assignment_3

joshna katta

2022-10-18

```
#LOADING LIBRARY FUNCTIONS
library(caret)
```

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

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

```
library(class)
library(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(reshape)
```

```
##
## Attaching package: 'reshape'
```

```
## The following object is masked from 'package:dplyr':
##
##     rename
```

```
## The following object is masked from 'package:class':
##
##     condense
```

```
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

```
## The following objects are masked from 'package:reshape':
##
##     colsplit, melt, recast
```

```
#READING CSV FILE
data1<-read.csv("C:/Users/sudhakar/Downloads/UniversalBank (1).csv",header=TRUE)
head(data1)
```

```
##   ID Age Experience Income ZIP.Code Family CCAvg Education Mortgage
## 1  1  25          1     49    91107      4   1.6         1        0
## 2  2  45         19     34    90089      3   1.5         1        0
## 3  3  39         15     11    94720      1   1.0         1        0
## 4  4  35          9    100    94112      1   2.7         2        0
## 5  5  35          8     45    91330      4   1.0         2        0
## 6  6  37         13     29    92121      4   0.4         2      155
##   Personal.Loan Securities.Account CD.Account Online CreditCard
## 1             0                  1          0      0          0
## 2             0                  1          0      0          0
## 3             0                  0          0      0          0
## 4             0                  0          0      0          0
## 5             0                  0          0      0          1
## 6             0                  0          0      1          0
```

```
#CONVERTING TO FACTORS
data1$Personal.Loan <- as.factor(data1$Personal.Loan)
data1$Online <- as.factor(data1$Online)
data1$CreditCard <- as.factor(data1$CreditCard)
is.factor(data1$Personal.Loan)
```

```
## [1] TRUE
```

```
is.factor(data1$Online)
```

```
## [1] TRUE
```

```
is.factor(data1$CreditCard)
```

```
## [1] TRUE
```

```
#PARTITIONING DATA TO 60:40 RATIO
set.seed(350)
data_partition<-createDataPartition(data1$Personal.Loan,p=.6,list=FALSE,times=1)
train<-data1[data_partition,]
valid<-data1[-data_partition,]
head(train)
```

```
##    ID Age Experience Income ZIP.Code Family CCAvg Education Mortgage
## 4   4  35          9    100    94112      1   2.7         2        0
## 8   8  50         24     22    93943      1   0.3         3        0
## 10 10  34          9    180    93023      1   8.9         3        0
## 11 11  65         39    105    94710      4   2.4         3        0
```

```
## 12 12  29          5     45    90277      3   0.1         2        0
## 13 13  48         23    114    93106      2   3.8         3        0
##    Personal.Loan Securities.Account CD.Account Online CreditCard
## 4              0                  0          0      0          0
## 8              0                  0          0      0          1
## 10             1                  0          0      0          0
## 11             0                  0          0      0          0
## 12             0                  0          0      1          0
## 13             0                  1          0      0          0
```

{A: Creating 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}

```
Melt_training = melt(train,id=c("CreditCard","Personal.Loan"),variable= "Online")
```

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

```
cast_training=dcast(Melt_training,CreditCard+Personal.Loan~Online)
```

```
## Aggregation function missing: defaulting to length
```

```
set.seed(15)
cast_training <-cast_training[c(1,2,14)]

cast_training
```

```
##   CreditCard Personal.Loan Online
## 1          0             0   1908
## 2          0             1    204
## 3          1             0    804
## 4          1             1     84
```

B: The probability of loan acceptance conditional on having a bank credit card

#(Loan=1,CC = 1,Online=1)=(53 /(468+53))=0.1017274

C:Creating separate pivot tables for the training data with Loan (rows) as a function of Online (columns)

```
Melt_training1 <- melt(train,id=c("Personal.Loan"),variable = "Online")
```

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

```
cast_training1  <- dcast(Melt_training1,Personal.Loan~Online)
```

```
## Aggregation function missing: defaulting to length
```

```
cast_training1 <-cast_training1[c(1,13)]

cast_training1
```

```
##   Personal.Loan Online
## 1             0   2712
## 2             1    288
```

```
set.seed(15)

melt_data2 <- melt(train,id=c("CreditCard"),variable = "Online")
```

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

```
cast_training2 <- dcast(melt_data2,CreditCard~Online)
```

```
## Aggregation function missing: defaulting to length
```

```
cast_training2 <-cast_training2[c(1,14)]

cast_training2
```

```
##   CreditCard Online
## 1          0   2112
## 2          1    888
```

```
set.seed(15)
Data1 <- train[c(13,10,14)]

table(Data1[,c(3,2)])
```

```
##           Personal.Loan
## CreditCard    0    1
##          0 1908  204
##          1  804   84
```

```
table(Data1[,c(1,2)])
```

```
##       Personal.Loan
## Online    0    1
##      0 1119  108
##      1 1593  180
```

```
table(Data1[,c(2)])
```

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

```
set.seed(15)

naivebayes = naiveBayes(Personal.Loan~.,data=Data1)

naivebayes
```

```
##
## 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 0.4126106 0.5873894
##   1 0.3750000 0.6250000
##
##    CreditCard
## Y           0         1
##   0 0.7035398 0.2964602
##   1 0.7083333 0.2916667
```

```
set.seed(15)
```