Naive Bayes

Dutt Thakkar

2023-03-05

#Installing required packages

```
library(caret)
```

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

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

```
library(ISLR)
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
```

```
library(ggplot2)
library(tinytex)
library(reshape2)
```

#Importing data

```
library(readr)
UniversalBank=read.csv("/Users/duttthakkar/Desktop/Business Analytics/Machine Learnin
g/Assignment 2/UniversalBank.csv")
df=UniversalBank
summary(df)
```

```
##        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   :93152
##  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
```

#Converting Personal.Loan, CreditCard, and Online into factor using as.factor function

```
df$Personal.Loan<-as.factor(df$Personal.Loan)
df$Online<-as.factor(df$Online)
df$CreditCard<-as.factor(df$CreditCard)
```

#Partitioning data into 60% training and 40% validation set

```
set.seed(123)
train.index=createDataPartition(df$Personal.Loan, p=0.6, list = FALSE)
validation.index=setdiff(row.names(df),train.index)
train.df=df[train.index,]
validation.df=df[validation.index,]
nrow(train.df)
```

```
## [1] 3000
```

#Question1: Creating pivot table for the training data with Online as a column variable, CreditCard as row variable, and loan as secondary row variable.

```
partition.bank=melt(train.df, id.vars = c("CreditCard","Personal.Loan"), measure.vars
= "Online")
pivot.table=dcast(partition.bank, CreditCard + Personal.Loan ~ variable, fun.aggregat
e = length)
pivot.table
```

```
##   CreditCard Personal.Loan Online
## 1          0             0   1935
## 2          0             1    204
## 3          1             0    777
## 4          1             1     84
```

```
Bank=ftable(df$CreditCard, df$Personal.Loan, df$Online)
Bank
```

```
##         0    1
##
## 0 0  1300 1893
##   1   128  209
## 1 0   527  800
##   1    61   82
```

#Question2: Considering 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 service (Online = 1)]

```
p.acceptance=(82/800)
p.acceptance
```

```
## [1] 0.1025
```

#The probability of loan acceptance conditional on having a bank credit card and being an active user of online banking service is 10.25%

#Question 3: Creating 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.

```
Loan_online <- addmargins(table(train.df[,c(13,10)]))
Loan_online
```

```
##       Personal.Loan
## Online    0    1  Sum
##    0   1101  112 1213
##    1   1611  176 1787
##    Sum 2712  288 3000
```

```
Loan_CC <- addmargins(table(train.df[,c(14,10)]))
Loan_CC
```

```
##           Personal.Loan
## CreditCard    0    1  Sum
##        0   1935  204 2139
##        1    777   84  861
##        Sum 2712  288 3000
```

#Question 4: Computing the following quantities [P(A | B) means “the probability ofA given B”]

```
#P(CC = 1 | Loan = 1) (the proportion of credit card holders among the loan acceptor
s)
I=(91/278)
paste("P (CC = 1 | Loan = 1) =", round(I*100,2),"%")
```

```
## [1] "P (CC = 1 | Loan = 1) = 32.73 %"
```

```
#P(Online = 1 | Loan = 1)
II=(179/278)
paste("P(Online=1|Loan=1) = ", round(II*100,2),"%")
```

```
## [1] "P(Online=1|Loan=1) =  64.39 %"
```

```
#P(Loan = 1) (the proportion of loan acceptors)
III=(278/3000)
paste("P (Loan = 1) = ", round(III*100,2),"%")
```

```
## [1] "P (Loan = 1) =  9.27 %"
```

```
#P(CC = 1 | Loan = 0)
IV=(792/2722)
paste("P(CC=1|Loan=0) = ", round(IV*100,2),"%")
```

```
## [1] "P(CC=1|Loan=0) =  29.1 %"
```

```
#P(Online = 1 | Loan = 0)
V=(1620/2722)
paste("P(Online=1|Loan=0) = ", round(V*100,2),"%")
```

```
## [1] "P(Online=1|Loan=0) =  59.52 %"
```

```
#P(Loan=0)
VI=(2722/3000)
paste("P(Loan=0) = ", round(VI*100,2),"%")
```

```
## [1] "P(Loan=0) =  90.73 %"
```

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

```
Naive_Bay_Prob <- ((I*II*III)/((I*II*III)+(IV*V*VI)))
Naive_Bay_Prob
```

```
## [1] 0.1105637
```

#Naive Bayes probability is 11.06%

#Question 6: Compare this value with the one obtained from the pivot table in (B). Which is a more accurate estimate? # 10.25% and 11.06 are very close and is comparable. The Naive Bayes method's predictions might be more adaptable, but they might also be less accurate because of the simplifying assumption of independence across features

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

```
NB.train= train.df[,c(10,13:14)]
NB.validation=validation.df[,c(10,13:14)]
N_bayes = naiveBayes(Personal.Loan~.,data=NB.train)
N_bayes
```

```
##
## 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
```

#Probability calculation from Naive Bayes model

```
Naive_Bayes = (0.4700881*0.4797134*0.092)/((0.4700881*0.4797134*0.092)+(0.4542897*0.4
909531*0.907))
Naive_Bayes
```

```
## [1] 0.09301808
```

#We got very close output as compared to what we received in Previous methods because the joint and marginal probabilities we calculated in question 5 are only slight different as given by the Naive Bayes function.