FML Assignment-04

Chandima Attanayake

2024-10-15

```r
# Load necessary libraries
library(caret)
```

```
## Loading required package: ggplot2

## Loading required package: lattice
```

```r
library(e1071)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
# Load the heart disease dataset with the corrected file path
heart_disease <- read.csv("/Users/chandimaattanayake/Downloads/Heart_disease.csv")

# Create 'Target' and 'BP_New' variables as required
heart_disease$Target <- ifelse(heart_disease$MAX_HeartRate > 170, "Yes", "No")
heart_disease$BP_New <- ifelse(heart_disease$Blood_Pressure > 120, "Yes", "No")

# Table to summarize the 'Target' variable
Target_table <- table(heart_disease$Target)
Target_table
```

```
##
##  No Yes
## 245  58
```

```r
# Calculate proportions
prop.table(Target_table)
```

```
##
##        No       Yes
## 0.8085809 0.1914191
```

```r
# Based on the higher proportion of Target = "No", it would predict "No" in this case..
```

Q 02. Analysis of the First 30 Records

- a. Compute Bayes Conditional Probabilities*

```r
# Assuming 'Heart_disease30' is the subset of the first 30 records
Heart_disease30 <- heart_disease[1:30, c("Target", "BP_New", "chest_pain_type")]

# Compute conditional probabilities directly by filtering for each combination
P1 <- sum(Heart_disease30$Target == "Yes" & Heart_disease30$BP_New == "No" & Heart_disease30$chest_pain_
      sum(Heart_disease30$BP_New == "No" & Heart_disease30$chest_pain_type == 0)

P2 <- sum(Heart_disease30$Target == "Yes" & Heart_disease30$BP_New == "No" & Heart_disease30$chest_pain_
      sum(Heart_disease30$BP_New == "No" & Heart_disease30$chest_pain_type == 1)

P3 <- sum(Heart_disease30$Target == "Yes" & Heart_disease30$BP_New == "Yes" & Heart_disease30$chest_pain
      sum(Heart_disease30$BP_New == "Yes" & Heart_disease30$chest_pain_type == 0)

P4 <- sum(Heart_disease30$Target == "Yes" & Heart_disease30$BP_New == "Yes" & Heart_disease30$chest_pain
      sum(Heart_disease30$BP_New == "Yes" & Heart_disease30$chest_pain_type == 1)

# Display the calculated probabilities
P1; P2; P3; P4
```

```
## [1] 0

## [1] 0.6

## [1] 0.3

## [1] 0.3846154
```

- b. Classification of Accidents*

```r
# Initialize a vector to store probabilities for each record
Probability_Target <- rep(0, 30)

# Manually classify records based on computed probabilities
for (i in 1:30) {
  if (Heart_disease30$BP_New[i] == "No" & Heart_disease30$chest_pain_type[i] == 0) {
    Probability_Target[i] <- P1
  } else if (Heart_disease30$BP_New[i] == "No" & Heart_disease30$chest_pain_type[i] == 1) {
    Probability_Target[i] <- P2
  } else if (Heart_disease30$BP_New[i] == "Yes" & Heart_disease30$chest_pain_type[i] == 0) {
    Probability_Target[i] <- P3
  } else if (Heart_disease30$BP_New[i] == "Yes" & Heart_disease30$chest_pain_type[i] == 1) {
```

```r
    Probability_Target[i] <- P4
  }
}

# Add Probability_Target and Pred_Probability to the dataset
Heart_disease30$Probability_Target <- Probability_Target
Heart_disease30$Pred_Probability <- ifelse(Heart_disease30$Probability_Target > 0.5, "Yes", "No")

# Display the updated dataset
print(Heart_disease30)
```

```
##    Target BP_New chest_pain_type Probability_Target Pred_Probability
## 1      No    Yes               0          0.3000000               No
## 2     Yes    Yes               1          0.3846154               No
## 3     Yes    Yes               1          0.3846154               No
## 4     Yes     No               1          0.6000000              Yes
## 5      No     No               0          0.0000000               No
## 6      No    Yes               0          0.3000000               No
## 7      No    Yes               1          0.3846154               No
## 8     Yes     No               1          0.6000000              Yes
## 9      No    Yes               1          0.3846154               No
## 10    Yes    Yes               1          0.3846154               No
## 11     No    Yes               0          0.3000000               No
## 12     No    Yes               1          0.3846154               No
## 13    Yes    Yes               1          0.3846154               No
## 14     No     No               0          0.0000000               No
## 15     No    Yes               0          0.3000000               No
## 16     No     No               1          0.6000000              Yes
## 17    Yes     No               1          0.6000000              Yes
## 18     No    Yes               0          0.3000000               No
## 19    Yes    Yes               0          0.3000000               No
## 20     No    Yes               0          0.3000000               No
## 21     No    Yes               0          0.3000000               No
## 22    Yes    Yes               1          0.3846154               No
## 23    Yes    Yes               0          0.3000000               No
## 24     No    Yes               1          0.3846154               No
## 25    Yes    Yes               0          0.3000000               No
## 26     No    Yes               1          0.3846154               No
## 27     No    Yes               1          0.3846154               No
## 28     No     No               1          0.6000000              Yes
## 29     No    Yes               1          0.3846154               No
## 30     No    Yes               1          0.3846154               No
```

- c. Manual Calculation of Naive Bayes Probability*

```r
# Manually calculate the naive Bayes conditional probability for BP_New = Yes and chest_pain_type = 1
manual_naive_bayes_prob <- P4  # As per the calculation above
manual_naive_bayes_prob
```

```
## [1] 0.3846154
```

3. Full Dataset Analysis

```r
# Set seed for reproducibility
set.seed(123)

# Split the data into 60% training and 40% validation
trainIndex <- createDataPartition(heart_disease$Target, p = 0.6, list = FALSE)
train_data <- heart_disease[trainIndex, ]
valid_data <- heart_disease[-trainIndex, ]

# Train Naive Bayes model using 'chest_pain_type' and 'BP_New' predictors
nb_model <- naiveBayes(Target ~ chest_pain_type + BP_New, data = train_data)

# Predict on the validation dataset
valid_pred <- predict(nb_model, valid_data)

# Ensure both actual and predicted values are factors with the same levels
valid_data$Target <- factor(valid_data$Target, levels = c("No", "Yes"))  # Adjust levels according to t
valid_pred <- factor(valid_pred, levels = c("No", "Yes"))  # Adjust levels according to the dataset

# Generate confusion matrix
conf_matrix <- confusionMatrix(valid_pred, valid_data$Target)

# Display confusion matrix
print(conf_matrix)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction No Yes
##        No  98  23
##        Yes  0   0
##
##                Accuracy : 0.8099
##                  95% CI : (0.7286, 0.8755)
##     No Information Rate : 0.8099
##     P-Value [Acc > NIR] : 0.5555
##
##                   Kappa : 0
##
##  Mcnemar's Test P-Value : 4.49e-06
##
##             Sensitivity : 1.0000
##             Specificity : 0.0000
##          Pos Pred Value : 0.8099
##          Neg Pred Value :    NaN
##              Prevalence : 0.8099
##          Detection Rate : 0.8099
##    Detection Prevalence : 1.0000
##       Balanced Accuracy : 0.5000
##
##        'Positive' Class : No
##
```