Assignment 3

Snehitha Anpur

2022-10-14

```
#install.packages("e1071")
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(e1071)
library(readr)
library(caret)
```

```
## Loading required package: ggplot2

## Loading required package: lattice
```

```
library(ISLR)
UniBank=read.csv("UniversalBank.csv")

set.seed(111)

#Partitioning train and test data

traindata_index = createDataPartition(UniBank$Personal.Loan, p=0.6, list= FALSE)
traindata = UniBank[traindata_index,]
Validationdata = UniBank[-traindata_index,]
```

```
#Creating Pivot Table for CreditCard, Online and Personal Loan

table(CreditCard=traindata$CreditCard,Online=traindata$Online,Loan=traindata$Personal.Loan)
```

```
## , , Loan = 0
##
```

```
##           Online
## CreditCard    0    1
##          0  797 1099
##          1  315  497
##
## , , Loan = 1
##
##           Online
## CreditCard    0    1
##          0   79  126
##          1   33   54
```

Probability of the customer accepting the loan given the condition that credit card =1 and online=1

```
B= 54/(54+497)
B
```

```
## [1] 0.09800363
```

```
#Creating Pivot tables for Loan vs Online and Loan vs Credit Card

LoanvsOnline= table(Loan=traindata$Personal.Loan,Online=traindata$Online)
print(LoanvsOnline)
```

```
##     Online
## Loan    0    1
##    0 1112 1596
##    1  112  180
```

```
LoanvsCC=table(Loan=traindata$Personal.Loan,CreditCard=traindata$CreditCard)
print(LoanvsCC)
```

```
##     CreditCard
## Loan    0    1
##    0 1896  812
##    1  205   87
```

```
#P(CC = 1 | Loan = 1)

D1 = 87/291

#P(Online = 1 | Loan = 1)

D2 = 180/(180+112)

#P(Loan = 1)

D3 = 292/(2708+292)

#P(CC = 1 | Loan = 0)
```

```r
D4 = 812/(1896+812)

#P(Online = 1 | Loan = 0)

D5 = 1596/(1112+1596)

#P(Loan = 0)

D6 = 1-(292/(2708+292))
```

Probability of given (Loan = 1 | CC = 1, Online = 1) using Naive Bayes probability

```r
E=(D1*D2*D3)/((D1*D2*D3)+(D4*D5*D6))
E
```

```
## [1] 0.101083
```

The Value obtained in B is 0.09800363 and E is 0.101083. When comparing we see that the value obtained in finding probability by solving naive Bayes problem is more accurate than that obtained in B.

```r
library(dplyr)
#Creating the test data for creditcard=1 and online=1

Test= filter(traindata,CreditCard==1 & Online==1)

#Building Naive Bayes model on the training data

Model=naiveBayes(Personal.Loan~Online+CreditCard,data=traindata)

#Predicting the test data using the Naive Bayes model

Predicted_Test_labels <-predict(Model,Test, type = "raw")

head(Predicted_Test_labels)
```

```
##              0         1
## [1,] 0.8989553 0.1010447
## [2,] 0.8989553 0.1010447
## [3,] 0.8989553 0.1010447
## [4,] 0.8989553 0.1010447
## [5,] 0.8989553 0.1010447
## [6,] 0.8989553 0.1010447
```

Value obtained in G is 0.1010447 and E is 0.101083 which are almost same