Assignment_3

2022-10-17

```
library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(e1071)
library(class)
library(tidyverse)

## ── Attaching packages
## ───────────────────────────────────────
## tidyverse 1.3.2 ──

## ✔ tibble  3.1.8      ✔ dplyr   1.0.10
## ✔ tidyr   1.2.1      ✔ stringr 1.4.1
## ✔ readr   2.1.3      ✔ forcats 0.5.2
## ✔ purrr   0.3.5
## ── Conflicts ──────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ✖ purrr::lift()   masks caret::lift()

#Importing dataset.
universaldata <- read.csv("UniversalBank.csv")
str(universaldata)

## '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 integer Variables to Factor Variables.

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

str(universaldata)

## '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 ...

#Data partition into Training and validation into 60% and 40% respectively.

set.seed(123)

index_train <- createDataPartition(universaldata$Personal.Loan, p=0.60, list
= FALSE)
train <- universaldata[index_train,]
validate <- universaldata[-index_train,]

#Normalization of data for current dataset
universal_norm <- preProcess(train[,-c(10,13,14)], method = "center",
"scale")
universal_tpredict <- predict(universal_norm, train)
universal_vpredict <- predict(universal_norm, validate)

#A. Pivot table for Personal.Loan, Online, CreditCard.
ptable <- ftable(universal_tpredict$Personal.Loan, universal_tpredict$Online,
universal_tpredict$CreditCard, dnn=c('Personal.loan', 'CreditCard',
'Online'))
ptable

##                            Online    0    1
## Personal.loan CreditCard
## 0             0                    791  310
##               1                   1144  467
```

```
## 1             0                    79   33
##               1                   125   51
```

#B.Probability of Loan=1, Online=1, CreditCard=1 is: 51/467+51 = 0.09845559845

```
#C.Two tables for personal loan:online and personal loan:credit card

ptable1 <- ftable(universal_tpredict$Personal.Loan,
universal_tpredict$Online, dnn=c('Personal.loan','Online'))
ptable1

##               Online    0    1
## Personal.loan
## 0                    1101 1611
## 1                     112  176

ptable2 <- ftable(universal_tpredict$Personal.Loan,
universal_tpredict$CreditCard, dnn=c('Personal.loan','CreditCard'))
ptable2

##               CreditCard    0    1
## Personal.loan
## 0                        1935  777
## 1                         204   84
```

#D.(i).P(CC=1 | Loan= 1) = 84/288=0.2916667

#D.(ii).P(Online=1 | Loan=1)=176/288=0.611111

#D.(iv).P(CC=1 | Loan=0)= 777/2712= 0.2865044

#D.(v).P(Online=1 | Loan=0)= 1611/2712= 0.5940265

```
#D.(iii).P(Loan=1),(vi).P(Loan=0)
ptable3 <-ftable(universal_tpredict$Personal.Loan, dnn= 'Personal.loan')
ptable3

## Personal.loan    0    1
##
##               2712  288

#P(Loan=1)= 288/3000= 0.096
#P(Loan=0)= 2712/3000= 0.904

#E.Calculating Naive Bayes from above.
#Naive Bayes=
0.2916667*0.611111*0.096/(0.2916667*0.611111*0.096+0.2865044*0.5940265*0.904)
= 0.10008607928
```

#F.The probability from B. is 0.09845559845 and E. 0.10008607928. The answer from E. higher than B.

```
#G.Running naives bayes directly on data set.
naive <- naiveBayes(Personal.Loan~Online+CreditCard, data =
universal_tpredict)
naive

##
## 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

#The probability derived from E. is equal to the answer derived in G.
```