Naive Bayes method

Arbuda Sivani

3/06/2022

```
#Importing Data and installing required packages:
UniversalBank <- read.csv("~/ML/Assignment/Assignment_2/UniversalBank.csv")
summary(UniversalBank)
```

```
##        ID            Age          Experience       Income          ZIP.Code
##  Min.   :   1   Min.   :23.00   Min.   :-3.0   Min.   :  8.00   Min.   : 9307
##  1st Qu.:1251   1st Qu.:35.00   1st Qu.:10.0   1st Qu.: 39.00   1st Qu.:91911
##  Median :2500   Median :45.00   Median :20.0   Median : 64.00   Median :93437
##  Mean   :2500   Mean   :45.34   Mean   :20.1   Mean   : 73.77   Mean   :93153
##  3rd Qu.:3750   3rd Qu.:55.00   3rd Qu.:30.0   3rd Qu.: 98.00   3rd Qu.:94608
##  Max.   :5000   Max.   :67.00   Max.   :43.0   Max.   :224.00   Max.   :96651
##      Family          CCAvg          Education        Mortgage
##  Min.   :1.000   Min.   : 0.000   Min.   :1.000   Min.   :  0.0
##  1st Qu.:1.000   1st Qu.: 0.700   1st Qu.:1.000   1st Qu.:  0.0
##  Median :2.000   Median : 1.500   Median :2.000   Median :  0.0
##  Mean   :2.396   Mean   : 1.938   Mean   :1.881   Mean   : 56.5
##  3rd Qu.:3.000   3rd Qu.: 2.500   3rd Qu.:3.000   3rd Qu.:101.0
##  Max.   :4.000   Max.   :10.000   Max.   :3.000   Max.   :635.0
##  Personal.Loan   Securities.Account   CD.Account         Online
##  Min.   :0.000   Min.   :0.0000     Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.000   1st Qu.:0.0000     1st Qu.:0.0000   1st Qu.:0.0000
##  Median :0.000   Median :0.0000     Median :0.0000   Median :1.0000
##  Mean   :0.096   Mean   :0.1044     Mean   :0.0604   Mean   :0.5968
##  3rd Qu.:0.000   3rd Qu.:0.0000     3rd Qu.:0.0000   3rd Qu.:1.0000
##  Max.   :1.000   Max.   :1.0000     Max.   :1.0000   Max.   :1.0000
##    CreditCard
##  Min.   :0.000
##  1st Qu.:0.000
##  Median :0.000
##  Mean   :0.294
##  3rd Qu.:1.000
##  Max.   :1.000
```

```
library(caret)
```

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

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

```
library(class)
library(ISLR)
library(e1071)

UniversalBank$Personal.Loan<-factor(UniversalBank$Personal.Loan)
UniversalBank$CreditCard <- factor(UniversalBank$CreditCard)
UniversalBank$Online <- factor(UniversalBank$Online)
summary(UniversalBank)
```

```
##        ID              Age            Experience       Income          ZIP.Code
##  Min.   :   1   Min.   :23.00   Min.   :-3.0   Min.   :  8.00   Min.   : 9307
##  1st Qu.:1251   1st Qu.:35.00   1st Qu.:10.0   1st Qu.: 39.00   1st Qu.:91911
##  Median :2500   Median :45.00   Median :20.0   Median : 64.00   Median :93437
##  Mean   :2500   Mean   :45.34   Mean   :20.1   Mean   : 73.77   Mean   :93153
##  3rd Qu.:3750   3rd Qu.:55.00   3rd Qu.:30.0   3rd Qu.: 98.00   3rd Qu.:94608
##  Max.   :5000   Max.   :67.00   Max.   :43.0   Max.   :224.00   Max.   :96651
##      Family          CCAvg          Education        Mortgage     Personal.Loan
##  Min.   :1.000   Min.   : 0.000   Min.   :1.000   Min.   :  0.0   0:4520
##  1st Qu.:1.000   1st Qu.: 0.700   1st Qu.:1.000   1st Qu.:  0.0   1: 480
##  Median :2.000   Median : 1.500   Median :2.000   Median :  0.0
##  Mean   :2.396   Mean   : 1.938   Mean   :1.881   Mean   : 56.5
##  3rd Qu.:3.000   3rd Qu.: 2.500   3rd Qu.:3.000   3rd Qu.:101.0
##  Max.   :4.000   Max.   :10.000   Max.   :3.000   Max.   :635.0
##  Securities.Account   CD.Account     Online   CreditCard
##  Min.   :0.0000     Min.   :0.0000   0:2016   0:3530
##  1st Qu.:0.0000     1st Qu.:0.0000   1:2984   1:1470
##  Median :0.0000     Median :0.0000
##  Mean   :0.1044     Mean   :0.0604
##  3rd Qu.:0.0000     3rd Qu.:0.0000
##  Max.   :1.0000     Max.   :1.0000
```

```
#Question 1: Create a pivot table for the training data with Online as a column variable,
#CC as a row variable, and Loan as a secondary row variable.

set.seed(64060)
Train_Index <- createDataPartition(UniversalBank$Personal.Loan, p=0.6,list = FALSE)
Train.df <- UniversalBank[Train_Index,]
Validation.df <- UniversalBank[-Train_Index,]

mytable <- xtabs(~ CreditCard+Online+Personal.Loan, data = Train.df)
ftable(mytable)
```

```
##                    Personal.Loan    0    1
## CreditCard Online
## 0          0                     772   75
##            1                    1152  120
## 1          0                     309   34
##            1                     479   59
```

```
#Question 2: Consider the task of classifying a customer who owns a bank credit card and is actively
#using online banking services. Looking at the pivot table, what is the probability that this customer
#will accept the loan offer?
```

```
#[This is the probability of loan acceptance (Loan = 1) conditional on having a bank credit card
#(CC = 1) and being an active user of online banking services (Online = 1)].

Probability = (59/(59+479))

print(Probability)
```

```
## [1] 0.1096654
```

```
#Question 3: Create two separate pivot tables for the training data. One will have Loan (rows)
#as a function of Online (columns) and the other will have Loan (rows) as a function of CC.

table(Online=Train.df$Online, Personal.Loan=Train.df$Personal.Loan)
```

```
##       Personal.Loan
## Online    0    1
##      0 1081  109
##      1 1631  179
```

```
table(CreditCard=Train.df$CreditCard, Personal.Loan=Train.df$Personal.Loan)
```

```
##           Personal.Loan
## CreditCard    0    1
##          0 1924  195
##          1  788   93
```

```
#Question 4: Compute [P(A | B) means "the probability ofA given B"]:
#i. P(CC = 1 | Loan = 1)
#(the proportion of credit card holders among the loan acceptors)
#ii. P(Online = 1 | Loan = 1)
#iii. P(Loan = 1) (the proportion of loan acceptors)
#iv. P(CC = 1 | Loan = 0)
#v. P(Online = 1 | Loan = 0)
#vi. P(Loan = 0)

#i.P(CC=1 | Loan=1)
Prob_1 <- (93/(93+195))
print(Prob_1)
```

```
## [1] 0.3229167
```

```
#ii.P(Online=1 | Loan=1)
Prob_2 <- (179/(179+109))
print(Prob_2)
```

```
## [1] 0.6215278
```

```
#iii.P(Loan)
table(Personal.Loan = Train.df$Personal.Loan)
```

```
## Personal.Loan
##    0    1
## 2712  288
```

```
Prob_3 <- (288/(288+2712))
print(Prob_3)
```

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

```
#iv.P(CC=1 | Loan =0)
Prob_4 <- (788/(1924+788))
print(Prob_4)
```

```
## [1] 0.2905605
```

```
#v.P(Online=1 | Loan=0)
Prob_5 <- (1631/(1631+1081))
print(Prob_5)
```

```
## [1] 0.6014012
```

```
#vi.P(Loan=0)
Prob_6 <- (2712/(2712+288))
print(Prob_6)
```

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

```
#Question 5: Use the quantities computed above to compute the naive Bayes probability
#P(Loan = 1 | CC = 1, Online = 1).

Prob_7 <- (Prob_1*Prob_2*Prob_3)/((Prob_1*Prob_2*Prob_3)+(Prob_4*Prob_5*Prob_6))
print(Prob_7)
```

```
## [1] 0.1087106
```

```
#Question 6: Compare this value with the one obtained from the pivot table in (B).
#Which is a more accurate estimate?

#The exact method would be needing the exact same independent variable classifications for prediction
#and Naive Bayes does not require.

#The values derived from Task2(exact method) and Task5(Naive Bayes method) are 0.1096654 and 0.1087106
#respectively.If we observe there is a minute difference between the values from both the methods.
#The value derived from exact method is
#more accurate because we have taken the values directly from the pivot table
```

```
#Question 7: Which of the entries in this table are needed for computing
#P(Loan = 1 | CC = 1, Online = 1)?
#Run naive Bayes on the data. Examine the model output on training data, and find the entry
```

```r
#that corresponds to P(Loan = 1 | CC = 1, Online = 1).
#Compare this to the number you obtained in (E).

nb.model<-naiveBayes (Personal.Loan~ Online +CreditCard, data=Train.df)
To_Predict=data.frame(Online= '1', CreditCard= '1')
predict(nb.model,To_Predict, type = 'raw')
```

```
##              0         1
## [1,] 0.8912894 0.1087106
```

```r
#After the comparison I observed that the outputs of Naive Bayes method and the previous method
#(Question 5 and Question 7) is exactly the same i.e.,0.1087106
```