MIS 64060: Assignment_3: Naive Bayes for classification

Eyob Tadele

10/15/2021

Project Objective

The purpose of this assignment is to use Naive Bayes for classification.

The file UniversalBank.csv contains data on 5000 customers of Universal Bank. The data include customer demographic information (age, income, etc.), the customer's relationship with the bank (mortgage, securities account, etc.), and the customer response to the last personal loan campaign (Personal Loan). Among these 5000 customers, only 480 (= 9.6%) accepted the personal loan that was offered to them in the earlier campaign. In this exercise, we focus on two predictors: Online (whether or not the customer is an active user of online banking services) and Credit Card (abbreviated CC below) (does the customer hold a credit card issued by the bank), and the outcome Personal Loan (abbreviated Loan below).

Importing a UniversalBank.csv dataset into r, load relevant libraries, and printout stats about the data.

```
library(caret)
```

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

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

```
library(e1071)

univBank <- read.csv("UniversalBank.csv")
str(univBank)
```

```
## '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 ...
```

```
head(univBank)
```

```
##   ID Age Experience Income ZIP.Code Family CCAvg Education Mortgage
## 1  1  25          1     49    91107      4   1.6         1        0
## 2  2  45         19     34    90089      3   1.5         1        0
## 3  3  39         15     11    94720      1   1.0         1        0
## 4  4  35          9    100    94112      1   2.7         2        0
## 5  5  35          8     45    91330      4   1.0         2        0
## 6  6  37         13     29    92121      4   0.4         2      155
##   Personal.Loan Securities.Account CD.Account Online CreditCard
## 1             0                  1          0      0          0
## 2             0                  1          0      0          0
## 3             0                  0          0      0          0
## 4             0                  0          0      0          0
## 5             0                  0          0      0          1
## 6             0                  0          0      1          0
```

Transforming the two predictors(Online,Credit Card) and the outcome(Personal.Loan) into factors.

```
univBank$CreditCard <- factor(univBank$CreditCard)
univBank$Online <- factor(univBank$Online)
univBank$Personal.Loan <- factor(univBank$Personal.Loan)
str(univBank)
```

```
## '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 validation (40%) sets.

```
selVars <- c(10,13,14)    # selecting Online, Credit Card, and Personal.Loan
set.seed(420)
univBank.part.mod <- createDataPartition(univBank$Personal.Loan,p=0.6, list = FALSE)
univBank.tr <- univBank[univBank.part.mod,selVars]
univBank.va <- univBank[-univBank.part.mod,selVars]
summary(univBank.tr)
```

```
##  Personal.Loan Online   CreditCard
##  0:2712        0:1217   0:2110
##  1: 288        1:1783   1: 890
```

```
summary(univBank.va)
```

```
##  Personal.Loan Online   CreditCard
##  0:1808        0: 799   0:1420
##  1: 192        1:1201   1: 580
```

Part A: 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.

```
# creating a pibot table using the ftable() function
attach(univBank.tr)
ftable(Online,CreditCard,Personal.Loan)
```

```
##                   Personal.Loan    0    1
## Online CreditCard
## 0      0                         783   72
##        1                         323   39
## 1      0                        1130  125
##        1                         476   52
```

```
detach(univBank.tr)
```

Part B: Looking at the pivot table, calculate the probability of loan offer acceptance for a customer who owns a bank credit card and is an active online banking services user.

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).

P(L=1 | C=1,O=1) = 52/528 = 0.0984

Part C: 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.

```
attach(univBank.tr)
ftable(Online,Personal.Loan)
```

```
##        Personal.Loan    0    1
## Online
## 0                     1106  111
## 1                     1606  177
```

```
ftable(CreditCard,Personal.Loan)
```

```
##            Personal.Loan    0    1
## CreditCard
## 0                         1913  197
## 1                          799   91
```

```
detach(univBank.tr)
```

Part D: Compute the conditional probabilities of:

i. P(CC = 1 | Loan = 1) (the proportion of credit card holders among the loan acceptors)

```
P(CC = 1 | Loan = 1) = 91/288 =  0.3159
```

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

```
P(Online = 1 | Loan = 1) = 177/288 = 0.6145
```

iii. P(Loan = 1) (the proportion of loan acceptors)

P(Loan = 1) = 288/3000 = 0.096

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

P(CC = 1 | Loan = 0) = 799/2712 = 0.2946

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

P(Online = 1 | Loan = 0) = 1606/2712 = 0.5921

vi. P(Loan = 0)

P(Loan = 0) = 2712/3000 = 0.904

Part E: Use the quantities computed above to compute the naive Bayes probability P(Loan = 1 | CC = 1, Online = 1).

P(L=1|CC=1,O=1) = P(CC=1|L=1) * P(O=1|L=1) * P(L=1) / P(CC=1|L=1) * P(O=1|L=1) * P(L=1) + P(CC=1|L=0) * P(O=1|L=0) * P(L=0) = (91/288)(177/288)(288/3000) / ((91/288)(177/288)(288/3000) + (799/2712)(1606/2712)(2712/3000))

P(L=1|CC=1,O=1) = 0.1057

Part F: Compare this value with the one obtained from the pivot table in (B). Which is a more accurate estimate?

The value obtained through naive Bayes method is 0.1057, and 0.0984 using direct method. Naive Bayes makes an assumption that all the features are independent of eachother and that may not be the case in real lifecycle. Making that assumption may make that a little less accurate.

Part G: Which of the entries in this table are needed for computing P(Loan = 1 | CC = 1, Online = 1)?

The entries needed are: P(CC=1|L=1), P(O=1|L=1), P(L=1), P(CC=1|L=0), P(O=1|L=0), P(L=0)

Part G: 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).

```
univBank.nb <- naiveBayes(Personal.Loan ~ ., data = univBank.tr)  # using the naiveBayes model
univBank.nb
```

```
##
## 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 0.4078171 0.5921829
##   1 0.3854167 0.6145833
##
##    CreditCard
## Y           0         1
##   0 0.7053835 0.2946165
##   1 0.6840278 0.3159722
```

Calculating values from the naiveBayes model probability tables results in a value of 0.1057 : (0.3159722)(0.6145833)(0.096) / ((0.3159722)(0.6145833)(0.096)+(0.2946165)(0.5921829)(0.904)) = 0.1057

This is the same as the value obtained in E.