Assignment3

2024-03-08

```
Bank=read.csv("C:/Users/chand/Downloads/UniversalBank.csv")
summary(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   :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
##  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
```

```
#converting the variables into factor
Bank$Personal.Loan<-factor(Bank$Personal.Loan)
Bank$Online<-factor(Bank$Online)
Bank$CreditCard<-factor(Bank$CreditCard)
```

```
#Question1 : Create a pivot table for the training data with Online as a column variable, CC as a rowva

#1.divide the data into 60 and 40
#install.packages("caret")
set.seed(123)
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
library(class)
train_index<-createDataPartition(Bank$Personal.Loan,p=0.6,list=FALSE)
train_data<-Bank[train_index,]
test_data<-Bank[-train_index,]
#table(CC=train_data$CreditCard, online=train_data$Online, loan=train_data$Personal.Loan)
table <- xtabs(~ CreditCard + Online + Personal.Loan , data = train_data)
ftable(table)
```

```
##                   Personal.Loan    0    1
## CreditCard Online
## 0          0                     791   79
##            1                    1144  125
## 1          0                     310   33
##            1                     467   51
```

```
#Question2: Consider the task of classifying a customer who owns a bank credit card and is actively usi
prob<- 51/(51+467)
prob
```

```
## [1] 0.0984556
```

```
#Question3:Create two separate pivot tables for the training data. One will have Loan (rows) as a funct

#table only with loan info as row
table(Personal.Loan = train_data$Personal.Loan)
```

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

```
#pivot table with column online data
table(Personal.Loan = train_data$Personal.Loan, Online = train_data$Online)
```

```
##              Online
## Personal.Loan    0    1
##             0 1101 1611
##             1  112  176
```

```
#pivot table with column creditcard info
table(Personal.Loan = train_data$Personal.Loan, CC =train_data$CreditCard)
```

```
##              CC
## Personal.Loan    0    1
##             0 1935  777
##             1  204   84
```

```r
#Question4
#i. P(CC = 1 | Loan = 1
prob1<- 84/(84+204)
prob1
```

```
## [1] 0.2916667
```

```r
#ii.P(Online = 1 | Loan = 1)
prob2<-176/(176+112)
prob2
```

```
## [1] 0.6111111
```

```r
#iii. P(Loan = 1) (the proportion of loan acceptors)
prob3<-288/(288+2712)
prob3
```

```
## [1] 0.096
```

```r
#iv P(CC = 1 | Loan = 0)
prob4<-777/(777+1935)
prob4
```

```
## [1] 0.2865044
```

```r
#v P(Online = 1 | Loan = 0)
prob5<-1611/(1611+1101)
prob5
```

```
## [1] 0.5940265
```

```r
#vi P(Loan = 0)
prob6<-2712/(2712+288)
prob6
```

```
## [1] 0.904
```

```r
#Question 5:Use the quantities computed above to compute the naive Bayes probability P(Loan = 1 | CC= 1

overallprob<-(prob1*prob2*prob3)/((prob1*prob2*prob3)+(prob4*prob5*prob6))
overallprob
```

```
## [1] 0.1000861
```

QUestion6: Compare this value with the one obtained from the pivot table in (B). Which is a moreaccurate estimate

ANS:Value we got from question 2 was 0.09323583 and in the question 5 0.1000861 have some differences in it. The difference between the two methods is, exact method need the same independent variables classification to predict, whereas the naive bayes method need not to have independent variables. The Value from the question2 is more accurate because the values are the exact values from the pivot table.

```r
#Question7:Run naive Bayes on the data. Examine the model output on training data, and find the entryth
library(e1071)
model <- naiveBayes(Personal.Loan~ Online + CreditCard, data = train_data)
predict(model,data.frame(Online=1, CreditCard= 1) ,type = 'raw')
```

```
## Warning in predict.naiveBayes(model, data.frame(Online = 1, CreditCard = 1), :
## 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(model, data.frame(Online = 1, CreditCard = 1), :
## 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,] 0.9079363 0.09206369
```

The values from the both cases are similar. value from question 7 is 0.09206369 and the value from question5 is 0.1000861. The difference is due to the rounding.