ML-Assigment-3 Naive bayes

Harish Kunaparaju

2022-10-13

```r
library(pivottabler)
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```r
library(ISLR)
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
library(e1071)
```

```r
#Importing data set for current enviroment.
library(readr)
data <- read.csv("UniversalBank.csv")
str(data)
```

```
## 'data.frame':    5000 obs. of  14 variables:
##  $ ID                : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Age               : int  25 45 39 35 35 37 53 50 35 34 ...
##  $ Experience        : int  1 19 15 9 8 13 27 24 10 9 ...
##  $ Income            : int  49 34 11 100 45 29 72 22 81 180 ...
##  $ ZIP.Code          : int  91107 90089 94720 94112 91330 92121 91711 93943 90089 93023 ...
##  $ Family            : int  4 3 1 1 4 4 2 1 3 1 ...
##  $ CCAvg             : num  1.6 1.5 1 2.7 1 0.4 1.5 0.3 0.6 8.9 ...
##  $ Education         : int  1 1 1 2 2 2 2 3 2 3 ...
##  $ Mortgage          : int  0 0 0 0 0 155 0 0 104 0 ...
##  $ Personal.Loan     : int  0 0 0 0 0 0 0 0 0 1 ...
##  $ Securities.Account: int  1 1 0 0 0 0 0 0 0 0 ...
##  $ CD.Account        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Online            : int  0 0 0 0 0 1 1 0 1 0 ...
##  $ CreditCard        : int  0 0 0 0 1 0 0 1 0 0 ...
```

```r
# converting Online,Credit Card,Personal loan,to factors from int.
data$Online<-as.factor(data$Online)
is.factor(data$Online)
```

```
## [1] TRUE
```

```
data$CreditCard<-as.factor(data$CreditCard)
is.factor(data$CreditCard)
```

```
## [1] TRUE
```

```
data$Personal.Loan<-as.factor(data$Personal.Loan)
is.factor(data$Personal.Loan)
```

```
## [1] TRUE
```

```
str(data)
```

```
## 'data.frame':    5000 obs. of  14 variables:
##  $ ID                : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Age               : int  25 45 39 35 35 37 53 50 35 34 ...
##  $ Experience        : int  1 19 15 9 8 13 27 24 10 9 ...
##  $ Income            : int  49 34 11 100 45 29 72 22 81 180 ...
##  $ ZIP.Code          : int  91107 90089 94720 94112 91330 92121 91711 93943 90089 93023 ...
##  $ Family            : int  4 3 1 1 4 4 2 1 3 1 ...
##  $ CCAvg             : num  1.6 1.5 1 2.7 1 0.4 1.5 0.3 0.6 8.9 ...
##  $ Education         : int  1 1 1 2 2 2 2 3 2 3 ...
##  $ Mortgage          : int  0 0 0 0 0 155 0 0 104 0 ...
##  $ Personal.Loan     : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 2 ...
##  $ Securities.Account: int  1 1 0 0 0 0 0 0 0 0 ...
##  $ CD.Account        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Online            : Factor w/ 2 levels "0","1": 1 1 1 1 1 2 2 1 2 1 ...
##  $ CreditCard        : Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 2 1 1 ...
```

```
#partition the data into Training (60%) and validate (40%).
set.seed(123)
Train_index<- createDataPartition(data$Personal.Loan, p=0.60, list = FALSE)
traning <- data[Train_index,]
validation<-data[-Train_index,]
```

```
#Data Normalization.
Mydata <- preProcess(data[,-c(10,13,14)], method = c("center", "scale"))
Feature_tdata <- predict(Mydata, traning)
Feature_vdata <- predict(Mydata, validation)
```

```
#A. Creating Pivot Table with Online as column variable and CC, Personal.Loan as row variables by using
pivot_data<- ftable(Feature_tdata$Personal.Loan, Feature_tdata$Online, Feature_tdata$CreditCard, dnn=c(
pivot_data
```

```
##                           Online    0    1
## Personal.loan CreditCard                  
## 0             0                   791  310
##               1                  1144  467
## 1             0                    79   33
##               1                   125   51
```

```
#B.Probability of Loan Acceptance (Loan=1) conditional on CC=1 and Online=1.
prob_data<-pivot_data[4,2]/(pivot_data[2,2]+pivot_data[4,2])
prob_data
```

```
## [1] 0.0984556
```

```r
# Creating two separate Pivot tables for the training data.
#C1.probability for personal loan and Online.
pivot_data<- ftable(Feature_tdata$Personal.Loan,Feature_tdata$Online,dnn=c('Personal.loan','Online'))
pivot_data
```

```
##               Online    0    1
## Personal.loan
## 0                    1101 1611
## 1                     112  176
```

```r
#C2.probability for personal loan and Credit Card.
pivot_data2<- ftable(Feature_tdata$Personal.Loan,Feature_tdata$CreditCard, dnn=c('Personal.loan','Credi
pivot_data2
```

```
##               CreditCard    0    1
## Personal.loan
## 0                        1935  777
## 1                         204   84
```

```r
#D.(i).P(CC=1 | Loan= 1)(The proporation of credit card holders among the loan acceptors)
data1<- pivot_data2[2,2]/(pivot_data2[2,2]+pivot_data2[2,1])
data1
```

```
## [1] 0.2916667
```

```r
#D.(ii).P(Online=1 | Loan=1)
data2 <- pivot_data[2,2]/(pivot_data[2,2]+pivot_data[2,1])
data2
```

```
## [1] 0.6111111
```

```r
#D.(iii).P(Loan=1)(The proporation of loan acceptors)
data3 <- ftable(Feature_tdata[,10])
data3
```

```
##     0    1
##
##  2712  288
```

```r
data3 <- data3[1,2]/(data3[1,2]+data3[1,1])
data3
```

```
## [1] 0.096
```

```r
#D.(iv).P(CC=1 | Loan=0)
data4 <- pivot_data2[1,2]/(pivot_data2[1,2]+pivot_data2[1,1])
data4
```

```
## [1] 0.2865044
```

```r
#D.(v).P(Online=1 | Loan=0)
data5 <- pivot_data[1,2]/(pivot_data[1,2]+pivot_data[1,1])
data5
```

```
## [1] 0.5940265
```

```r
#D.(vi).P(Loan=0)
data6 <- ftable(Feature_tdata[,10])
data6
```

```
##     0    1
```

```
##
##  2712  288
```

```r
data6 <- data6[1,1]/(data6[1,1]+data6[1,2])
data6
```

```
## [1] 0.904
```

```r
#E.Computing Naive Bayes using conditional probabilities from D [P(Loan=1/Creditcard=1,Online=1)].
nb <- (data1*data2*data3)/(data1*data2*data3+data4*data5*data6)
nb
```

```
## [1] 0.1000861
```

#F.Compare E values with one obtained from the pivot table in B,Whih is more Accurate estimate.

The probability derived from Bayes probability i.e., B. is 0.0984556 and the probability derived from Naive's Bayes i.e., is 0.1000. The comparison between Bayes and Naive bayes shows that Naive Bayes has a higher probability.

```r
#G.Using Naive Bayes directly applied to the data.
nb_model <-naiveBayes(Personal.Loan~Online+CreditCard, data=Feature_tdata)
nb_model
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##     0     1
## 0.904 0.096
##
## Conditional probabilities:
##    Online
## Y           0         1
##   0 0.4059735 0.5940265
##   1 0.3888889 0.6111111
##
##    CreditCard
## Y           0         1
##   0 0.7134956 0.2865044
##   1 0.7083333 0.2916667
```

```r
#From the below table we can observe that for P(Loan=1| CC=1, Online=1), following values are to be con
```