Fml\_assignment\_3

Abhinav

2024-03-10

UN\_bank <- read.csv("/Users/keerthanavonteddu/Desktop/abhinav/UniversalBank.csv")  
summary(UN\_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

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)

UN\_bank$Personal.Loan <- factor(UN\_bank$Personal.Loan)  
UN\_bank$Online <- factor(UN\_bank$Online)  
UN\_bank$CreditCard <- factor(UN\_bank$CreditCard)

# 1.

set.seed(231)  
Index\_train <- createDataPartition(UN\_bank$Personal.Loan,p = 0.6,list = FALSE)  
Train\_df <- UN\_bank[Index\_train,]  
validation.df <- UN\_bank[-Index\_train,]

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

## Personal.Loan 0 1  
## CreditCard Online   
## 0 0 780 81  
## 1 1150 117  
## 1 0 308 42  
## 1 474 48

# 2.

probabality = 48/(48+474)  
probabality

## [1] 0.09195402

# 3.

table(Personal.Loan = Train\_df$Personal.Loan, Online = Train\_df$Online)

## Online  
## Personal.Loan 0 1  
## 0 1088 1624  
## 1 123 165

table(Personal.Loan = Train\_df$Personal.Loan, CreditCard = Train\_df$CreditCard)

## CreditCard  
## Personal.Loan 0 1  
## 0 1930 782  
## 1 198 90

table(Personal.Loan = Train\_df$Personal.Loan)

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

# 4.

# (i)

Probabality.1 <- 90/(90+198)  
Probabality.1

## [1] 0.3125

# (ii)

Probabality.2 <- 165/(165+123)  
Probabality.2

## [1] 0.5729167

# (iii)

Probabality.3 <- 288/(288+2712)  
Probabality.3

## [1] 0.096

# (iv)

Probabality.4 <- 782/(782+1930)  
Probabality.4

## [1] 0.2883481

# (v)

Probabality.5 <- 1624/(1624+1088)  
Probabality.5

## [1] 0.5988201

# (vi)

Probability.6 <- 2712/(2712+288)  
Probability.6

## [1] 0.904

# 5.

calculated\_probabilities <- (Probabality.1 \* Probabality.2 \* Probabality.3)/((Probabality.1 \* Probabality.2 \* Probabality.3) + (Probabality.4 \* Probabality.5 \* Probability.6))  
calculated\_probabilities

## [1] 0.09918921

# 6.

We obtained a #Value of 0.09195402 from question 2, which is nearly identical to the value of 0.09918921 from question 5. The precise approach and the naïve bayes technique are identical in every way, however the former requires the precise categorization of every independent variable for prediction, while the latter does not. The figure obtained from question 2 is more accurate, as we can validate. Considering that the pivot table’s precise values were used.

# 7.

naviebayes.model <- naiveBayes(Personal.Loan ~ Online + CreditCard, data = Train\_df)  
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,] 0.9070972 0.09290275

#The value we obtained from task 5 is 0.09918921, and the value we get from question 7 is 0.9070972. The outcome is nearly identical to what we obtained from task 5. #The rounding results in a very little change. The output’s rank order will remain unaffected by the discrepancy.