<u>Fundamentals of Machine Learning-Assignment-3</u>

Naive Bayes- Personal Loan Prediction

Problem Statement

The file UniversalBank.csv contains data on 5000 customers of Universal Bank. The data include customer demographic information (age, income, etc.), the customer's relationship with the bank (mortgage, securities account, etc.), and the customer response to the last personal loan campaign (Personal Loan). Among these 5000 customers, only 480 (= 9.6%) accepted the personal loan that was offered to them in the earlier campaign. In this exercise, we focus on two predictors: Online (whether the customer is an active user of online banking services) and Credit Card (abbreviated CC below) (does the customer hold a credit card issued by the bank), and the outcome Personal Loan

Data Preparation

I loaded my data frame from UnivversalBank.csv onto R

The response (Dependent variable) is Personal Loan, and the predictors are Online users and Credit card holders

Steps:

A. I have converted select/required variables including Personal Loan, and the predictors are Online users and Credit card holders
B. I simultaneously loaded from the library the following packages to enable analize and produce output:
   library(class), library(caret), library(lattice), library(ggplot2), library (ISLR), library(pROC), and library(e1071)
C. I have set the seed to 15 to retain my dataset
D. I then Partitioned the data into training (60%) and validation (40%) sets.
   I have this output:

Personal.Loan	Online	CreditCard
0:2712	0:1238	0:2128
1: 288	1:1762	1: 872

Personal.Loan	Online	CreditCard
0:1808	0: 778	0:1402
1: 192	1:1222	1: 598

QA. 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()*.

Action-Solution:

I created a pivot table using prop.table() that output the count tables and a proportion table of a cross table of Online users, Credit Card holders and Personal Loan acceptance. I have this output:

Pivot Table 1-Personal Loan and Online Users

Personal.Loan <fctr>	Online <fctr>	<int>
0	0	1827
1	0	189
0	1	2693
1	1	291

4 rows

Personal.Loan <fctr>	CreditCard <fctr>	<int>
0	0	3193
1	0	337
0	1	1327
1	1	143

4 rows

QB. 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? [This is the probability of loan acceptance (Loan = 1) conditional on having a bank credit card (CC = 1) and being an active user of online banking services (Online = 1)].

Action-Solution:

```
      No  Yes
No  1121 1591
Yes  117  171

      No  Yes
No  1921  791
Yes  207   81
```

The join probability that active Online users will accept Personal Loan is:
171/2000=0.085
The join probability that active Credit card users will accept Personal Loan is:
81/2000=0.0405
So, the join probability that an active Online users and active Credit card users will accept Personal Loan is:
0.085*0.0405=0.00344

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.

<u>Loan (rows) as a function of Online (columns)</u>

```
prop.table(table(Assignment3_UpdatedNew.tr$Personal.Loan,Assignment3_UpdatedNew.tr$Online),margin = 1)

           No       Yes
No  0.4133481 0.5866519
Yes 0.4062500 0.5937500
```

<u>Loan (rows) as a function of CC</u>

```
prop.table(table(Assignment3_UpdatedNew.tr$Personal.Loan,Assignment3_UpdatedNew.tr$CreditCard),margin = 1)

           No       Yes
No  0.7083333 0.2916667
Yes 0.7187500 0.2812500
```

D. Compute the following quantities [P(A | B) means "the probability of A given B"]: i. P(CC = 1 | Loan = 1) (the proportion of credit card holders among the loan acceptors)

Solution:

P(CC = 1 | Loan = 1)

0.2812*0.144 = 0.0405

ii. P(Online = 1 | Loan = 1)

0.5866*0.0405 = 0.0237

iii. P(Loan = 1) (the proportion of loan acceptors)

171/2,000*81/2,000 = 0.00346

iv. P(CC = 1 | Loan = 0)

0.7083

v. P(Online = 1 | Loan = 0)

0.4133

vi. P(Loan = 0)

1,1121/2,000*1,921/2,000 = 0.5383

E. Use the quantities computed above to compute the naive Bayes probability P(Loan = 1 | CC = 1, Online = 1).

$$\text{Approx..} = \frac{P(CC\backslash P=1)\ P(Online\backslash P=1)/P(P=1)}{P(CC\backslash P=0)\ P(Online\backslash P=0)/P(P=0)}$$

= <u>0.0405*0.00237/0.000346</u>
0.0405*0.00237/0.000346 + 0.708*0.4133*/0.5383
Approx..= 0.3378

F. Compare this value with the one obtained from the pivot table in (B). Which is a more accurate estimate?

the join probability that an active Online users and active Credit card users will accept Personal Loan = 0.00344

Naïve Bayes probability P(Loan = 1 | CC = 1, Online = 1) = 0.3378

Naïve Bayes is a more accurate estimate

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).

```
Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
   No   Yes 
0.904 0.096 

Conditional probabilities:
     Online
Y            No       Yes
  No  0.4133481 0.5866519
  Yes 0.4062500 0.5937500

     CreditCard
Y            No       Yes
  No  0.7083333 0.2916667
  Yes 0.7187500 0.2812500
```

I now generated a confusion matrix which indicates:

a. Accuracy=0.904
95% CI= (0.8929, 0.9143)
P-Value=0.5157
Sensitivity=1.000
Specificity=0.000

```
Confusion Matrix and Statistics

          Reference
Prediction   No  Yes
       No  2712  288
       Yes    0    0

               Accuracy : 0.904
                 95% CI : (0.8929, 0.9143)
    No Information Rate : 0.904
    P-Value [Acc > NIR] : 0.5157

                  Kappa : 0

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 1.000
            Specificity : 0.000
         Pos Pred Value : 0.904
         Neg Pred Value :   NaN
             Prevalence : 0.904
         Detection Rate : 0.904
   Detection Prevalence : 1.000
      Balanced Accuracy : 0.500

       'Positive' Class : No
```

Finally, I plotted the ROC and produce the output as depicted below: