Untitled

Lukas van der Watt

11/18/2021

Business Problem

A hospital wants to be able to understand what patients are at higher risk of developing heart disease. They have collected a set of records of information on patients with and without heart disease and would like to be able to predict and classify whether an individual is at high/low risk of developing heart disease. They would also like to know which of the data variables are more important to look at when trying to predict heart disease.

They have a patient with the following data profile and would like to know the chances of this individual developing heart disease. Female , Age = 44, Chest Pain Type = ATA, Cholesterol = 220 , FastingBS = 1 ,ExcerciseAngina = "N" , OldPeak = 4.2 , StSlope = "Flat"

```
#Calling packages required to run the various commands
library(ISLR)
library(caret)
```

```
## Loading required package: lattice

## Loading required package: ggplot2
```

```
library(e1071)
library(ggplot2)
library(cowplot)
#Reading in the data
Heart <- read.csv('heart.csv')
str(Heart) #Looking at the structure of the data to see what kind of variables are present
```

```
## 'data.frame':    918 obs. of  13 variables:
##  $ Serial.No.    : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Age           : int  40 49 37 48 54 39 45 54 37 48 ...
##  $ Sex           : chr  "M" "F" "M" "F" ...
##  $ ChestPainType : chr  "ATA" "NAP" "ATA" "ASY" ...
##  $ RestingBP     : int  140 160 130 138 150 120 130 110 140 120 ...
##  $ Cholesterol   : int  289 180 283 214 195 339 237 208 207 284 ...
##  $ FastingBS     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ RestingECG    : chr  "Normal" "Normal" "ST" "Normal" ...
##  $ MaxHR         : int  172 156 98 108 122 170 170 142 130 120 ...
##  $ ExerciseAngina: chr  "N" "N" "N" "Y" ...
##  $ Oldpeak       : num  0 1 0 1.5 0 0 0 0 1.5 0 ...
##  $ ST_Slope      : chr  "Up" "Flat" "Up" "Flat" ...
##  $ HeartDisease  : int  0 1 0 1 0 0 0 0 1 0 ...
```

```r
head(Heart) #Looking at the first part of the data.
```

```
##   Serial.No. Age Sex ChestPainType RestingBP Cholesterol FastingBS RestingECG
## 1          1  40   M           ATA       140         289         0     Normal
## 2          2  49   F           NAP       160         180         0     Normal
## 3          3  37   M           ATA       130         283         0         ST
## 4          4  48   F           ASY       138         214         0     Normal
## 5          5  54   M           NAP       150         195         0     Normal
## 6          6  39   M           NAP       120         339         0     Normal
##   MaxHR ExerciseAngina Oldpeak ST_Slope HeartDisease
## 1   172              N     0.0       Up            0
## 2   156              N     1.0     Flat            1
## 3    98              N     0.0       Up            0
## 4   108              Y     1.5     Flat            1
## 5   122              N     0.0       Up            0
## 6   170              N     0.0       Up            0
```

```r
#Converting the required variables to factors
Heart$Sex <-as.factor(Heart$Sex)
Heart$ChestPainType <-as.factor(Heart$ChestPainType)
Heart$RestingECG <-as.factor(Heart$RestingECG)
Heart$ExerciseAngina <-as.factor(Heart$ExerciseAngina)
Heart$ST_Slope<-as.factor(Heart$ST_Slope)
#Omitting any missing values from the data.
Heart<- na.omit(Heart)
str(Heart)
```

```
## 'data.frame':    918 obs. of  13 variables:
##  $ Serial.No.    : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Age           : int  40 49 37 48 54 39 45 54 37 48 ...
##  $ Sex           : Factor w/ 2 levels "F","M": 2 1 2 1 2 2 1 2 2 1 ...
##  $ ChestPainType : Factor w/ 4 levels "ASY","ATA","NAP",..: 2 3 2 1 3 3 2 2 1 2 ...
##  $ RestingBP     : int  140 160 130 138 150 120 130 110 140 120 ...
##  $ Cholesterol   : int  289 180 283 214 195 339 237 208 207 284 ...
##  $ FastingBS     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ RestingECG    : Factor w/ 3 levels "LVH","Normal",..: 2 2 3 2 2 2 2 2 2 2 ...
##  $ MaxHR         : int  172 156 98 108 122 170 170 142 130 120 ...
##  $ ExerciseAngina: Factor w/ 2 levels "N","Y": 1 1 1 2 1 1 1 1 2 1 ...
##  $ Oldpeak       : num  0 1 0 1.5 0 0 0 0 1.5 0 ...
##  $ ST_Slope      : Factor w/ 3 levels "Down","Flat",..: 3 2 3 2 3 3 3 3 2 3 ...
##  $ HeartDisease  : int  0 1 0 1 0 0 0 0 1 0 ...
```

```r
head(Heart)
```

```
##   Serial.No. Age Sex ChestPainType RestingBP Cholesterol FastingBS RestingECG
## 1          1  40   M           ATA       140         289         0     Normal
## 2          2  49   F           NAP       160         180         0     Normal
## 3          3  37   M           ATA       130         283         0         ST
## 4          4  48   F           ASY       138         214         0     Normal
## 5          5  54   M           NAP       150         195         0     Normal
## 6          6  39   M           NAP       120         339         0     Normal
##   MaxHR ExerciseAngina Oldpeak ST_Slope HeartDisease
```

```
## 1   172              N     0.0       Up           0
## 2   156              N     1.0     Flat           1
## 3    98              N     0.0       Up           0
## 4   108              Y     1.5     Flat           1
## 5   122              N     0.0       Up           0
## 6   170              N     0.0       Up           0
```

O

```r
# Creating a table to see if  variables related to sex are distributed throughout the data set. If it i
xtabs(~HeartDisease + Sex, data = Heart)
```

```
##             Sex
## HeartDisease   F   M
##            0 143 267
##            1  50 458
```

```r
xtabs(~ChestPainType + Sex, data = Heart)
```

```
##              Sex
## ChestPainType   F   M
##           ASY  70 426
##           ATA  60 113
##           NAP  53 150
##           TA   10  36
```

```r
xtabs(~RestingECG + Sex, data = Heart)
```

```
##           Sex
## RestingECG   F   M
##     LVH     47 141
##     Normal 118 434
##     ST      28 150
```

Looking at the tales above it is clear that the variable data is distributed throughout the data.

To be able to understand the relationship between the dependent variable and one or more independent variables in the data I am estimating probabilities using a logistic regression model. This type of analysis is used to help predict the likelihood of an event happening or a choice being made. In this case I am determining whether a patient does or does not have heart disease

```r
#Creating a logistical regression model called Log_mod

Log_mod <- glm(HeartDisease ~ . , data = Heart, family = "binomial") #Specifying that the binomial fami
summary(Log_mod)
```

```
##
## Call:
## glm(formula = HeartDisease ~ ., family = "binomial", data = Heart)
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max  
## -2.6566  -0.3739   0.1774   0.4482   2.5777  
## 
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)    
## (Intercept)      -1.1557913  1.4160841  -0.816 0.414392    
## Serial.No.        0.0002254  0.0004887   0.461 0.644644    
## Age               0.0145904  0.0138578   1.053 0.292403    
## SexM              1.4720611  0.2802828   5.252 1.50e-07 ***
## ChestPainTypeATA -1.8148593  0.3279399  -5.534 3.13e-08 ***
## ChestPainTypeNAP -1.6924335  0.2665782  -6.349 2.17e-10 ***
## ChestPainTypeTA  -1.4916447  0.4324012  -3.450 0.000561 ***
## RestingBP         0.0043751  0.0060191   0.727 0.467303    
## Cholesterol      -0.0041406  0.0010888  -3.803 0.000143 ***
## FastingBS         1.1384075  0.2749570   4.140 3.47e-05 ***
## RestingECGNormal -0.1279691  0.2916698  -0.439 0.660845    
## RestingECGST     -0.2116033  0.3710958  -0.570 0.568534    
## MaxHR            -0.0049068  0.0052031  -0.943 0.345647    
## ExerciseAnginaY   0.8982456  0.2446288   3.672 0.000241 ***
## Oldpeak           0.3779859  0.1186547   3.186 0.001445 ** 
## ST_SlopeFlat      1.4705611  0.4295853   3.423 0.000619 ***
## ST_SlopeUp       -0.9816831  0.4494492  -2.184 0.028948 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 1262.14  on 917  degrees of freedom
## Residual deviance:  593.97  on 901  degrees of freedom
## AIC: 627.97
## 
## Number of Fisher Scoring iterations: 6
```

Interpreting the Logistic model output: In knowing whether a variable is statistically significant we have to look at the P value as well as the effect size (Estimate) of the variable. A P-value smaller than 0.05 is likely to be a significant variable to the target variable. A small p-value together with a higher effect size indicates that the variable is important in the determination of the target variable which is the HeartDisease variable. For Example: From the above results we can see that Age p-value is at 0.292403 which is quite high and above 0.05 with an effect size of 0.0146 (smaller related to other estimates) indicating that this variable is not very useful. Sex is a good predictor because the p-value 1.50e-07 being far below 0.05. The effect size is also bigger when compared to other variables.

Based on the Logistical model output I have selected the following variables: Sex, ChestPainType, Cholesterol, FastingBS, ExerciseAngina, Oldpeak, ST_Slope, HeartDisease

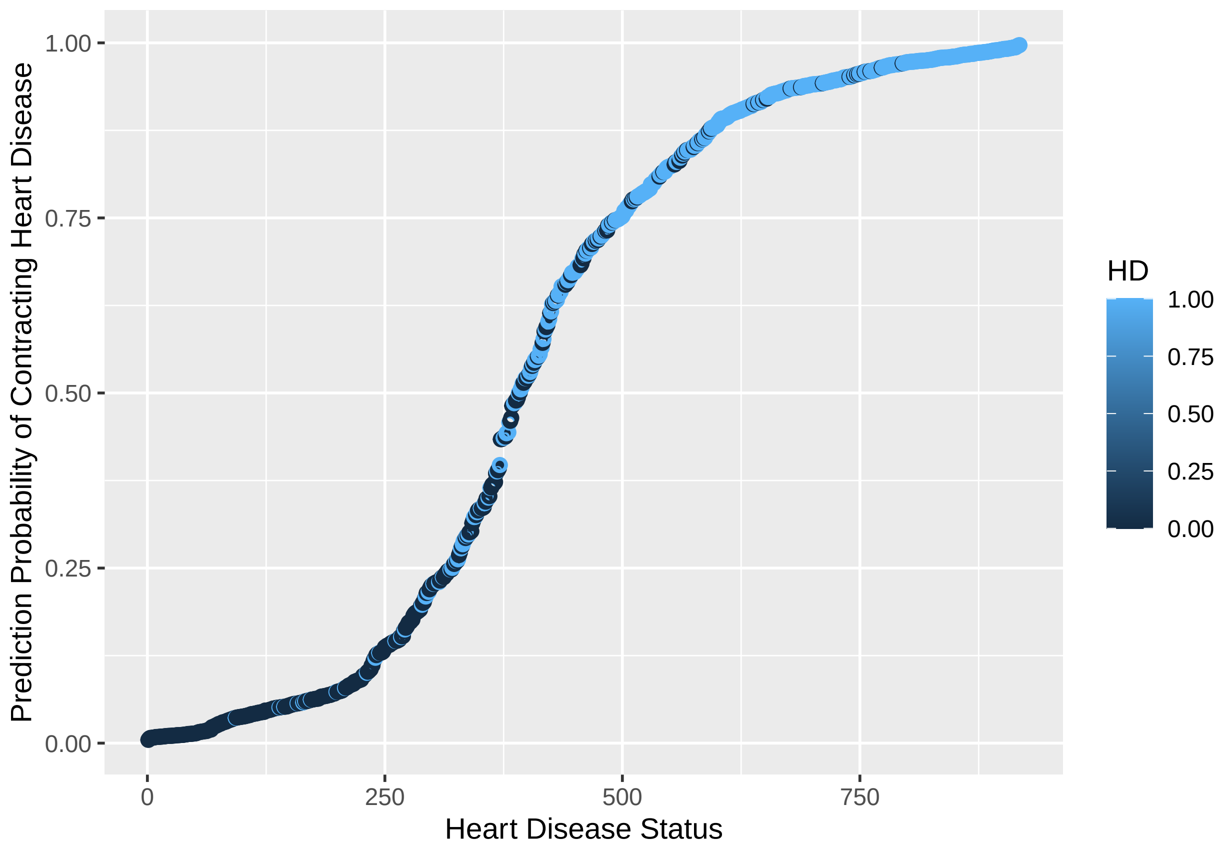
```
#Calculation of the McFAdden's Pseudo R^2
#The null
log_liklihood_null <- Log_mod$null.deviance/-2
log_liklihood_prop <- Log_mod$deviance/-2
#The calculation for the Pseudo R^2:
(log_liklihood_null-log_liklihood_prop)/log_liklihood_null
```

```
## [1] 0.5293913
```

This R squared is also known as the over-all effect size of the model. For the model above we get 0.5293913. This means the model explains 53% variability to the target variable. This accuracy of 53% is not very good but explains why it can be difficult in the real world to predict the development of heart disease. A way to improve this model might be to look at the genetics of the patient and family history pertaining to heart problems. Having these additional could help to improve the accuracy or over-all effect size.

```
#Plotting a probability graph for the Logistical Regression Model
P_data <- data.frame(Prob_HD = Log_mod$fitted.values,HD = Heart$HeartDisease)
#Sorting the data from low to high probability
P_data <- P_data[order(P_data$Prob_HD,decreasing = FALSE),]
P_data$rank <- 1:nrow(P_data)

ggplot(data = P_data, aes(x = rank, y = Prob_HD))+geom_point(aes(color=HD),alpha = 4,shape=1, stroke = 
```




```
ggsave("Heart Probability.pdf")
```

```
## Saving 6.5 x 4.5 in image
```

In the above graph I am plotting the prediction of each patient contracting heart disease against their actual heart disease status. The light blue indicates that most people who have heart disease have a high probability of contracting heart disease. Similarly we can also see the low probability end indicated with the dark blue. These are patient that does not have heart disease and have a low probability of getting heart disease. We can see that there are a few cases where a patient without heart disease(dark blue markers) have a high probability of contracting heart disease at some point. This is what we want to be able to identify and predict.

```r
#Creating the Naive Bays Model
#Partition the data into training(60) validation(40)
selected.var <- Heart[,c(3,4,6,7,10,11,12,13)] #Selecting variables to be partitioned

set.seed(123) #randomize
train.in <- createDataPartition(selected.var$HeartDisease, p = 0.7, list = FALSE) #creating a training
Heart.train <- selected.var[train.in,] #Training set
Heart.valid <- selected.var[-train.in,] #Validation set
str(Heart.train) #structure of the training set
```

```
## 'data.frame':    643 obs. of  8 variables:
##  $ Sex           : Factor w/ 2 levels "F","M": 1 2 2 2 1 1 2 2 1 2 ...
##  $ ChestPainType : Factor w/ 4 levels "ASY","ATA","NAP",..: 3 3 3 2 2 3 2 1 2 1 ...
##  $ Cholesterol   : int  180 195 339 208 284 211 204 234 273 248 ...
##  $ FastingBS     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ ExerciseAngina: Factor w/ 2 levels "N","Y": 1 1 1 1 1 1 1 2 1 1 ...
##  $ Oldpeak       : num  1 0 0 0 0 0 0 1 1.5 1 ...
##  $ ST_Slope      : Factor w/ 3 levels "Down","Flat",..: 2 3 3 3 3 3 3 2 2 2 ...
##  $ HeartDisease  : int  1 0 0 0 0 0 0 1 0 1 ...
```

```r
#Method 1: This shows the pivot table with row variables (Online and CreditCard) and the column variabl
attach(Heart.train) # Attaching the training set to the following statements
ftable(HeartDisease, Sex, ChestPainType)#Creating a pivot table with ChestPainType as a column variable
```

```
##                  ChestPainType ASY ATA NAP  TA
## HeartDisease Sex
## 0            F                  25  45  29   5
##              M                  41  66  68  11
## 1            F                  30   2   5   1
##              M                 245  13  41  16
```

```r
ftable(HeartDisease,FastingBS,ExerciseAngina)#Creating a pivot table with ExerciseAngina as a column va
```

```
##                        ExerciseAngina   N   Y
## HeartDisease FastingBS
## 0            0                        217  36
##              1                         30   7
## 1            0                         85 152
##              1                         56  60
```

The above pivot tables show the conditional probabilities as they relate to Heart Diease.

```r
#The Following table is a probability representation of the pivot tables previously formed
prop.table(ftable(HeartDisease, Sex, ChestPainType))
```

```
##                  ChestPainType        ASY        ATA        NAP         TA
## HeartDisease Sex
## 0            F                 0.03888025 0.06998445 0.04510109 0.00777605
##              M                 0.06376361 0.10264386 0.10575428 0.01710731
## 1            F                 0.04665630 0.00311042 0.00777605 0.00155521
##              M                 0.38102644 0.02021773 0.06376361 0.02488336
```

```r
prop.table(ftable(HeartDisease,FastingBS,ExerciseAngina))
```

```
##                          ExerciseAngina          N          Y
## HeartDisease FastingBS
## 0            0                          0.33748056 0.05598756
##              1                          0.04665630 0.01088647
## 1            0                          0.13219285 0.23639191
##              1                          0.08709176 0.09331260
```

```r
detach(Heart.train)
```

In the above results we see that the

Creating The Naive Bayes Model because I am predicting the probability of different classes. I am making the assumption of independence. Meaning it is making the prediction that the effect of the value of a predictor (x) on a given class (c) is independent of the values of other predictors. In this dataset it (real world data) is very rare that there would be predictors which are completely independent. Nevertheless it is a technique that performs well with categorical variables where a sick/not-sick outcome is expected and despite the literal naive assumption of idependance the technique does very well as it ouperforms more sophisticated methods.

```r
Heartdata.nb <- naiveBayes(HeartDisease~., data = Heart.train)# Creating the Naive Bayes Model on the t
Heartdata.nb # Showcasing the model
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##         0         1
## 0.4510109 0.5489891
##
## Conditional probabilities:
##    Sex
## Y           F         M
##   0 0.3586207 0.6413793
##   1 0.1076487 0.8923513
##
##    ChestPainType
## Y          ASY        ATA        NAP         TA
##   0 0.22758621 0.38275862 0.33448276 0.05517241
##   1 0.77903683 0.04249292 0.13031161 0.04815864
##
##    Cholesterol
## Y       [,1]      [,2]
##   0 223.2552  75.90595
##   1 178.5637 125.01401
##
##    FastingBS
```

```
## Y         [,1]      [,2]
##   0 0.1275862 0.3342052
##   1 0.3286119 0.4703753
##
##    ExerciseAngina
## Y           N         Y
##   0 0.8517241 0.1482759
##   1 0.3994334 0.6005666
##
##    Oldpeak
## Y         [,1]      [,2]
##   0 0.4134483 0.6698075
##   1 1.3186969 1.1723818
##
##    ST_Slope
## Y          Down       Flat         Up
##   0 0.03448276 0.19310345 0.77241379
##   1 0.08781870 0.77337110 0.13881020
```

```
pre <- predict(Heartdata.nb,Heart.valid)#Class membership
pre.prob <- predict(Heartdata.nb,newdata = Heart.valid,type = "raw")#Probabilities
```

```
Heart.valid$HeartDisease<-as.factor(Heart.valid$HeartDisease)
cfm <- confusionMatrix(pre,Heart.valid$HeartDisease)
cfm
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 105  19
##          1  15 136
##
##                Accuracy : 0.8764
##                  95% CI : (0.8315, 0.9128)
##     No Information Rate : 0.5636
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.7496
##
##  Mcnemar's Test P-Value : 0.6069
##
##             Sensitivity : 0.8750
##             Specificity : 0.8774
##          Pos Pred Value : 0.8468
##          Neg Pred Value : 0.9007
##              Prevalence : 0.4364
##          Detection Rate : 0.3818
##    Detection Prevalence : 0.4509
##       Balanced Accuracy : 0.8762
##
##        'Positive' Class : 0
##
```

Bear in mind that a 0 indicates the patient is normal where a 1 indicates the presence of Heart Disease. False Negatives : A Total of 19 cases of the 275 observations in the validation set False Positives : A total of 15 cases of the 275 observations in the validation set Therefore a total of 34 missclassification errors

Sensitivity (TP + FN) is the proportion of positives correctly identified and in this Naive Bayes model we see that 87.5% of the positives are correctly identified.

Specificity is the True Negative Rate and in this model we see that 87.7% of the negatives are correctly identified.