Assignment 3_ML

Lukas van der Watt

10/16/2021

```r
#Calling packages required to run the various commands
library(e1071)
library(caret)
```

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

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

```r
UB.data <- read.csv("UniversalBank.csv") #reading the .csv file for universal bank data
str(UB.data) #Looking at the structure of the data to see what kind of variables are present.
```

```
## '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 to factors
UB.data$Personal.Loan <- as.factor(UB.data$Personal.Loan) #Converting the Personal Loan variable to as.
UB.data$Online <- as.factor(UB.data$Online)#Converting the Online variable to as.factor
UB.data$CreditCard <- as.factor(UB.data$CreditCard)#Converting the CreditCard variable to as.factor
str(UB.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) validation(40)
selected.var <- c(10,13,14) #Selecting variables to be partitioned

set.seed(123) #randomize
train.in <- createDataPartition(UB.data$Online,p = 0.6, list = FALSE) #creating a training index with t
ub.data.train <- UB.data[train.in,selected.var] #Training set
ub.data.valid <- UB.data[-train.in,selected.var] #Validation set
str(ub.data.train) #structure of the training set
```

```
## 'data.frame':    3001 obs. of  3 variables:
##  $ Personal.Loan: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Online       : Factor w/ 2 levels "0","1": 1 1 1 1 2 2 1 1 2 1 ...
##  $ CreditCard   : Factor w/ 2 levels "0","1": 1 1 1 2 1 1 2 1 1 1 ...
```

PART A : Creating a Pivot Table

```
#Method 1: This shows the pivot table with row variables (Online and CreditCard) and the column variabl
attach(ub.data.train) # Attaching the training set to the following statements
ftable(Personal.Loan, CreditCard, Online) #Creating a pivot table with personal loan as a column variab
```

```
##                            Online    0    1
## Personal.Loan CreditCard
## 0             0                    786 1147
##               1                    309  478
## 1             0                     77  119
##               1                     38   47
```

```
detach(ub.data.train)
```

Part B Ans. P(Personal.Loan=1 | CreditCard=1 , Online=1) = 47/(47+478) = 0.08952381

```
#Method 2: This shows the pivot table with row variables (Personal Loan and CreditCard) and the column
prop.table(ftable(ub.data.train, row.vars = c(1,3), column.vars = 2 ),margin = 1)
```

```
##                            Online         0         1
## Personal.Loan CreditCard
## 0             0                   0.4066218 0.5933782
##               1                   0.3926302 0.6073698
## 1             0                   0.3928571 0.6071429
##               1                   0.4470588 0.5529412
```

PART C

```r
#Creating two seperate pivot tables
attach(ub.data.train)
prop.table(table(Personal.Loan, Online), margin = 1)
```

```
##              Online
## Personal.Loan         0         1
##             0 0.4025735 0.5974265
##             1 0.4092527 0.5907473
```

```r
prop.table(table(Personal.Loan, CreditCard), margin = 1)
```

```
##              CreditCard
## Personal.Loan         0         1
##             0 0.7106618 0.2893382
##             1 0.6975089 0.3024911
```

```r
detach(ub.data.train)
```

i. P(CC = 1 | Loan = 1) (CC=1 means CreditCard holder, Loan=1 means Loan was accepted) Ans. = 0.30249

ii. P(Online = 1 | Loan = 1) Ans. = 0.59075

iii. P(Loan = 1) (the proportion of loan acceptors) Ans. = (77+119+38+47)/3001 = 0.09363545

iv. P(CC = 1 | Loan = 0) Ans. = 0.2893382

v. P(Online = 1 | Loan = 0) Ans. = 0.5974265

vi. P(Loan = 0) Ans. = (786+309+1147+478)/3001 = 0.9063645 or = 1-P(Loan=1) = 1-0.09363545 = 0.9063646

PART E: P(Loan = 1 | CC= 1, Online = 1) = [P(CC=1 , Online=1 | Loan = 1)*P(Loan=1)]/P(CC=1 , Online=1)

= [P(CC=1 | Loan=1) * P(Online=1 | Loan=1)* P(Loan=1)]/[P(CC=1 | Loan=1) * P(Online=1 | Loan=1) * P(Loan=1) +P(CC=1 | Loan=0) * P(Online=1 | Loan=0)*P(Loan=0)]

=((0.3024911) * (0.59075) * (0.09363545))/(((0.3024911) * (0.59075) * (0.09363545))+((0.2893382) * (0.5974265) * (0.9063645)))

= 0.09649284

*******PART F: The Results between the Pivot table and calculated Naive Bayes value differs. For the Naive Bayes a value probability of 0.09649284 was achieved whereas the Pivot Table indicated a probability of 0.08952381 for the P(Loan=1 | CC=1 , Online=1) statement. The Naive Bayes calculation is higher than the value in the Pivot table. The Naive Bayes calculation is not exact as we are making assumptions. Only the numerator is approximated in this calculation. The probability value in the pivot table is therefore more accurate.

PART G:

```r
UBdata.nb <- naiveBayes(Personal.Loan~., data = ub.data.train)# Creating the NAivive Bayes Model on the
UBdata.nb
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##          0          1
## 0.90636455 0.09363545
##
## Conditional probabilities:
##    Online
## Y           0         1
##   0 0.4025735 0.5974265
##   1 0.4092527 0.5907473
##
##    CreditCard
## Y           0         1
##   0 0.7106618 0.2893382
##   1 0.6975089 0.3024911
```

Naive Bayes Calculation of P(Loan = 1 | CC = 1, Online = 1) = [P(CC=1 , Online=1 | Loan = 1)*P(Loan=1)]/P(CC=1 , Online=1)

= ((0.3024911) * (0.5907473) * (0.09363545)) / (((0.3024911) * (0.5907473) * (0.09363545))+((0.2893382) * (0.5974265) * (0.90636455)))

= 0.09649244

Number in E = 0.09649284

As we can see the values are essentially the same between Part E and Part G values.