Assignment _3

Andrew Gutierrez

2022-10-17

Note that my responses to the applicable assignment prompts are actually contained in the 'A.Gutierrez Assignment 3 Responses' TXT file that is also included in my GitHub folder for this assignment.

First, I'll install the requisite libraries:

```
library(caret)
```

```
## Loading required package: ggplot2

## Loading required package: lattice
```

```
library(ISLR)
library(e1071)
library(reshape)
library(reshape2)
```

```
##
## Attaching package: 'reshape2'

## The following objects are masked from 'package:reshape':
##
##     colsplit, melt, recast
```

Then, I'll read the UniversalBank.csv file into a DataFrame in R:

```
##### Note: the below file path may need to be adjusted, as it currently references a local location on
UB = read.csv("C:\\Users\\gutiera9\\Documents\\MSBA KSU\\UniversalBank.csv",header=T,sep=",")

head(UB)
```

```
##   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
```

And as my final step for preparation, I'll split the data into training and test sets (60-40 split)

```
test_index = createDataPartition(UB$Personal.Loan,p=0.4,list=FALSE) # Set aside 40% for the test
TestData = UB[test_index,]
TrainData = UB[-test_index,] # Remaining data becomes the Training set

print('Summary of Training Data Set: ')
```

```
## [1] "Summary of Training Data Set: "
```

```
summary(TrainData)
```

```
##        ID            Age          Experience        Income          ZIP.Code    
##  Min.   :   3   Min.   :23.0   Min.   :-3.00   Min.   :  8.00   Min.   : 9307  
##  1st Qu.:1232   1st Qu.:35.0   1st Qu.:10.00   1st Qu.: 38.00   1st Qu.:91910  
##  Median :2506   Median :45.0   Median :20.00   Median : 62.00   Median :93407  
##  Mean   :2507   Mean   :45.3   Mean   :20.07   Mean   : 72.47   Mean   :93138  
##  3rd Qu.:3774   3rd Qu.:55.0   3rd Qu.:30.00   3rd Qu.: 95.00   3rd Qu.:94608  
##  Max.   :5000   Max.   :67.0   Max.   :43.00   Max.   :224.00   Max.   :96651  
##      Family          CCAvg          Education        Mortgage     
##  Min.   :1.000   Min.   : 0.00   Min.   :1.000   Min.   :  0.00  
##  1st Qu.:1.000   1st Qu.: 0.67   1st Qu.:1.000   1st Qu.:  0.00  
##  Median :2.000   Median : 1.50   Median :2.000   Median :  0.00  
##  Mean   :2.415   Mean   : 1.93   Mean   :1.896   Mean   : 56.23  
##  3rd Qu.:3.000   3rd Qu.: 2.50   3rd Qu.:3.000   3rd Qu.:102.00  
##  Max.   :4.000   Max.   :10.00   Max.   :3.000   Max.   :635.00  
##  Personal.Loan   Securities.Account   CD.Account           Online     
##  Min.   :0.000   Min.   :0.0000     Min.   :0.00000   Min.   :0.000  
##  1st Qu.:0.000   1st Qu.:0.0000     1st Qu.:0.00000   1st Qu.:0.000  
##  Median :0.000   Median :0.0000     Median :0.00000   Median :1.000  
##  Mean   :0.093   Mean   :0.1013     Mean   :0.05767   Mean   :0.607  
##  3rd Qu.:0.000   3rd Qu.:0.0000     3rd Qu.:0.00000   3rd Qu.:1.000  
##  Max.   :1.000   Max.   :1.0000     Max.   :1.00000   Max.   :1.000  
##    CreditCard    
##  Min.   :0.0000  
##  1st Qu.:0.0000  
##  Median :0.0000  
##  Mean   :0.2917  
##  3rd Qu.:1.0000  
##  Max.   :1.0000  
```

```
print('Summary of Test Data Set: ')
```

```
## [1] "Summary of Test Data Set: "
```

```
summary(TestData)
```

```
##        ID            Age          Experience        Income      
##  Min.   :   1   Min.   :23.00   Min.   :-3.00   Min.   :  8.00  
##  1st Qu.:1279   1st Qu.:35.00   1st Qu.:10.00   1st Qu.: 40.00  
##  Median :2490   Median :45.00   Median :20.00   Median : 65.00  
##  Mean   :2491   Mean   :45.39   Mean   :20.15   Mean   : 75.73  
##  3rd Qu.:3708   3rd Qu.:55.00   3rd Qu.:30.00   3rd Qu.:102.00  
##  Max.   :4998   Max.   :67.00   Max.   :43.00   Max.   :204.00  
##     ZIP.Code         Family          CCAvg         Education    
##  Min.   :90005   Min.   :1.000   Min.   :0.00   Min.   :1.000  
##  1st Qu.:92007   1st Qu.:1.000   1st Qu.:0.70   1st Qu.:1.000  
##  Median :93555   Median :2.000   Median :1.60   Median :2.000  
##  Mean   :93174   Mean   :2.369   Mean   :1.95   Mean   :1.859  
##  3rd Qu.:94611   3rd Qu.:3.000   3rd Qu.:2.60   3rd Qu.:3.000  
##  Max.   :96651   Max.   :4.000   Max.   :9.30   Max.   :3.000  
##     Mortgage       Personal.Loan    Securities.Account   CD.Account    
##  Min.   :  0.00   Min.   :0.0000   Min.   :0.000      Min.   :0.0000  
##  1st Qu.:  0.00   1st Qu.:0.0000   1st Qu.:0.000      1st Qu.:0.0000  
##  Median :  0.00   Median :0.0000   Median :0.000      Median :0.0000  
##  Mean   : 56.91   Mean   :0.1005   Mean   :0.109      Mean   :0.0645  
##  3rd Qu.: 98.00   3rd Qu.:0.0000   3rd Qu.:0.000      3rd Qu.:0.0000  
##  Max.   :617.00   Max.   :1.0000   Max.   :1.000      Max.   :1.0000  
##      Online         CreditCard    
##  Min.   :0.0000   Min.   :0.0000  
##  1st Qu.:0.0000   1st Qu.:0.0000  
##  Median :1.0000   Median :0.0000  
##  Mean   :0.5815   Mean   :0.2975  
##  3rd Qu.:1.0000   3rd Qu.:1.0000  
##  Max.   :1.0000   Max.   :1.0000
```

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. The values inside the table should convey the count.

For this problem, I'll use the melt() and cast() functions in R.

```
pivot <- melt(TrainData,id=c("Online", "CreditCard","Personal.Loan"))

UB_pivot <- dcast(pivot,Personal.Loan+CreditCard~Online,length)
UB_pivot[,3:4] <- UB_pivot[,3:4]/11
print(UB_pivot)
```

```
##   Personal.Loan CreditCard   0    1
## 1             0          0 770 1168
## 2             0          1 296  487
## 3             1          0  73  114
## 4             1          1  40   52
```

B. 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)].

For this problem, I'll divide the total number of instances where customers in the dataset have accepted a personal loan, have a bank credit card, and uses online services by the total number of instances where a customer has a bank credit card and uses online services.

```
print(UB_pivot[4,4] / (UB_pivot[4,4] + UB_pivot[2,4]))
```

```
## [1] 0.09647495
```

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.

Instead of melt() and cast(), I'll use the table function for this problem.

```
# Personal Loan / Online pivot table
loanOnline <- table(TrainData[,c(13,10)])

# Personal Loan / CC pivot table
loanCC <- table(TrainData[,c(14,10)])

print("Personal loan as a function of online: ")
```

```
## [1] "Personal loan as a function of online: "
```

```
print(loanOnline)
```

```
##       Personal.Loan
## Online    0    1
##      0 1066  113
##      1 1655  166
```

```
print("Personal loan as a function of credit card: ")
```

```
## [1] "Personal loan as a function of credit card: "
```

```
print(loanCC)
```

```
##           Personal.Loan
## CreditCard    0    1
##          0 1938  187
##          1  783   92
```

D. Compute the following quantities [P(A | B) means "the probability of A 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)

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

```
## [1] "i. P(CC = 1 | Loan = 1) (the proportion of credit card holders among the loan acceptors): "
```

```r
print(loanCC[2,2] / (loanCC[1,2]+loanCC[2,2]))
```

```
## [1] 0.3297491
```

```r
print("ii. P(Online = 1 | Loan = 1): ")
```

```
## [1] "ii. P(Online = 1 | Loan = 1): "
```

```r
print(loanOnline[2,2] / (loanOnline[1,2]+loanOnline[2,2]))
```

```
## [1] 0.5949821
```

```r
print("iii. P(Loan = 1) (the proportion of loan acceptors): ")
```

```
## [1] "iii. P(Loan = 1) (the proportion of loan acceptors): "
```

```r
print((loanCC[1,2]+loanCC[2,2]) / 3000)
```

```
## [1] 0.093
```

```r
print("iv. P(CC = 1 | Loan = 0): ")
```

```
## [1] "iv. P(CC = 1 | Loan = 0): "
```

```r
print(loanCC[2,1] / (loanCC[1,1]+loanCC[2,1]))
```

```
## [1] 0.2877619
```

```r
print("v. P(Online = 1 | Loan = 0): ")
```

```
## [1] "v. P(Online = 1 | Loan = 0): "
```

```r
print(loanOnline[2,1] / (loanOnline[1,1]+loanOnline[2,1]))
```

```
## [1] 0.6082323
```

```r
print("vi. P(Loan = 0): ")
```

```
## [1] "vi. P(Loan = 0): "
```

```r
print((loanCC[1,1]+loanCC[2,1]) / 3000)
```

```
## [1] 0.907
```

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

Now, I'll plug in the values from the above pivot table into the Bayes probability formula below.

```r
print("P(Loan = 1 | CC = 1, Online = 1): ")
```

```
## [1] "P(Loan = 1 | CC = 1, Online = 1): "
```

```r
((loanCC[2,2] / (loanCC[1,2]+loanCC[2,2]))*(loanOnline[2,2] / (loanOnline[1,2]+loanOnline[2,2]))*((loanC
```

```
## [1] 0.1030885
```

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

Note that the answer to this question is contained in the 'A.Gutierrez Assignment 3 Responses' TXT file that is also included in my GitHub folder for this assignment.

G. 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).

```r
#Build a naive Bayes classifier
nb_model <- naiveBayes(Personal.Loan~Online+CreditCard,data = TrainData)
nb_model
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##     0     1
## 0.907 0.093
##
## Conditional probabilities:
##    Online
## Y        [,1]      [,2]
##   0 0.6082323 0.4882350
##   1 0.5949821 0.4917776
##
##    CreditCard
## Y        [,1]      [,2]
##   0 0.2877619 0.4528027
##   1 0.3297491 0.4709667
```