ASSIGNMENT - 3

Sai Harshitha Akula

2023-11-06

#Importing the dataset.

Univ\_Bank\_1 <- read.csv("C:/Users/saiha/OneDrive/Documents/R PROGRAMMING/UniversalBank-1.csv")

#Loading the required packages.

library("caret")

## Warning: package 'caret' was built under R version 4.3.2

## Loading required package: ggplot2

## Loading required package: lattice

library("ISLR")  
library("ggplot2")  
library("class")  
library("lattice")  
library("reshape2")

## Warning: package 'reshape2' was built under R version 4.3.2

library("melt")

## Warning: package 'melt' was built under R version 4.3.2

#Transforming to factor variable.

Univ\_Bank\_1$Personal.Loan <- as.factor(Univ\_Bank\_1$Personal.Loan)  
Univ\_Bank\_1$Online <- as.factor(Univ\_Bank\_1$Online)  
Univ\_Bank\_1$CreditCard <- as.factor(Univ\_Bank\_1$CreditCard)

#Checking the summary of the dataset.

summary(Univ\_Bank\_1)

## 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 :93153   
## 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 Personal.Loan  
## Min. :1.000 Min. : 0.000 Min. :1.000 Min. : 0.0 0:4520   
## 1st Qu.:1.000 1st Qu.: 0.700 1st Qu.:1.000 1st Qu.: 0.0 1: 480   
## 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   
## Securities.Account CD.Account Online CreditCard  
## Min. :0.0000 Min. :0.0000 0:2016 0:3530   
## 1st Qu.:0.0000 1st Qu.:0.0000 1:2984 1:1470   
## Median :0.0000 Median :0.0000   
## Mean :0.1044 Mean :0.0604   
## 3rd Qu.:0.0000 3rd Qu.:0.0000   
## Max. :1.0000 Max. :1.0000

#Splitting the 100% of data into training and testing.  
#60% for training and 40% for validation.

set.seed(23)

Split\_Index <- createDataPartition(Univ\_Bank\_1$Personal.Loan, p = 0.6, list = FALSE)  
Train\_data <- Univ\_Bank\_1[Split\_Index,]  
Validation\_data <- Univ\_Bank\_1[-Split\_Index,]

dim(Train\_data)

## [1] 3000 14

dim(Validation\_data)

## [1] 2000 14

normalising1 <- preProcess(Train\_data[,-c(10,13:14)],method=c("center","scale"))  
Train\_data1 <-predict(normalising1,Train\_data)  
Validation\_data1 <-predict(normalising1,Validation\_data)

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. The values inside the table should convey the count. In R use functions melt() and cast(), or function table(). In Python, use panda dataframe methods melt() and pivot().

PTable\_1 <- ftable(Train\_data1[,c(14,10,13)])  
PTable\_1

## Online 0 1  
## CreditCard Personal.Loan   
## 0 0 773 1127  
## 1 82 114  
## 1 0 315 497  
## 1 39 53

#Here 14- CreditCard, 10- Personal.Loan, 13- Online

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?

Ans).Utilizing the data from the pivot table, we can calculate the likelihood of the customer accepting the loan offer as 52 / (52 + 503), resulting in a probability of 0.096.

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.

melt\_1 <- melt(Train\_data1,id=c("Personal.Loan"),variable="Online")

## Warning: attributes are not identical across measure variables; they will be  
## dropped

melt\_2 <- melt(Train\_data1,id=c("Personal.Loan"), variable="CreditCard")

## Warning: attributes are not identical across measure variables; they will be  
## dropped

cast\_1 = dcast(melt\_1, Personal.Loan~Online)

## Aggregation function missing: defaulting to length

cast\_2 <- dcast(melt\_2, Personal.Loan~CreditCard)

## Aggregation function missing: defaulting to length

D).D. Compute the following quantities [P(A | B) means “the probability ofA 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)

ftable(Train\_data1[,c(10,13)])

## Online 0 1  
## Personal.Loan   
## 0 1088 1624  
## 1 121 167

ftable(Train\_data1[,c(10,14)])

## CreditCard 0 1  
## Personal.Loan   
## 0 1900 812  
## 1 196 92

1. P(CC = 1 | Loan = 1) = (92/92+196) = 0.319
2. P(Online = 1 | Loan = 1) = (167/167+121) = 0.579
3. P(Loan = 1) = (288/288+2712) = 0.096
4. P(CC = 1 | Loan = 0) = (812/812+1900) = 0.299
5. P(Online = 1 | Loan = 0) = (1624/ 1624+1088) = 0.598
6. P(Loan = 0) = (2712/ 2712+288) = 0.904

E). Use the quantities computed above to compute the naive Bayes probability P(Loan = 1 | CC = 1, Online = 1). Ans). (0. 319\* 0.579\* 0.096) / (0.319 \* 0.579 \* 0.096) + (0.299 \* 0.598 \* 0.904) = 0.098

F).Compare this value with the one obtained from the pivot table in (B). Which is a more accurate estimate? Ans).In section B, we obtained a probability value of 0.096, and in the preceding question, we calculated a probability value of 0.098. While these values exhibit slight variations, it’s important to note that in part B, we considered a more comprehensive set of dependent information. Therefore, we can confidently assert that the value derived in part B is more accurate and specific in representing the underlying data.

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).

library("naivebayes")

## Warning: package 'naivebayes' was built under R version 4.3.2

## naivebayes 0.9.7 loaded

naive\_b <- naive\_bayes(Personal.Loan~Online+CreditCard,data=Train\_data1)  
naive\_b

##   
## ================================== Naive Bayes ==================================   
##   
## Call:   
## naive\_bayes.formula(formula = Personal.Loan ~ Online + CreditCard,   
## data = Train\_data1)  
##   
## ---------------------------------------------------------------------------------   
##   
## Laplace smoothing: 0  
##   
## ---------------------------------------------------------------------------------   
##   
## A priori probabilities:   
##   
## 0 1   
## 0.904 0.096   
##   
## ---------------------------------------------------------------------------------   
##   
## Tables:   
##   
## ---------------------------------------------------------------------------------   
## ::: Online (Bernoulli)   
## ---------------------------------------------------------------------------------   
##   
## Online 0 1  
## 0 0.4011799 0.4201389  
## 1 0.5988201 0.5798611  
##   
## ---------------------------------------------------------------------------------   
## ::: CreditCard (Bernoulli)   
## ---------------------------------------------------------------------------------   
##   
## CreditCard 0 1  
## 0 0.7005900 0.6805556  
## 1 0.2994100 0.3194444  
##   
## ---------------------------------------------------------------------------------

–The customer who accepts the loan, uses a credit card, and engages in online banking yields a probability of 0.096 according to the Naive Bayes Model. This result closely mirrors the value obtained in section E of our analysis.