Naive Bayes Assignment

Setting default values to get a clean output

```
knitr::opts_chunk$set(message = FALSE)
knitr::opts_chunk$set(warning = FALSE)
```

Loading all the required packages as well as the data

```
library("class")
library("caret")
library("e1071")
library("dplyr")
library("ggplot2")
library("gmodels")
library("melt")
library("reshape")
library("reshape2")
library("readr")
library("ISLR")
library("pROC")


data.df <- read.csv("UniversalBank.csv")
```

Data Cleaning and Normalization

```
#Converting the predictor attributes to factors
data.df$Personal.Loan <- as.factor(data.df$Personal.Loan)
data.df$Online <- as.factor(data.df$Online)
data.df$CreditCard <- as.factor(data.df$CreditCard)

#checking for na values
test.na <- is.na.data.frame(data.df)

#Data Partition
set.seed(123)
data_part <- createDataPartition(data.df$Personal.Loan,p=.6, list=F)
Train <- data.df[data_part,]
Validate <- data.df[-data_part,]

#Data Normalization
norm_model <- preProcess(Train[,-c(10,13:14)],
                method=c("center","scale"))
Train_norm <- predict(norm_model,Train)
Validate_norm <- predict(norm_model,Validate)
```

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

```
table_1 <- ftable(Train_norm[,c(14,10,13)])
table_1
```

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

B. The probability of customer accepting loan and using credit card plus being an online banking user = 51/(51+467) = 0.0984

C. Creation of pivot tables for the training data where one will have Loan (rows) as a function of Online (columns) and the other will have Loan (rows) as a function of CC (columns)

```
melt_t1 <- melt(Train_norm,id=c("Personal.Loan"),variable="Online")
melt_t2 <- melt(Train_norm,id=c("Personal.Loan"), variable="CreditCard")

cast_t1 <- dcast(melt_t1, Personal.Loan~Online)
cast_t2 <- dcast(melt_t2, Personal.Loan~CreditCard)
```

D.Compute the following quantities [P(A | B) i.e. the probability of A given B]

```
ftable(Train_norm[,c(10,13)])
```

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

```
ftable(Train_norm[,c(10,14)])
```

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

```
ftable(Train_norm[,10])
```

```
##    0    1
##
##  2712  288
```

1. *P(CC = 1 | Loan = 1) = 84/(84+204) =* ***0.2916***
2. *P(Online= 1 | Loan= 1) = 176/(176+112) =* ***0.6111***
3. *P(Loan = 1) = 288/(288+2712) =* ***0.096***
4. *P(CC= 1 | Loan= 0) = 777/(777+1935) =* ***0.2865***
5. *P(Online=1 |Loan=0) = 1611/(1611+1101) =* ***0.5940***
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)

(0.2916 x 0.6111 x 0.096) / (0.2916 x 0.6111 x 0.096) + (0.2865 x 0.5940 x 0.904) = ***0.1000***

F. By comparing the value obtained above by using the Naive Bayes probability i.e. 0.1000 to the value obtained in step b i.e. 0.0984 we get to see that both the values are almost similar, but Naive Bayes has a bit higher probability when compared to that with the direct calculation.

G. Run the Naive Bayes Model

```
naive <- naiveBayes(Personal.Loan~Online+CreditCard,data=Train_norm)
naive
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##     0     1
## 0.904 0.096
##
## Conditional probabilities:
##    Online
## Y           0         1
##   0 0.4059735 0.5940265
##   1 0.3888889 0.6111111
##
##    CreditCard
## Y           0         1
##   0 0.7134956 0.2865044
##   1 0.7083333 0.2916667
```

The value obtained by running the Naive Bayes Model for the customer who is accepting the loan and using credit card plus being an online banking user is 0.1000 which is equivalent to the value obtained in E

Predicting the Naive Bayes model over the validation data and also looking at the AUC Value and ROC Curve

```
pred_labels <- predict(naive,Validate_norm,type = "raw")
head(pred_labels)
```

```
##              0          1
## [1,] 0.9082737 0.09172629
## [2,] 0.9021538 0.09784623
## [3,] 0.9061594 0.09384060
## [4,] 0.9082737 0.09172629
## [5,] 0.9082737 0.09172629
## [6,] 0.8999139 0.10008606
```

```
roc(Validate_norm$Online,pred_labels[,2])
```

```
##
## Call:
## roc.default(response = Validate_norm$Online, predictor = pred_labels[,     2])
##
## Data: pred_labels[, 2] in 803 controls (Validate_norm$Online 0) < 1197 cases (Validate_norm$Online 1)
## Area under the curve: 1
```

```
plot.roc(Validate_norm$Online,pred_labels[,2])
```