Logistic Regression-Classification Project Documentation

Problem Definition:

The objective of this project is to find the personal loan willingness of a customer by modelling the past campaign data. The data is of 5000 customers of a commercial bank, who are having mortgage loans and credit cards with the bank. Based on the features of the customer like age, income, previous loans and the response for personal loan in the previous campaign, we are able to build the model under logistic regression and random forest methods

Primary objectives:

1. Building a classification model to find weather the customer is interested in taking personal loan
2. Building a predictive model to predict the probability of a customer to take the personal loan

Data collection and understanding:

The data related to 5000 customers belongs to a commercial bank. They are having mortgage loans and credit card with the bank. In a campaign they are asked about the willingness of taking personal loan and the response was recorded for 5000 customers.

The column details of the data sheet

Attributes for Bank_Personal_Loan_Modelling - copy.csv.

ID	Customer ID
Age	Customer's age in completed years
Experience	#years of professional experience
Income	Annual income of the customer ($000)
ZIPCode	Home Address ZIP code.
Family	Family size of the customer
CCAvg	Avg. spending on credit cards per month ($000)
Education	Education Level. 1: Undergrad; 2: Graduate; 3: Advanced/Professional
Mortgage	Value of house mortgage if any. ($000)
Personal Loan	Did this customer accept the personal loan offered in the last campaign?
Securities Account	Does the customer have a securities account with the bank?
CD Account	Does the customer have a certificate of deposit (CD) account with the bank?
Online	Does the customer use internet banking facilities?
CreditCard	Does the customer use a credit card issued by UniversalBank?

The primary objective is to build a predictive model to predict personal loan willingness of a customer from the data set given. The variable description is as follows

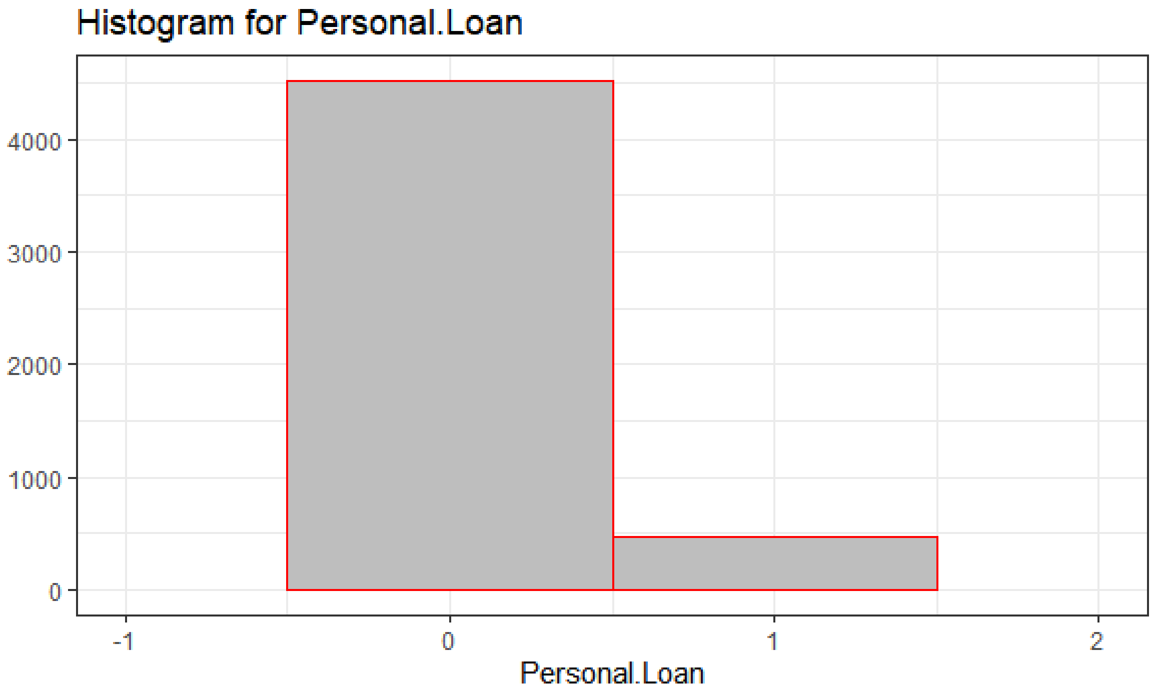
1. Categorical Variables(14) : ZIPCode, Family, Education, Securities Account, CD Account, Online, CreditCard,
2. Continuous variables(5): ID, Age, Experience, Income, CCAvg, Mortgage
3. Response Variables(2): Personal Loan

<u>Data cleaning and analysis :</u>

```
##################################

###  CASE STUDY ANALSYSIS

##################################

##### Importing data

data1=read.csv("C:/Users/HPP/Desktop/R programs/CLASSIFICATION PROJ/archive (5)/Bank_Personal_Loan_Modelling - Copy.csv")


N=ncol(data1)

N

#As the Zipcode is not necessary for model building, the Zipcode column is dropped

#### Data preprocessing according to your data and problem statement

######### Filtering relevant columns needed for analysis dropping zipcode column

########################################################

data_an=data1[,c(1,2,3,4,6,7,8,9,10,11,12,13,14)]

n=ncol(data_an)

n

head(data_an)

nn=nrow(data1)

nn

### Install and activate package 'ggplot2' needed for histogram and box plot

install.packages("ggplot2")

library(ggplot2)
```

Histogram of the response variable

```
qplot(data1$Personal.Loan,
      geom="histogram",
      binwidth=1,
      main="Histogram for Personal.Loan ",
      xlab="Personal.Loan",
      xlim=c(-1,2),
      fill=I("gray"),
      col=I("red"))+theme_bw()
```



From the histogram, we were able to see that nearly 4500 people were not interested in taking the personal loan

Obtaining descriptive statistics

```
install.packages("pastecs") # Install package 'pastecs' needed for obtaining descriptive stats
```

library(pastecs)

stat.desc(data1$Personal.Loan) # stat_desc(): function for displaying the descriptive statistics - mean, median, SD etc.

```
nbr.val     nbr.null      nbr.na         min          max        range
5.000000e+03 4.520000e+03 0.000000e+00 0.000000e+00 1.000000e+00 1.000000e+00
         sum       median        mean      SE.mean CI.mean.0.95          var
4.800000e+02 0.000000e+00 9.600000e-02 4.166566e-03 8.168297e-03 8.680136e-02
     std.dev     coef.var
2.946207e-01 3.068966e+00
```

The above statistics show the mean median and standard deviation of response variable

###perfrom shapiro test

shapiro.test(data1$Personal.Loan)

Shapiro-Wilk normality test

data: data1$Personal.Loan

W = 0.33425, p-value < 2.2e-16

Shapiro wilk test of response variable is as shown above

###perform t test

t.test(data1$Personal.Loan)

One Sample t-test

data: data1$Personal.Loan

t = 23.041, df = 4999, p-value < 2.2e-16

alternative hypothesis: true mean is not equal to 0

95 percent confidence interval:

 0.0878317 0.1041683

sample estimates:

mean of x

0.096

Model building

First part of model building is to split the data into test and train

Here the data is split into 9:1 ratio

```
### Creating training and test set

set.seed(123)

indx=sample(1:nn,0.9*nn)

traindata=data_an[indx,]

testdata=data_an[-indx,]
```

After splitting the data then now comes the modelling part by using 'glm' function

here response variable Personal.Loan is separated by ~ and the remaining predictor variables are written to the right side of ~. We are using training data for modelling and family is "binomial" for classification model

```
#### Fitting full logistic regression (LR) model with all features

fullmod=glm(Personal.Loan~ID+Age +Experience +Income +Family +CCAvg

        +Education +Mortgage +Securities.Account+CD.Account+Online

        +CreditCard,data=traindata,family="binomial")

summary(fullmod)
```

Output

```
Call:

glm(formula = Personal.Loan ~ ID + Age + Experience + Income +

    Family + CCAvg + Education + Mortgage + Securities.Account +

    CD.Account + Online + CreditCard, family = "binomial", data = traindata)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1452  -0.1972  -0.0770  -0.0296   3.6382

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)        -1.233e+01  1.756e+00  -7.020 2.22e-12 ***
ID                 -4.040e-05  5.438e-05  -0.743  0.45748
Age                -4.937e-02  6.520e-02  -0.757  0.44893
Experience          5.906e-02  6.481e-02   0.911  0.36213
Income              5.560e-02  2.818e-03  19.730  < 2e-16 ***
Family              6.909e-01  7.933e-02   8.710  < 2e-16 ***
CCAvg               1.086e-01  4.238e-02   2.563  0.01037 *
Education           1.761e+00  1.230e-01  14.325  < 2e-16 ***
Mortgage            6.575e-04  5.903e-04   1.114  0.26533
Securities.Account -9.461e-01  3.027e-01  -3.126  0.00177 **
CD.Account          3.807e+00  3.408e-01  11.171  < 2e-16 ***
Online             -6.795e-01  1.671e-01  -4.066 4.78e-05 ***
CreditCard         -1.071e+00  2.144e-01  -4.995 5.88e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2827.9  on 4499  degrees of freedom
Residual deviance: 1138.2  on 4487  degrees of freedom
AIC: 1164.2
```

Number of Fisher Scoring iterations: 8

The key takeaways from the output are as follows:

- The parameter estimates or the regression coefficients and their respective standard errors calculated. Using this, we can judge the relationship between that predictor variable and the response variable.
- The response variable is mostly varied with parameters like experience income and family according to the high coefficient values
- As per the p-value income, family, education, CD account, online, creditcard are the most significant features in predicting the response variable.
- The residual deviance is 1138, it is lower value than null deviance, it shown the model is perfectly fitted
- The AIC value is 1164.2. the less value of AIC shows that the measure of information lost is low in this model.

Selecting features for fitting reduced logistic regression model

Feature selection

By applying feature selection methods, we can directly obtain an optimum set of variables to reap the maximum benefit for the model.

The measure for the feature selection here is AIC

```
library(MASS)

step=stepAIC(fullmod)

Start:  AIC=1164.16

Personal.Loan ~ ID + Age + Experience + Income + Family + CCAvg +

    Education + Mortgage + Securities.Account + CD.Account +
```

```
    Online + CreditCard

                    Df Deviance    AIC
- ID                 1   1138.7 1162.7
- Age                1   1138.8 1162.8
- Experience         1   1139.0 1163.0
- Mortgage           1   1139.4 1163.4
<none>                   1138.2 1164.2
- CCAvg              1   1144.8 1168.8
- Securities.Account  1   1149.0 1173.0
- Online             1   1155.0 1179.0
- CreditCard         1   1166.2 1190.2
- Family             1   1223.1 1247.1
- CD.Account         1   1288.7 1312.7
- Education          1   1419.3 1443.3
- Income             1   1819.7 1843.7

Step:  AIC=1162.72
Personal.Loan ~ Age + Experience + Income + Family + CCAvg + 
    Education + Mortgage + Securities.Account + CD.Account + 
    Online + CreditCard

                    Df Deviance    AIC
- Age                1   1139.3 1161.3
```

```
- Experience         1   1139.6 1161.6
- Mortgage           1   1140.0 1162.0
<none>                   1138.7 1162.7
- CCAvg              1   1145.5 1167.5
- Securities.Account  1   1149.4 1171.4
- Online             1   1155.7 1177.7
- CreditCard         1   1166.9 1188.9
- Family             1   1223.7 1245.7
- CD.Account         1   1289.5 1311.5
- Education          1   1419.3 1441.3
- Income             1   1820.0 1842.0

Step:  AIC=1161.29
Personal.Loan ~ Experience + Income + Family + CCAvg + Education +
    Mortgage + Securities.Account + CD.Account + Online + CreditCard

                   Df Deviance    AIC
- Mortgage           1   1140.5 1160.5
<none>                   1139.3 1161.3
- Experience         1   1141.6 1161.6
- CCAvg              1   1146.0 1166.0
- Securities.Account  1   1149.8 1169.8
- Online             1   1156.2 1176.2
- CreditCard         1   1167.5 1187.5
```

```
- Family             1   1224.4 1244.4
- CD.Account         1   1290.7 1310.7
- Education          1   1428.2 1448.2
- Income             1   1832.0 1852.0

Step:  AIC=1160.52
Personal.Loan ~ Experience + Income + Family + CCAvg + Education +
    Securities.Account + CD.Account + Online + CreditCard

                     Df Deviance    AIC
<none>                    1140.5 1160.5
- Experience          1   1142.9 1160.9
- CCAvg               1   1146.8 1164.8
- Securities.Account  1   1151.2 1169.2
- Online              1   1157.4 1175.4
- CreditCard          1   1169.0 1187.0
- Family              1   1226.4 1244.4
- CD.Account          1   1292.8 1310.8
- Education           1   1428.5 1446.5
- Income              1   1858.8 1876.8
```

From the above AIC values AIC=1160.5 was selected and the corresponding model is build again below

Model with best AIC is as follows

mod2=glm(Personal.Loan ~ Experience +Income +Family +CCAvg

```
    +Education +Securities.Account+CD.Account+Online
    +CreditCard,data=traindata,family="binomial")
summary(mod2)
Call:
glm(formula = Personal.Loan ~ Experience + Income + Family +
  CCAvg + Education + Securities.Account + CD.Account + Online +
  CreditCard, family = "binomial", data = traindata)

Deviance Residuals:
  Min       1Q   Median      3Q      Max
-3.1821  -0.1961  -0.0766  -0.0296   3.6922

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)       -13.638406   0.637288 -21.401  < 2e-16 ***
Experience          0.010335   0.006764   1.528  0.12652
Income              0.056024   0.002797  20.028  < 2e-16 ***
Family              0.695240   0.079424   8.754  < 2e-16 ***
CCAvg               0.105463   0.042180   2.500  0.01241 *
Education           1.734524   0.120517  14.392  < 2e-16 ***
Securities.Account  -0.938868   0.302235  -3.106  0.00189 **
CD.Account          3.826309   0.340829  11.226  < 2e-16 ***
Online             -0.678854   0.166697  -4.072 4.65e-05 ***
CreditCard         -1.078442   0.214251  -5.034 4.81e-07 ***
```

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2827.9 on 4499 degrees of freedom

Residual deviance: 1140.5 on 4490 degrees of freedom

AIC: 1160.5

Number of Fisher Scoring iterations: 8

In the above model the securities account, credit card and online facility has negative impact on personal loan willing ness. i.e the customer having credit card or security account is less interested in taking personal loan.

Moreover a person having CD account has more chances of taking personal loan

The above model is more fitted model where the most predictor variables are significant. We see the AIC value is much decreased with the optimum predictor variables

<u>Validating the model</u>

By using the test data the model is evaluated for predicting the probability of personal loan willingness.

In test data we are selecting only optimum predictor variables which are selected by step AIC model

In Pred_prob we are predicting the probability of a customer who is willing to take personal loan

predicting success probabilities using the LR model

head(testdata)

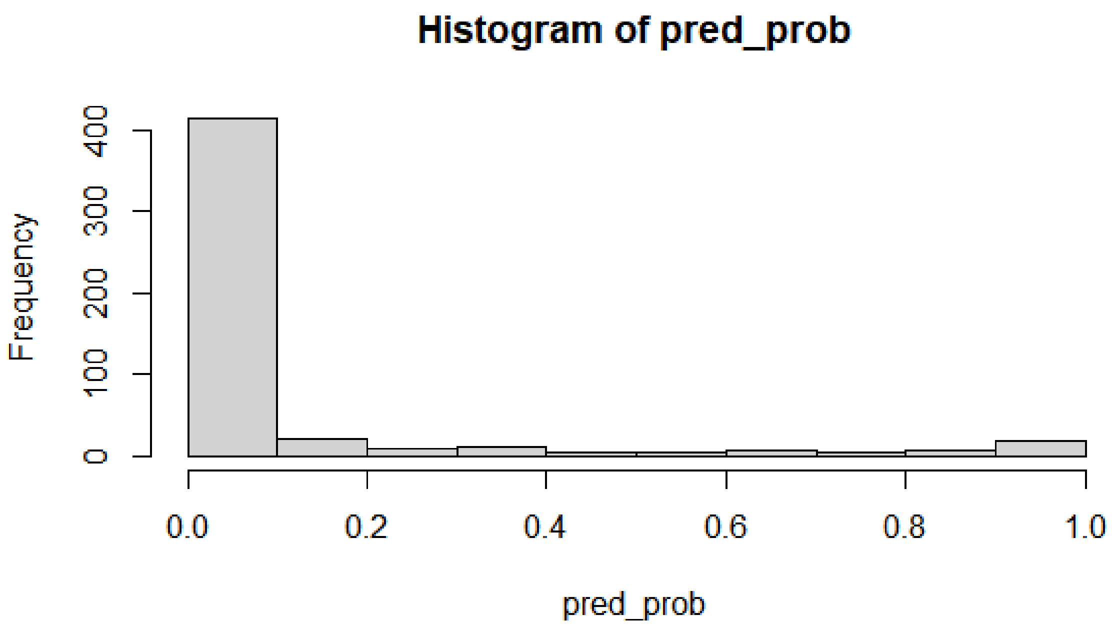
testdata_new=testdata[,c(3,4,5,6,7,10,11,12,13)]

pred_prob=predict(mod2,testdata_new,type="response")

hist(pred_prob)

by taking the values of pred_prob and plotting in a histogram look lies below.

As per the histogram we can observe that more than 0.1 probability the people are not interested in personal loan.



Now we will predict any random customer willingness to personal loan by using the above model.

In sampletest dataframe we are feeding some random values for a customer and predicting the probability of taking personal loan

predicting success probability for an individual

sampletest=data.frame(t(c(10,175,2,8.5,2,1,0,1,1)))

colnames(sampletest)=c("Experience","Income","Family","CCAvg","Education"," Securities.Account","CD.Account","Online","CreditCard")

sampletest

predict(mod2,sampletest,type="response")

predict(mod2,sampletest,type="response")

1

0.3382626

The predicted probability of the customer is 0.33

To find the threshold value we should plot ROC curve by using the following command. Here we are considering test data into consideration and response variable in column 9.

legacy.axes=TRUE shows y axes as sensitivity and X axes as 1- specificity

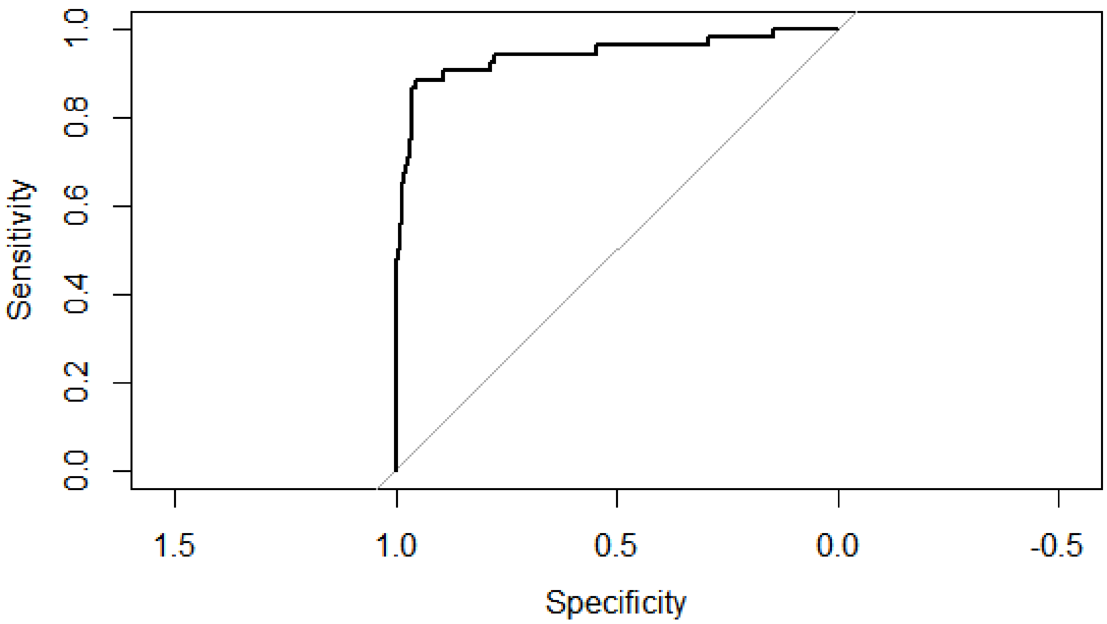
Plotting ROC

library(pROC)

roc1=roc(testdata[,9],pred_prob,plot=TRUE,legacy.axes=TRUE)

plot(roc1)

roc1$auc



The ROC curve shown above shows the plot was mostly tending towards left and the model is perfectly fitting.

Area under the curve: 0.9409

The area under the curve is obtained as 0.9409 shows the model is accurate in predicting the personal loan willingness of a customer.

To decide on the threshold we create a data frame with sensitivity ,specificity and thresholds from ROC

Using ROC in deciding threshold

thres=data.frame(sen=roc1$sensitivities,
spec=roc1$specificities,thresholds=roc1$thresholds)

here we are limiting threshold vales with sensitivity and specificity values as 0.94 and 0.68 respectively

thres[thres$sen>0.94&thres$spec>0.68,]

thres

sen spec thresholds

309 0.9423077 0.6808036 0.01886896

310 0.9423077 0.6830357 0.01922951

311 0.9423077 0.6852679 0.01968033

312 0.9423077 0.6875000 0.01995178

313 0.9423077 0.6897321 0.02009702

314 0.9423077 0.6919643 0.02027403

315 0.9423077 0.6941964 0.02033141

316 0.9423077 0.6964286 0.02037594

317 0.9423077 0.6986607 0.02080466

318 0.9423077 0.7008929 0.02132295

The above are the threshold values as per the restricted sensitivity and specificity values . The optimum threshold from the above values is 0.02

For the sampletest customer we have predicted the probability as 0.33.

since 0.33>0.02 we can conclude that the customer is willing to take personal loan

by taking threshold value as 0.02 we now create the confusion matrix for the testdata by the below code

for the pred_prob values which are more than threshold value 0.02 aretreated as 1 otherwise 0

library(caret)

pred_Y=ifelse(pred_prob > 0.02,1,0)

pred_Y

```
confusionMatrix(as.factor(testdata[,9]), as.factor(pred_Y))
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 309 139
         1   3  49

               Accuracy : 0.716
                 95% CI : (0.6743, 0.7551)
    No Information Rate : 0.624
    P-Value [Acc > NIR] : 9.33e-06

                  Kappa : 0.2932

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9904
            Specificity : 0.2606
         Pos Pred Value : 0.6897
         Neg Pred Value : 0.9423
             Prevalence : 0.6240
         Detection Rate : 0.6180
   Detection Prevalence : 0.8960
      Balanced Accuracy : 0.6255
```

'Positive' Class : 0

The above is the confusion matrix for threshold value 0.02. confusion matrix has the parameters of predictor variable and pred_prob.

- **The accuracy of the model is 71% which is good value**
- **It is observed that 139 people who are really willing to take personal loan were wrongly classified as 0.**
- **The sensitivity of the model is 0.99 means the model is exactly precited the customer who is not interested in personal loan**
- **The specificity is 0.26 means the it has high chance of wrongly classify a customer who is willing to take personal loan.**

Now let us check build another model using random forest and check for the accuracy.

To use random forest model all the categorical values must be converted in to categorical format using as.factor

```
################################
## Random Forest
################################
library(randomForest)
###create train data###create train data
head(data_an)
data_an$Personal.Loan=as.factor(data_an$Personal.Loan)
data_an$Family=as.factor(data_an$Family)
```

data_an$Education=as.factor(data_an$Education)

data_an$Securities.Account=as.factor(data_an$Securities.Account)

data_an$CD.Account=as.factor(data_an$CD.Account)

data_an$Online=as.factor(data_an$Online)

data_an$CreditCard=as.factor(data_an$CreditCard)

after conversion of categorical values model is built using random forest and saved into modRF. Here we are using the whole dataset for 5000 customers.

The number of trees used are 500.

###RF model

modRF=randomForest(Personal.Loan~ ., data=data_an,ntree=500, mtry=6)

modRF

Call:

randomForest(formula = Personal.Loan ~ ., data = data_an, ntree = 500, mtry = 6)

Type of random forest: classification

Number of trees: 500

No. of variables tried at each split: 6

OOB estimate of error rate: 1.12%

Confusion matrix:

	0	1	class.error
0	4509	11	0.002433628
1	45	435	0.093750000

By observing the output of the summary we can see that 4509+435 customers are correctly calculated. That is the accuracy of 0.988.

Out of bag error is very low and it is 1.12% it shows the random forest is an accurate model than logistic regression

To test the model we are considering the teat data as below

```
###create test data

head(testdata)

nrow(testdata)

testdata$Personal.Loan=as.factor(testdata$Personal.Loan)

testdata$Family=as.factor(testdata$Family)

testdata$Education=as.factor(testdata$Education)

testdata$Securities.Account=as.factor(testdata$Securities.Account)

testdata$CD.Account=as.factor(testdata$CD.Account)

testdata$Online=as.factor(testdata$Online)

testdata$CreditCard=as.factor(testdata$CreditCard)


print(testdata[4,])

ID Age Experience Income Family CCAvg Education Mortgage Personal.Loan

 48  37         12        194        4          0.2         3              211             1

   Securities.Account CD.Account Online CreditCard

48                   1          1      1          1
```

We are considering the 4th row of the test data and the data is as shown above

By taking the random forest model we are predicting the personal loan data by below code for 4th row droppin response variable.

```
predict(modRF,testdata[4,-9],type="response")
```

48

1

Levels: 0 1

The prediction shows the personal loan value as 1 and as per the row data it is actually so the model works very well with the test data.

Conclusion:

- **We have considered data of 5000 customers to predict the personal loan willingness based on the previous campaign data**
- **We have built logistic regression model with classification to find out the customer is interested in taking personal loan or not and also to know the probability of customer saying yes to personal loan**
- **By building the fitted model with step AIC and threshold 0.02 we have obtained the accuracy of 71%. Here the specificity is very less and it is 26% which is not at all recommended here due to the model is wrongly classing the person who is willing to take the personal loan**
- **Whereas another model was built by randomforest method for which we got the accuracy as 98% and specificity as 97% which is desirable**
- **Both the models worked well with test data and random sample data.**

Data analytics