Assignment\_3

Keerthana

2024-03-11

UNIVERSAL\_Bank <- read.csv("/Users/keerthanavonteddu/Desktop/assignment-3\_FML/UniversalBank.csv")  
summary(UNIVERSAL\_Bank)

## 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

#Loading the packages

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(ISLR)

library(e1071)

library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(class)

library(ggplot2)

library(tidyr)

library(gmodels)

library(lattice)

UNIVERSAL\_Bank$Personal.Loan <- factor(UNIVERSAL\_Bank$Personal.Loan)  
UNIVERSAL\_Bank$Online <- factor(UNIVERSAL\_Bank$Online)  
UNIVERSAL\_Bank$CreditCard <- factor(UNIVERSAL\_Bank$CreditCard)

# 1.

set.seed(312)  
train\_of\_Index <- createDataPartition(UNIVERSAL\_Bank$Personal.Loan,p = 0.6,list = FALSE)  
df.Train <- UNIVERSAL\_Bank[train\_of\_Index,]  
validation.df <- UNIVERSAL\_Bank[-train\_of\_Index,]

pv\_table <- xtabs(~ CreditCard + Online + Personal.Loan,data = df.Train)  
ftable(pv\_table)

## Personal.Loan 0 1  
## CreditCard Online   
## 0 0 783 70  
## 1 1152 135  
## 1 0 297 37  
## 1 480 46

# 2.

Probabality = 46/(46+480)  
Probabality

## [1] 0.08745247

# 3.

table(Personal.Loan = df.Train$Personal.Loan, Online = df.Train$Online)

## Online  
## Personal.Loan 0 1  
## 0 1080 1632  
## 1 107 181

table(Personal.Loan = df.Train$Personal.Loan, CreditCard = df.Train$CreditCard)

## CreditCard  
## Personal.Loan 0 1  
## 0 1935 777  
## 1 205 83

table(Personal.Loan = df.Train$Personal.Loan)

## Personal.Loan  
## 0 1   
## 2712 288

# 4.

# (i)

Probability\_a <- 83/(83+205)  
Probability\_a

## [1] 0.2881944

# (ii)

Probability\_b <- 181/(181+107)  
Probability\_b

## [1] 0.6284722

# (iii)

Probability\_c <- 288/(288+2712)  
Probability\_c

## [1] 0.096

# (iv)

Probability\_d <- 777/(777+1935)  
Probability\_d

## [1] 0.2865044

# (v)

Probability\_e <- 1632/(1632+1080)  
Probability\_e

## [1] 0.6017699

# (vi)

Probability\_f <- 2712/(2712+288)  
Probability\_f

## [1] 0.904

# 5.

calculated\_probabilities <- (Probability\_a \* Probability\_b \* Probability\_c)/((Probability\_a \* Probability\_b \* Probability\_c) + (Probability\_d \* Probability\_e \* Probability\_f))  
calculated\_probabilities

## [1] 0.1003643

# 6.

In question 2, we found a #Value of 0.08745247, which is almost the same as the value of 0.1003643 in question 5. The exact method and the naïve bayes technique are the same except that the former requires each independent variable to be precisely classified in order to make a prediction, while the latter does not. We can verify that the value derived from question 2 is more accurate. Taking into account that the exact values from the pivot table were utilized.

# 7.

naviebayes.model <- naiveBayes(Personal.Loan ~ Online + CreditCard, data = df.Train)  
to.predict = data.frame(Online=1, CreditCard= 1)  
predict(naviebayes.model, to.predict,type = 'raw')

## Warning in predict.naiveBayes(naviebayes.model, to.predict, type = "raw"): Type  
## mismatch between training and new data for variable 'Online'. Did you use  
## factors with numeric labels for training, and numeric values for new data?

## Warning in predict.naiveBayes(naviebayes.model, to.predict, type = "raw"): Type  
## mismatch between training and new data for variable 'CreditCard'. Did you use  
## factors with numeric labels for training, and numeric values for new data?

## 0 1  
## [1,] 0.9081939 0.09180609

#We have the value 0.1003643 from task 5, and we have 0.9081939 from question 7. This is very much the same result that we got from assignment 5. ###Very little is changed as a result of rounding. The difference won’t have an impact on the output’s ranking order.