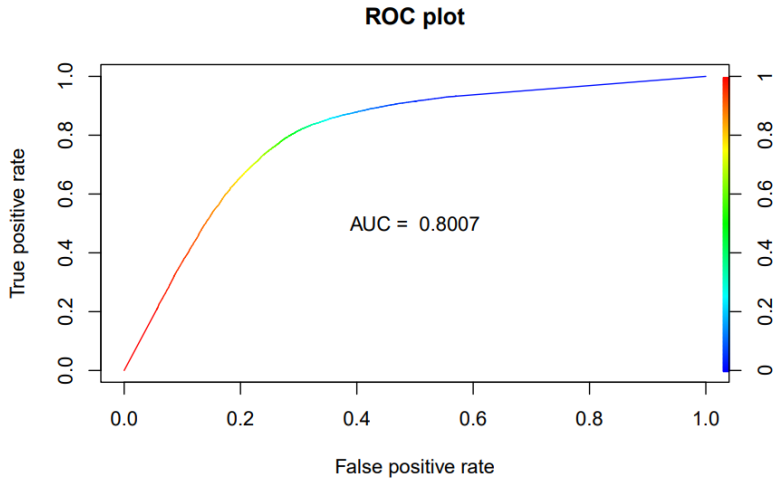

Figure 15: Random Forest ROC Curve

7. Naive Bayes

A Naive Bayes classifier is a probabilistic machine learning model that's used for classification tasks. The crux of the classifier is based on the Bayes theorem (Gandhi, 2018), which is $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$ in short. In the Naive Bayes Modeling process, we also split balanced data into two sub-datasets, train and test with proportions 80% and 20%.

```
================================ Naive Bayes ================================

 Call:
naive_bayes.formula(formula = HeartDisease ~ ., data = train,
    usekernel = T)

-----------------------------------------------------------------------------

Laplace smoothing: 0

-----------------------------------------------------------------------------

 A priori probabilities:

  0   1
0.5 0.5
```

Table 13: Naive Bayes Model Summary

```
 Tables:

-----------------------------------
 ::: GenHealth (Categorical)
-----------------------------------

GenHealth          0          1
        0 0.02461412 0.13987579
        1 0.09338752 0.25979542
        2 0.28500320 0.34697233
        3 0.37172345 0.19960727
        4 0.22527171 0.05374920
```

Table 14: Genhealth

The prior probabilities of having heart disease and not having heart disease is 0.5, and the R program also provides the conditional probability for each feature separately.

The probability of people having heart disease given 2 levels of GenHealth is highest, and The probability of people not having heart disease given 3 levels of GenHealth is highest. We could draw the conclusion that GenHealth plays a role for heart disease in different levels, but in an obscure way.(because the highest probability is not extreme far from others)

```
--------------------------------------

AgeCategory          0          1
          0 0.19042835 0.01790118
          1 0.21170883 0.05566718
          2 0.27879258 0.25244315
          3 0.31907023 0.67398849

--------------------------------------
 ::: DiffWalking (Bernoulli)
--------------------------------------

DiffWalking          0          1
          0 0.8824094 0.6344872
          1 0.1175906 0.3655128

--------------------------------------
 ::: Stroke (Bernoulli)
--------------------------------------

Stroke          0          1
     0 0.9734679 0.8388437
     1 0.0265321 0.1611563
```

Table 15: Variable Summary 1

```
KidneyDisease          0          1
            0 0.97301123 0.87423509
            1 0.02698877 0.12576491

------------------------------------------------------------
 ::: Smoking (Bernoulli)
------------------------------------------------------------

Smoking          0          1
      0 0.6077267 0.4145584
      1 0.3922733 0.5854416

------------------------------------------------------------
 ::: PhysicalActivity (Bernoulli)
------------------------------------------------------------

PhysicalActivity          0          1
               0 0.2102932 0.3623162
               1 0.7897068 0.6376838

------------------------------------------------------------
 ::: SkinCancer (Bernoulli)
------------------------------------------------------------

SkinCancer          0          1
         0 0.91866837 0.81938990
         1 0.08133163 0.18061010

------------------------------------------------------------
 ::: Sex (Bernoulli)
------------------------------------------------------------

Sex          0          1
  0 0.5300484 0.4109508
  1 0.4699516 0.5890492
```

Table 16: Variable Summary 2

For AgeCategory, we could easily make a conclusion that the older people are, the more likely they have heart disease based on the probability of people having heart disease given AgeCategory in level 3 is extreme and highest. DiffWalking, Stroke, and KidneyDisease are similar, as the factor is No(0), people are more likely to be healthy. And as the factor is Yes(1), the probability of having heart disease is higher than that of not having heart disease. As for the factor Smoking, we could observe that the probability of non-smoking people not having heart disease is 0.6064, and the probability of smoking people having heart disease is 0.5881. Thus we draw the conclusion that smoking is an indicator of heart disease. As PhysicalActivity is Yes(1), people are less likely to have heart disease. It seems that for people who have disease, the probability of that person being male is 0.59, and we might make a conclusion that male tend to have heart disease.

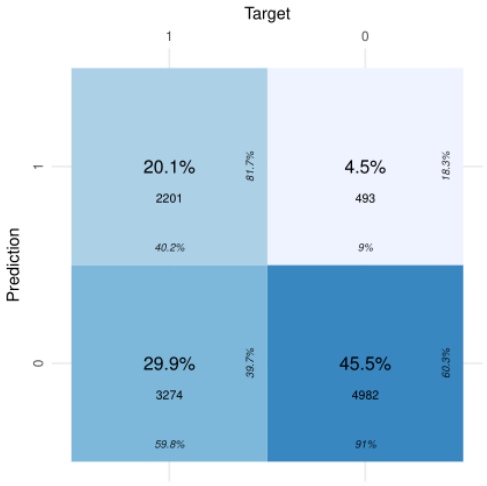

Figure 16: Naive Bayes Confusion Matrix

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 4982 3274
         1  493 2201

               Accuracy : 0.656
                 95% CI : (0.647, 0.6649)
    No Information Rate : 0.5
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.312

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9100
            Specificity : 0.4020
         Pos Pred Value : 0.6034
         Neg Pred Value : 0.8170
             Prevalence : 0.5000
         Detection Rate : 0.4550
   Detection Prevalence : 0.7540
```

Table 17: NB Confusion Matrix Statistics

We could calculate the accuracy by the confusion matrix which is similar to other models.

The accuracy of Naive Bayes Classification is 65.6%. And the reason accuracy is lower than other models might be two assumptions that each variable is independent and each variable has equal effects on outcome.

Part IV Final model

Overall, based on the results of the accuracy of the above 7 models, we get: SVM (75.55%) > Logistic Regression (75.35%) > Random Forest (75.25%) > Decision Tree (75.03%) > Gradient Boosting (74.67%) > KNN (74.33%) > Naive Bayes (65.6%). Then, we compared the AUC of the top four models again in order to be more discreet, and we get: Logistic Regression (0.8301) > SVM (0.8259) > Decision Tree (0.816) > Random Forest (0.8007). Therefore, based on the comparison of accuracy and AUC, we finally made a choice between SVM and Logistic

Regression. Based on the features of the two models, we believe that the Logistic Regression model has more analyzable information and better interpretation, so we took Logistic Regression as the final choice, and analyzed it in more detail.

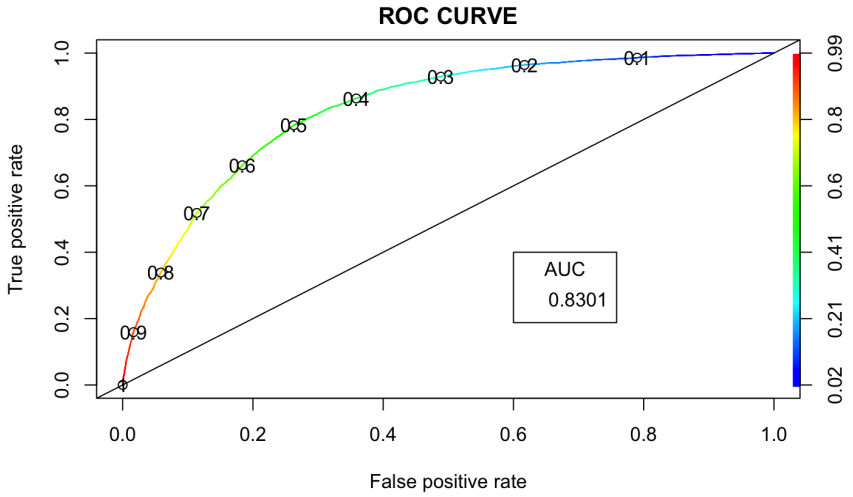

Figure 17: Logistic Regression ROC Curve

According to the ROC curve, the more the area under the curve, the better the model. AUC is 0.8301, so the more AUC is, the better the model performs.

```
Call:
glm(formula = HeartDisease ~ GenHealth + AgeCategory + DiffWalking +
    Stroke + PhysicalHealth + KidneyDisease + Smoking + PhysicalActivity +
    SkinCancer + Sex + BMI, family = "binomial", data = Train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2015  -0.8151  -0.0390   0.8249   2.8416

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)      -1.865257   0.093497 -19.950  < 2e-16 ***
GenHealth        -0.581678   0.015072 -38.594  < 2e-16 ***
AgeCategory       0.904869   0.015810  57.234  < 2e-16 ***
DiffWalking       0.335049   0.036141   9.271  < 2e-16 ***
Stroke            1.289966   0.055982  23.042  < 2e-16 ***
PhysicalHealth    0.004697   0.001668   2.815  0.00488 **
KidneyDisease     0.759939   0.057016  13.329  < 2e-16 ***
Smoking           0.404070   0.025949  15.572  < 2e-16 ***
PhysicalActivity -0.047993   0.030634  -1.567  0.11719
SkinCancer        0.280050   0.038756   7.226 4.97e-13 ***
Sex               0.690955   0.026317  26.255  < 2e-16 ***
BMI               0.011775   0.002120   5.554 2.79e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 50849  on 36679  degrees of freedom
Residual deviance: 36789  on 36668  degrees of freedom
AIC: 36813
```

Table 18: Logistic Regression Model

```
exp(coefficients(mylogit))
```

(Intercept)	GenHealth	AgeCategory	DiffWalking	Stroke
0.1548564	0.5589597	2.4716069	1.3980094	3.6326640
PhysicalHealth	**KidneyDisease**	**Smoking**	**PhysicalActivity**	**SkinCancer**
1.0047078	2.1381468	1.4979082	0.9531402	1.3231960
Sex	**BMI**			
1.9956204	1.0118443			

Table 19: Coefficients of LR Model After Exponential

After we fit the full model, we found that the p-value of the "PhysicalActivity" variable (0.11719) was much higher than the other variables, indicating that this evidence was not sufficient to indicate the presence of an effect on heart disease. To ensure that whether this variable makes sense for our model, we also performed a stepwise model selection and found that coefficients of the variables did not change, so we chose not to remove this variable and kept using the full model.

As for the coefficients, we firstly exponentiate the values of beta to obtain the odds ratio that associates variables of risk of heart diseases. The GenHealth group has 44% (0.56– 1 = -0.44) less odds of having heart disease than the non-GenHealth group which specifies a quantitative measurement to show how a healthy body condition could reduce the risk of having heart diseases. One extreme factor is Stroke. We could observe that people who had a stroke have 263% more odds of having heart diseases than people who do not suffer from a stroke. Another factor which is non-influential is PhysicalHealth, the PhysicalHealth has 0.5% more odds of having heart diseases. Therefore, by observing the exponential of coefficients, we could measure the probability of risk of heart disease for each variable respectively in a concrete way.

Part V Conclusion

After comparing the absolute value of Z-score, we found that “ageCategory”, “genhealth” and “sex” are the first three important factors. Based on the results we found, we give the following recommendations:

1. As people get older, they should pay more attention to this aspect of the disease. People should check their body regularly, and by including more whole grains, fruits, vegetables, oats, beans and nuts in their daily diet since they can increase their intake of dietary fiber.
2. Although the health of genetics cannot be changed, however, if people know their family history of heart disease, they can start taking care of their health from early on. They can always have the equipment and medications for heart disease in their home, just in case.
3. Based on the results of our analysis, we conclude that men are more likely to have the disease than women. Here we suggest that men should reduce smoking and alcohol abuse, reduce the behavior of intaking high cholesterol, increase exercise and maintain emotional stability.

The Cited Page

Gandhi, Rohith. (2018, May 5). Naive Bayes Classifier.

Retrieved April 30, 2022, from Naive Bayes Classifier. What is a classifier? | by Rohith Gandhi | Towards Data Science

Joby, Amal. (2021, July 19). What Is K-Nearest Neighbor? An ML Algorithm to Classify Data.

Retrieved April 30, 2022, from https://learn.g2.com/k-nearest-neighbor

Kumar, A. (2020, June 30). Introduction to the gradient boosting algorithm. Medium. Retrieved April 30, 2022, from https://medium.com/analytics-vidhya/introduction-to-the-gradient-boosting-algorithm-c25c653f826b

Kurama, V. (2021, April 9). *A complete guide to decision trees*. Paperspace Blog. Retrieved April 30, 2022, from https://blog.paperspace.com/decision-trees/

LaValley, Michael P. "Logistic regression." Circulation 117.18 (2008): 2395-2399.

Lloyd-Jones, Donald M., et al. "Lifetime risk of developing coronary heart disease." The Lancet 353.9147 (1999): 89-92.

Pytlak, Kamil. “Personal Key Indicators of Heart Disease.” Kaggle, 16 Feb. 2022, https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease.