Churn Prediction

Le Thi Thanh Tam

9/1/2022

Objective

Case : Which customers are likely to churn ?

- **Customer Churn** occurs when customers leave/stop doing business with the company or service.
- The ability to predict when a customer is at a high risk of churning is valuable for every business with returning customers. Churn is defined as the number of customers cancelling within a time period divided by the number of active customers at the start of that period. In order to apply a modeling technique to predict churn, we need to understand the customer behavior and characteristics which signal the risk of customers churn.
- For this analytics, I will look into a bank customer data to predict whether the customer will leave the credit card services of the bank.
- I will use R for this project. The number of data is not too large so can use R directly. (R is one of the predominant languages in data science ecosystem and makes it simple to efficiently implement statistical techniques and thus it is excellent choice for machine learning tasks).
- How can we categorize our customers and take actions separately for each group?

```
# load data
churn <- read.csv("E:/ThanhTam_DA/Project/Prediction/Bank Churners/churn.csv")
churn$X = NULL # Drop the first index column
```

```
# Transformation type
churn$Attrition_Flag = as.factor(churn$Attrition_Flag)
churn$Gender = as.factor(churn$Gender)
churn$Marital_Status = as.factor(churn$Marital_Status)

# Convert income category into numberic
churn$Income = rep('', nrow(churn))
churn$Income[churn$Income_Category == 9] = 0
churn$Income[churn$Income_Category == 1] = 40000/2
churn$Income[churn$Income_Category == 2] = (40000+60000)/2
churn$Income[churn$Income_Category == 3] = (60000+80000)/2
churn$Income[churn$Income_Category == 4] = (80000+120000)/2
churn$Income[churn$Income_Category == 5] = 120000
is.factor(churn$Attrition_Flag)
```

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

```r
churn$Income = as.numeric(churn$Income)
is.numeric(churn$Income)
```

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

```r
churn$Income_Category = NULL
summary(churn)
```

```
##  Attrition_Flag  Customer_Age   Gender   Dependent_count Education_Level
##  0:8500         Min.   :26.00   0:5358   Min.   :0.000   Min.   : 0.00
##  1:1627         1st Qu.:41.00   1:4769   1st Qu.:1.000   1st Qu.:12.00
##                 Median :46.00            Median :2.000   Median :16.00
##                 Mean   :46.33            Mean   :2.346   Mean   :25.57
##                 3rd Qu.:52.00            3rd Qu.:3.000   3rd Qu.:16.00
##                 Max.   :73.00            Max.   :5.000   Max.   :99.00
##  Marital_Status Card_Category      Total_Relationship_Count
##  0:3943         Length:10127       Min.   :1.000
##  1:4687         Class :character   1st Qu.:3.000
##  3: 748         Mode  :character   Median :4.000
##  9: 749                            Mean   :3.813
##                                    3rd Qu.:5.000
##                                    Max.   :6.000
##  Months_Inactive_12_mon Contacts_Count_12_mon  Credit_Limit
##  Min.   :0.000          Min.   :0.000         Min.   : 1438
##  1st Qu.:2.000          1st Qu.:2.000         1st Qu.: 2555
##  Median :2.000          Median :2.000         Median : 4549
##  Mean   :2.341          Mean   :2.455         Mean   : 8632
##  3rd Qu.:3.000          3rd Qu.:3.000         3rd Qu.:11068
##  Max.   :6.000          Max.   :6.000         Max.   :34516
##  Total_Revolving_Bal Total_Amt_Chng_Q4_Q1 Total_Trans_Amt Total_Ct_Chng_Q4_Q1
##  Min.   :   0        Min.   :0.0000       Min.   :  510   Min.   :0.0000
##  1st Qu.: 359        1st Qu.:0.6310       1st Qu.: 2156   1st Qu.:0.5820
##  Median :1276        Median :0.7360       Median : 3899   Median :0.7020
##  Mean   :1163        Mean   :0.7599       Mean   : 4404   Mean   :0.7122
##  3rd Qu.:1784        3rd Qu.:0.8590       3rd Qu.: 4741   3rd Qu.:0.8180
##  Max.   :2517        Max.   :3.3970       Max.   :18484   Max.   :3.7140
##  Avg_Utilization_Ratio     Income
##  Min.   :0.0000        Min.   :     0
##  1st Qu.:0.0230        1st Qu.: 20000
##  Median :0.1760        Median : 50000
##  Mean   :0.2749        Mean   : 49333
##  3rd Qu.:0.5030        3rd Qu.: 70000
##  Max.   :0.9990        Max.   :120000
```

```r
# create a new data frame not include unknown values _ Remove unknown‘
churn2 <- churn[!(churn$Marital_Status == 9|churn$Education_Level == 99),]
summary(churn2)
```

```
##  Attrition_Flag  Customer_Age   Gender   Dependent_count Education_Level
##  0:6716         Min.   :26.00   0:4222   Min.   :0.000   Min.   : 0.00
##  1:1257         1st Qu.:41.00   1:3751   1st Qu.:1.000   1st Qu.:12.00
##                 Median :46.00            Median :2.000   Median :15.00
```

```
##                 Mean   :46.37            Mean   :2.329   Mean   :12.62
##                 3rd Qu.:52.00            3rd Qu.:3.000   3rd Qu.:16.00
##                 Max.   :73.00            Max.   :5.000   Max.   :22.00
##  Marital_Status Card_Category      Total_Relationship_Count
##  0:3322         Length:7973        Min.   :1.000
##  1:3999         Class :character   1st Qu.:3.000
##  3: 652         Mode  :character   Median :4.000
##  9:   0                            Mean   :3.821
##                                    3rd Qu.:5.000
##                                    Max.   :6.000
##  Months_Inactive_12_mon Contacts_Count_12_mon  Credit_Limit
##  Min.   :0.000          Min.   :0.000         Min.   : 1438
##  1st Qu.:2.000          1st Qu.:2.000         1st Qu.: 2550
##  Median :2.000          Median :2.000         Median : 4522
##  Mean   :2.346          Mean   :2.457         Mean   : 8589
##  3rd Qu.:3.000          3rd Qu.:3.000         3rd Qu.:10973
##  Max.   :6.000          Max.   :6.000         Max.   :34516
##  Total_Revolving_Bal Total_Amt_Chng_Q4_Q1 Total_Trans_Amt Total_Ct_Chng_Q4_Q1
##  Min.   :   0        Min.   :0.0000       Min.   :  510   Min.   :0.0000
##  1st Qu.: 451        1st Qu.:0.6310       1st Qu.: 2132   1st Qu.:0.5830
##  Median :1281        Median :0.7360       Median : 3870   Median :0.7000
##  Mean   :1165        Mean   :0.7609       Mean   : 4377   Mean   :0.7121
##  3rd Qu.:1783        3rd Qu.:0.8590       3rd Qu.: 4739   3rd Qu.:0.8180
##  Max.   :2517        Max.   :3.3970       Max.   :17995   Max.   :3.7140
##  Avg_Utilization_Ratio     Income
##  Min.   :0.0000        Min.   :     0
##  1st Qu.:0.0250        1st Qu.: 20000
##  Median :0.1780        Median : 50000
##  Mean   :0.2764        Mean   : 49227
##  3rd Qu.:0.5040        3rd Qu.: 70000
##  Max.   :0.9990        Max.   :120000
```

```
attach(churn2)
```

1. Logistic Regression

```
glm.fits = glm(Attrition_Flag~., churn2, family = binomial)
summary(glm.fits)
```

```
##
## Call:
## glm(formula = Attrition_Flag ~ ., family = binomial, data = churn2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0930  -0.5003  -0.2914  -0.1377   3.7374
##
## Coefficients:
##                            Estimate Std. Error z value Pr(>|z|)
## (Intercept)               1.851e+00  3.584e-01   5.165 2.41e-07 ***
## Customer_Age             -1.628e-03  4.675e-03  -0.348 0.727730
```

```
## Gender1                  -5.884e-01  1.252e-01  -4.701 2.59e-06 ***
## Dependent_count           6.213e-02  2.948e-02   2.108 0.035050 *
## Education_Level           4.770e-03  6.009e-03   0.794 0.427306
## Marital_Status1          -1.705e-01  7.843e-02  -2.174 0.029686 *
## Marital_Status3          -3.246e-03  1.420e-01  -0.023 0.981767
## Card_CategoryGold         9.035e-01  3.710e-01   2.435 0.014878 *
## Card_CategoryPlatinum     1.392e+00  7.133e-01   1.952 0.050981 .
## Card_CategorySilver       5.331e-01  1.900e-01   2.806 0.005018 **
## Total_Relationship_Count -4.325e-01  2.686e-02 -16.105  < 2e-16 ***
## Months_Inactive_12_mon    4.229e-01  3.656e-02  11.568  < 2e-16 ***
## Contacts_Count_12_mon     4.829e-01  3.600e-02  13.413  < 2e-16 ***
## Credit_Limit             -1.636e-05  6.406e-06  -2.554 0.010641 *
## Total_Revolving_Bal      -7.183e-04  7.214e-05  -9.957  < 2e-16 ***
## Total_Amt_Chng_Q4_Q1     -5.758e-02  1.981e-01  -0.291 0.771301
## Total_Trans_Amt          -2.024e-04  1.887e-05 -10.728  < 2e-16 ***
## Total_Ct_Chng_Q4_Q1      -4.084e+00  2.249e-01 -18.162  < 2e-16 ***
## Avg_Utilization_Ratio    -5.491e-01  2.431e-01  -2.259 0.023899 *
## Income                    6.407e-06  1.742e-06   3.678 0.000235 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6948.7  on 7972  degrees of freedom
## Residual deviance: 4835.9  on 7953  degrees of freedom
## AIC: 4875.9
##
## Number of Fisher Scoring iterations: 6
```

Some findings from glm : * **Gender1** has small p-value means it is associated with our target. The negative coefficient for this predictor suggests **Male** is less likely to churn. * **Income** has a positive relationship with the churn likelihood. * Person made **The total transaction amount** large tended to not churned

```r
# first split data into training and test sets
set.seed(1)
train_set=sample(nrow(churn2), 0.8*nrow(churn2), replace = FALSE) # 80% dataset is the train set
train = churn2[train_set,]
test = churn2[-train_set,]

# fit a logistic regression model on train data set
glm.fits = glm(Attrition_Flag~.,data=train, family = binomial)

# predict probabilities of churn customers on test set
glm.probs=predict(glm.fits,test,type='response')

# compute the predctions and compare them to the actual churn customers
glm.pred=rep("0",nrow(test))
glm.pred[glm.probs>.5]="1"
table(glm.pred,test$Attrition_Flag)
```

Better assess the accuracy of the logistic regression model

```
##
## glm.pred    0    1
##        0 1334  138
##        1   25   98
```

```
(1316+104)/nrow(test)
```

```
## [1] 0.8902821
```

```
#--> test error rate equal 100-89 is 11% !!!!
```

- We recall that the logistic regression model, the small p-values associted with almost all of the predictors.
- In theory, consider the distribution of the predictors X (EDA part) is approximately normal in each of the classes, the logistic regression model may be unstable.
- Therefore we will consider to obtain better model for this project

RandomForest

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 4.1.3
```

```
## randomForest 4.7-1.1
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
set.seed(1)
rf.churn=randomForest(Attrition_Flag~., train, importance=TRUE)

yhat.rf= predict(rf.churn, newdata = test, type = "class")
table(yhat.rf,test$Attrition_Flag)
```

```
##
## yhat.rf    0    1
##       0 1342   54
##       1   17  182
```

```
(1329+177)/nrow(test)
```

```
## [1] 0.9442006
```

```
# mean(yhat.rf == test$Attrition_Flag)
#--> The classification accuracy of the model on the test set is 94.4% , test error rate is 5.6%
```