Bank Customer Churn Prediction

Keying Wu

2024-10

Table of Contents:

1. Ask to Clarify the Business Task

1.1 Business Task

The goal is to analyze the bank customer churn data to determine which factors influence why customers are leaving the service. By understanding these factors, we aim to implement targeted strategies to reduce churn and improve customer retention.

1.2 Key Objecives

a. Identify the most relevant factors that cause customers to churn or stay with the bank.

b. Predict which customers are more or less likely to churn.

2. Prepare the Data for Analysis

We will be using the bank-customer-churn dataset for our analysis. This dataset contains the customer data for an anonymous multinational bank. The dataset is public and free to use.

```r
# Import the libraries and the dataset

library(plyr)
library(corrplot)
library(ggplot2)
library(gridExtra)
library(ggthemes)
library(caret)
library(party)

destop_path = file.path(Sys.getenv("USERPROFILE"),"Desktop")
file_path = 'bank/Customer-Churn-Records.csv'
csv_path = paste(destop_path,file_path,sep = "/")
churn <- read.csv(csv_path)
```

```r
# Split the data

churn1 <- churn[, 1:6]
churn2 <- churn[, 7:12]
churn3 <- churn[, 13:18]

# Preview the data

knitr::kable(head(churn1))
```

RowNumber	CustomerId	Surname	CreditScore	Geography	Gender
1	15634602	Hargrave	619	France	Female
2	15647311	Hill	608	Spain	Female
3	15619304	Onio	502	France	Female
4	15701354	Boni	699	France	Female
5	15737888	Mitchell	850	Spain	Female
6	15574012	Chu	645	Spain	Male

```
knitr::kable(head(churn2))
```

Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember
42	2	0.00	1	1	1
41	1	83807.86	1	0	1
42	8	159660.80	3	1	0
39	1	0.00	2	0	0
43	2	125510.82	1	1	1
44	8	113755.78	2	1	0

```
knitr::kable(head(churn3))
```

EstimatedSalary	Exited	Complain	Satisfaction.Score	Card.Type	Point.Earned
101348.88	1	1	2	DIAMOND	464
112542.58	0	1	3	DIAMOND	456
113931.57	1	1	3	DIAMOND	377
93826.63	0	0	5	GOLD	350
79084.10	0	0	5	GOLD	425
149756.71	1	1	5	DIAMOND	484

```
colnames(churn)
```

```
##  [1] "RowNumber"          "CustomerId"         "Surname"
##  [4] "CreditScore"        "Geography"          "Gender"
##  [7] "Age"                "Tenure"             "Balance"
```

```
## [10] "NumOfProducts"     "HasCrCard"          "IsActiveMember"
## [13] "EstimatedSalary"   "Exited"             "Complain"
## [16] "Satisfaction.Score" "Card.Type"         "Point.Earned"
```

```r
str(churn)
```

```
## 'data.frame':    10000 obs. of  18 variables:
##  $ RowNumber         : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ CustomerId        : int  15634602 15647311 15619304 15701354 15737888 15574012 155
##  $ Surname           : chr  "Hargrave" "Hill" "Onio" "Boni" ...
##  $ CreditScore       : int  619 608 502 699 850 645 822 376 501 684 ...
##  $ Geography         : chr  "France" "Spain" "France" "France" ...
##  $ Gender            : chr  "Female" "Female" "Female" "Female" ...
##  $ Age               : int  42 41 42 39 43 44 50 29 44 27 ...
##  $ Tenure            : int  2 1 8 1 2 8 7 4 4 2 ...
##  $ Balance           : num  0 83808 159661 0 125511 ...
##  $ NumOfProducts     : int  1 1 3 2 1 2 2 4 2 1 ...
##  $ HasCrCard         : int  1 0 1 0 1 1 1 1 0 1 ...
##  $ IsActiveMember    : int  1 1 0 0 1 0 1 0 1 1 ...
##  $ EstimatedSalary   : num  101349 112543 113932 93827 79084 ...
##  $ Exited            : int  1 0 1 0 0 1 0 1 0 0 ...
##  $ Complain          : int  1 1 1 0 0 1 0 1 0 0 ...
##  $ Satisfaction.Score: int  2 3 3 5 5 5 2 2 3 3 ...
##  $ Card.Type         : chr  "DIAMOND" "DIAMOND" "DIAMOND" "GOLD" ...
##  $ Point.Earned      : int  464 456 377 350 425 484 206 282 251 342 ...
```

3. Process/Clean the Data

3.1 Check for missing values

```r
sapply(churn, function(x) sum(is.na(x)))
```

```
##          RowNumber          CustomerId             Surname         CreditScore
##                  0                   0                   0                   0
##          Geography              Gender                 Age              Tenure
##                  0                   0                   0                   0
##            Balance       NumOfProducts           HasCrCard      IsActiveMember
##                  0                   0                   0                   0
##    EstimatedSalary              Exited            Complain  Satisfaction.Score
##                  0                   0                   0                   0
##          Card.Type        Point.Earned
##                  0                   0
```

3.2 Organize the tenure into groups

```r
# Check for the minimum and maximum tenure

print(max(churn$Tenure))
```

```
## [1] 10
```

```r
print(min(churn$Tenure))
```

```
## [1] 0
```

```r
# Organize tenure into groups: 0-1 year, 2-3 years, 4-5 years, and >5 years

group_tenure <- function(Tenure)
{
```

```r
    if (Tenure >= 0 & Tenure <= 1){
        return('0-1 Years')
    }else if(Tenure >= 2 & Tenure <= 3){
        return('2-3 Years')
    }else if (Tenure >= 4 & Tenure <= 5){
        return('4-5 Years')
    }else if (Tenure > 5 ){
        return('> 5 Years')
 }
}

churn$Tenure.Group <- sapply(churn$Tenure,group_tenure)
churn$Tenure.Group <- as.factor(churn$Tenure.Group)
```

3.3 Change 0 and 1 inputs to "No" and "Yes"

```r
churn$HasCrCard <- as.factor(mapvalues(churn$HasCrCard,
                                       from=c("0","1"),
                                       to=c("No", "Yes")))

churn$IsActiveMember <- as.factor(mapvalues(churn$IsActiveMember,
                                       from=c("0","1"),
                                       to=c("No", "Yes")))

churn$Exited <- as.factor(mapvalues(churn$Exited,
                                       from=c("0","1"),
                                       to=c("No", "Yes")))

churn$Complain <- as.factor(mapvalues(churn$Complain,
                                       from=c("0","1"),
                                       to=c("No", "Yes")))
```

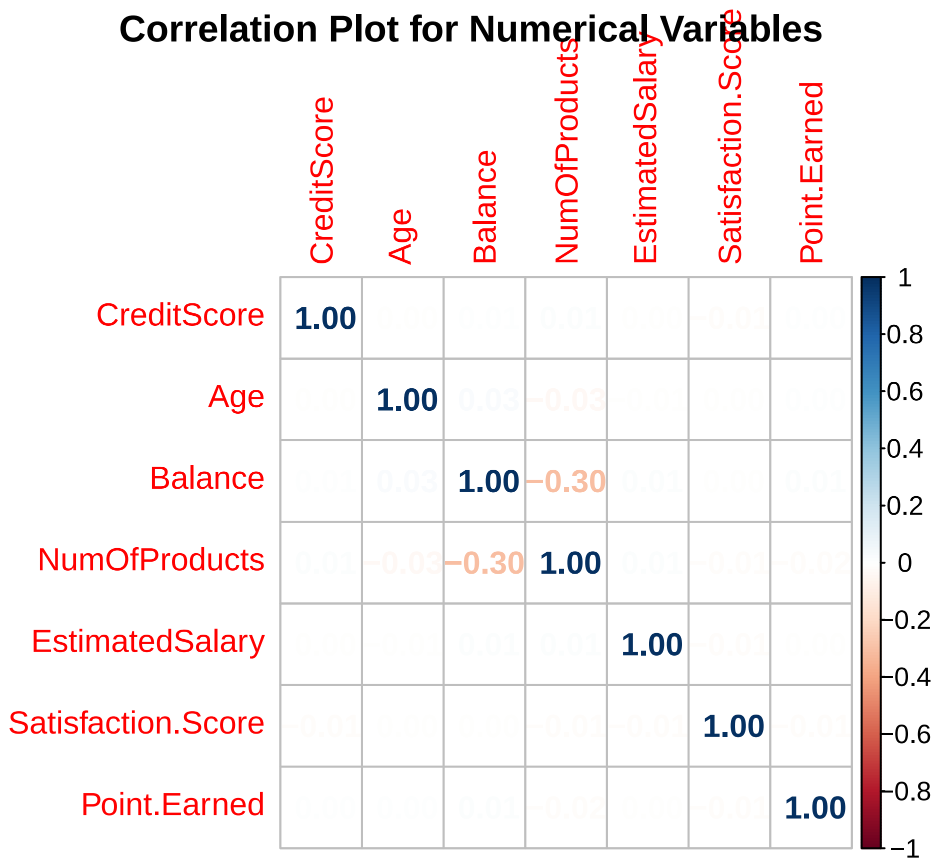
3.4 Remove unnecessary columns

```
churn$RowNumber <- NULL
churn$CustomerId <- NULL
churn$Surname <- NULL
churn$Tenure <- NULL
churn$tenure_group <- NULL
```

4. Analyze the Data

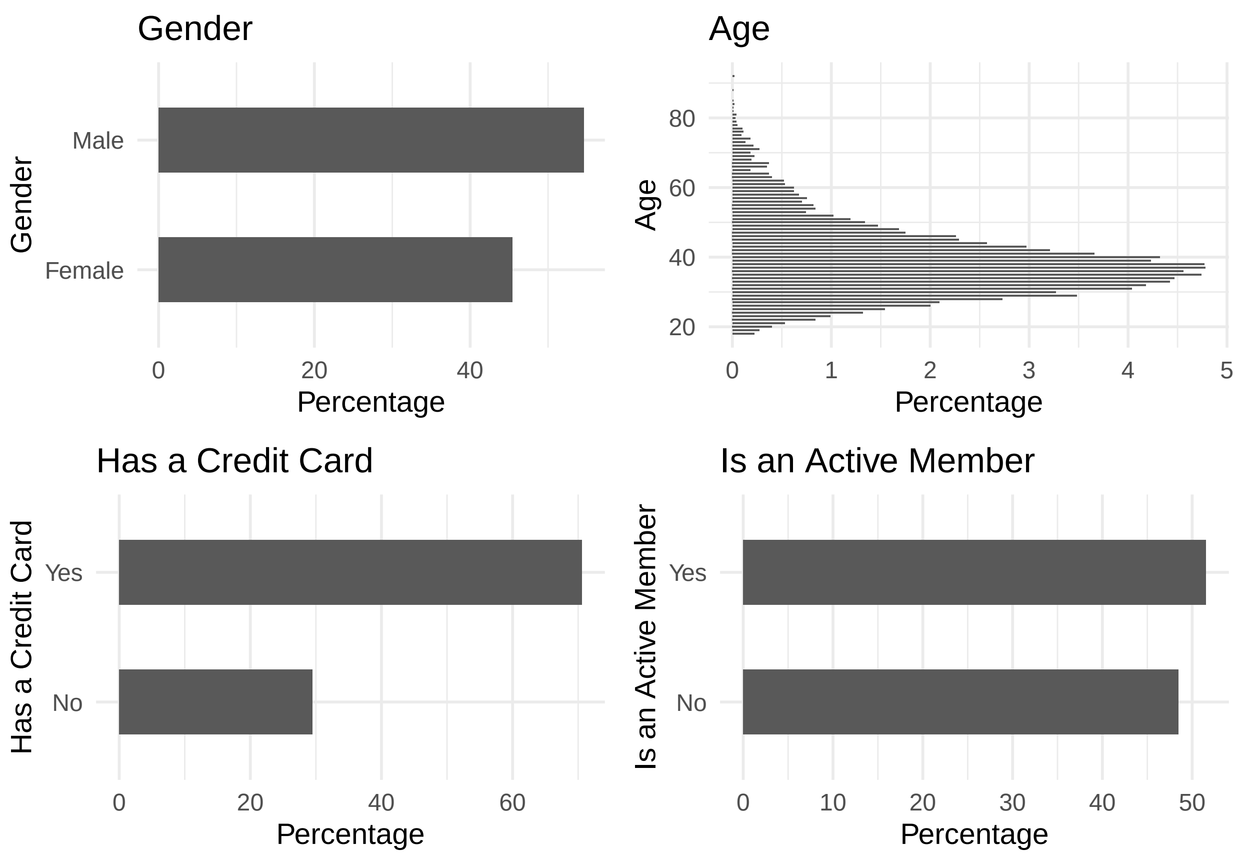
4.1 Perform exploratory data analysis

```
numeric.var <- sapply(churn, is.numeric)
corr.matrix <- cor(churn[,numeric.var])
corrplot(corr.matrix, main="\nCorrelation Plot for Numerical Variables",
         method="number")
```



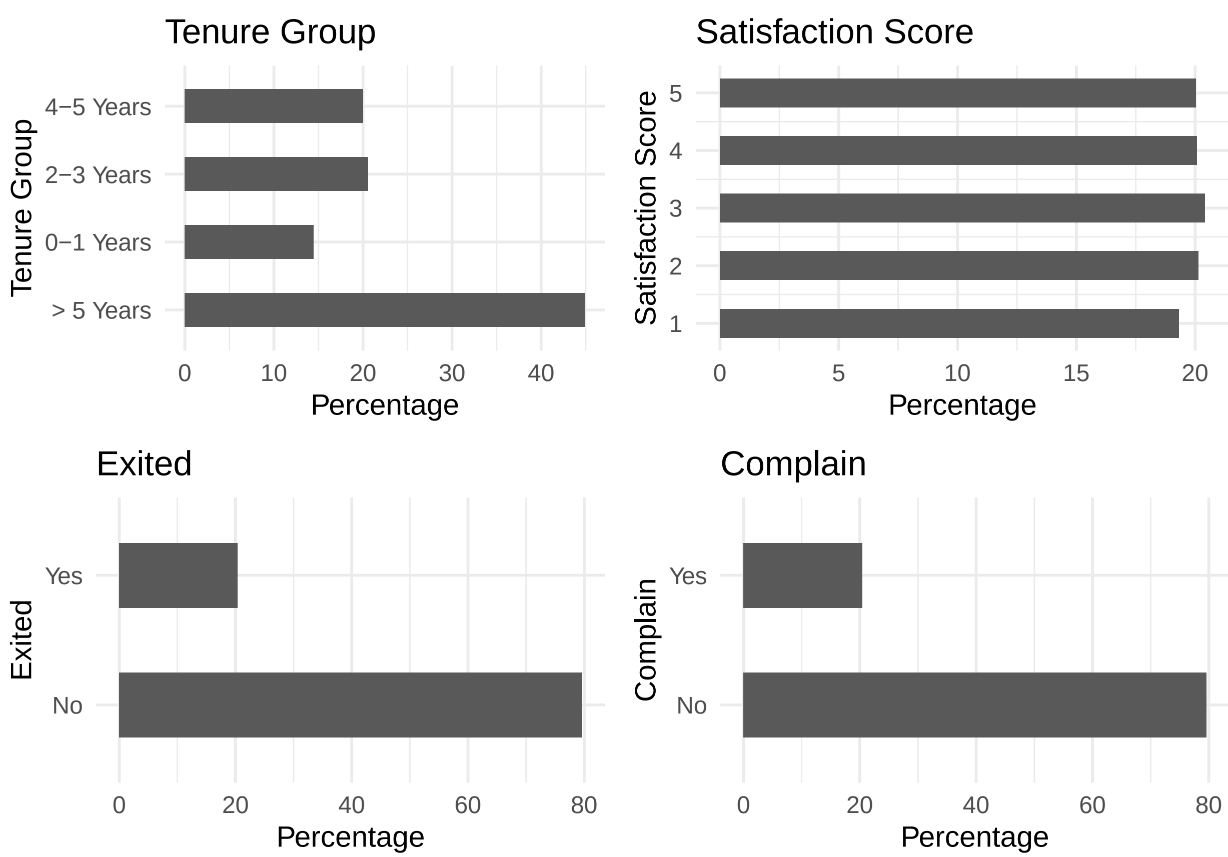

```
# Number of Products and Balance are correlated
# so we'll remove the Number of Products

churn$NumOfProducts <- NULL
```

```r
# Bar Plots

p1 <- ggplot(churn, aes(x=Gender)) + ggtitle("Gender") + xlab("Gender") +
  geom_bar(aes(y = 100*after_stat(count)/sum(after_stat(count))),
           width = 0.5) + ylab("Percentage") + coord_flip() + theme_minimal()

p2 <- ggplot(churn, aes(x=Age)) + ggtitle("Age") + xlab("Age") +
  geom_bar(aes(y = 100*after_stat(count)/sum(after_stat(count))),
           width = 0.5) + ylab("Percentage") + coord_flip() + theme_minimal()

p3 <- ggplot(churn, aes(x=HasCrCard)) +
  ggtitle("Has a Credit Card") + xlab("Has a Credit Card") +
  geom_bar(aes(y = 100*after_stat(count)/sum(after_stat(count))),
           width = 0.5) + ylab("Percentage") + coord_flip() + theme_minimal()

p4 <- ggplot(churn, aes(x=IsActiveMember)) +
  ggtitle("Is an Active Member") + xlab("Is an Active Member") +
  geom_bar(aes(y = 100*after_stat(count)/sum(after_stat(count))),
           width = 0.5) + ylab("Percentage") + coord_flip() + theme_minimal()

grid.arrange(p1, p2, p3, p4, ncol=2)
```




```
p1 <- ggplot(churn, aes(x=Tenure.Group)) +
  ggtitle("Tenure Group") + xlab("Tenure Group") +
  geom_bar(aes(y = 100*after_stat(count)/sum(after_stat(count))),
           width = 0.5) + ylab("Percentage") + coord_flip() + theme_minimal()

p2 <- ggplot(churn, aes(x=Satisfaction.Score)) +
  ggtitle("Satisfaction Score") + xlab("Satisfaction Score") +
  geom_bar(aes(y = 100*after_stat(count)/sum(after_stat(count))),
           width = 0.5) + ylab("Percentage") + coord_flip() + theme_minimal()

p3 <- ggplot(churn, aes(x=Exited)) + ggtitle("Exited") + xlab("Exited") +
  geom_bar(aes(y = 100*after_stat(count)/sum(after_stat(count))),
           width = 0.5) + ylab("Percentage") + coord_flip() + theme_minimal()

p4 <- ggplot(churn, aes(x=Complain)) + ggtitle("Complain") + xlab("Complain") +
```

```
  geom_bar(aes(y = 100*after_stat(count)/sum(after_stat(count))),
           width = 0.5) + ylab("Percentage") + coord_flip() + theme_minimal()

grid.arrange(p1, p2, p3, p4, ncol=2)
```



4.2 Logistic Regression

```
# Split the data into training and testing subsets

intrain<- createDataPartition(churn$Exited,p=0.7,list=FALSE)
set.seed(2024)
training<- churn[intrain,]
testing<- churn[-intrain,]

dim(training); dim(testing)
```

```
## [1] 7001   14
```

```
## [1] 2999   14
```

```r
# Fit the data to the logistic regression model

LogModel <- glm(Exited ~ .,family=binomial(link="logit"),data=training)
print(summary(LogModel))
```

```
##
## Call:
## glm(formula = Exited ~ ., family = binomial(link = "logit"),
##     data = training)
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -9.742e+00  3.761e+00  -2.591 0.009582 **
## CreditScore            1.165e-03  4.095e-03   0.284 0.776110
## GeographyGermany       1.246e-01  9.572e-01   0.130 0.896387
## GeographySpain         1.212e+00  1.038e+00   1.168 0.242894
## GenderMale            -1.733e+00  8.879e-01  -1.952 0.050926 .
## Age                    1.134e-01  3.343e-02   3.392 0.000693 ***
## Balance                6.795e-06  6.816e-06   0.997 0.318747
## HasCrCardYes          -5.828e-01  8.510e-01  -0.685 0.493466
## IsActiveMemberYes     -1.646e+00  8.149e-01  -2.020 0.043368 *
## EstimatedSalary       -1.381e-06  6.356e-06  -0.217 0.827940
## ComplainYes            1.688e+01  1.930e+00   8.746  < 2e-16 ***
## Satisfaction.Score    -2.241e-01  2.676e-01  -0.838 0.402266
## Card.TypeGOLD         -9.110e-01  1.077e+00  -0.846 0.397714
## Card.TypePLATINUM     -9.424e-01  1.056e+00  -0.893 0.372063
## Card.TypeSILVER       -3.068e-02  1.049e+00  -0.029 0.976668
## Point.Earned          -5.946e-03  2.396e-03  -2.482 0.013058 *
## Tenure.Group0-1 Years  2.100e+00  1.369e+00   1.535 0.124899
## Tenure.Group2-3 Years  1.945e+00  1.283e+00   1.516 0.129418
## Tenure.Group4-5 Years  3.530e-01  1.035e+00   0.341 0.733014
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7080.302  on 7000  degrees of freedom
## Residual deviance:   84.326  on 6982  degrees of freedom
## AIC: 122.33
##
## Number of Fisher Scoring iterations: 12
```

```r
# According to the results, Complaints, Age,
# and Points Earned are the most significant factors

# Next, we'll check the accuracy of the model

testing$Exited <- as.character(testing$Exited)
testing$Exited[testing$Exited=="No"] <- "0"
testing$Exited[testing$Exited=="Yes"] <- "1"
fitted.results <- predict(LogModel,newdata=testing,type='response')
fitted.results <- ifelse(fitted.results > 0.5,1,0)
misClasificError <- mean(fitted.results != testing$Exited)
print(paste('Logistic Regression Accuracy',1-misClasificError))
```

```
## [1] "Logistic Regression Accuracy 0.997999333111037"
```

```r
print("Confusion Matrix for Logistic Regression");
```

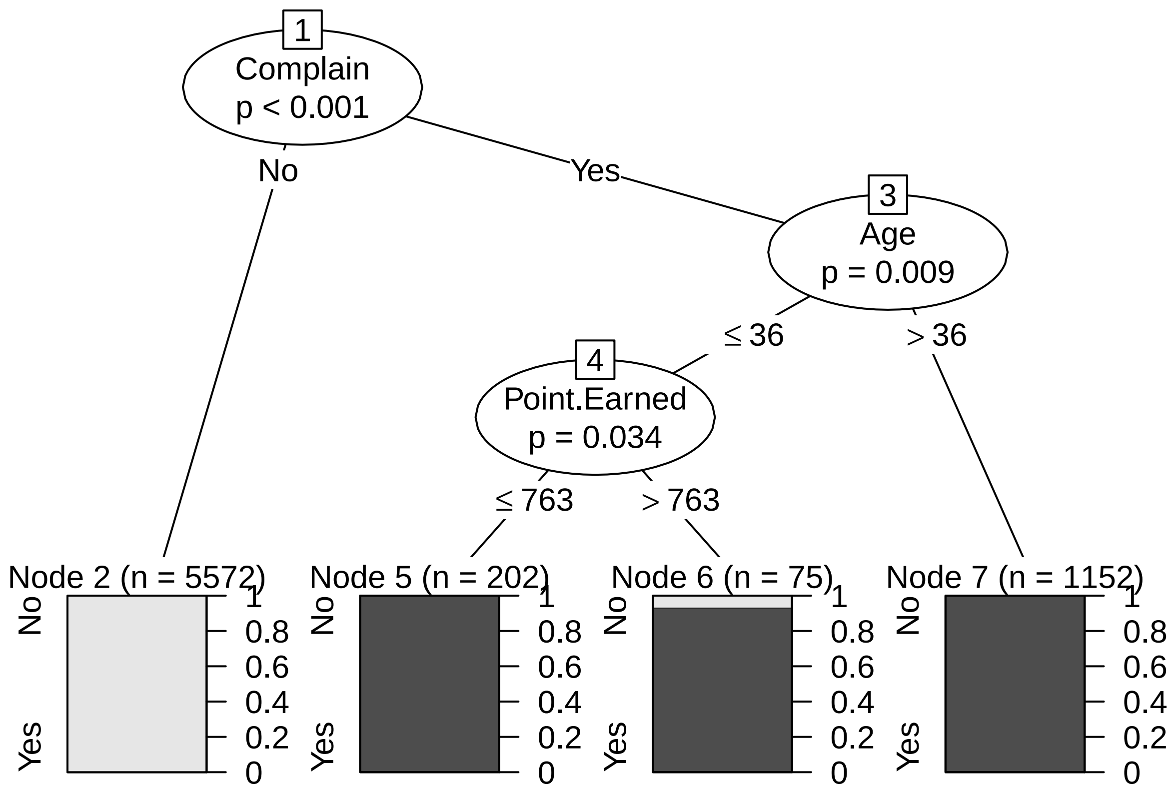
```
## [1] "Confusion Matrix for Logistic Regression"
```

```r
table(testing$Exited, fitted.results > 0.5)
```

```
##
##     FALSE TRUE
##   0  2383    5
##   1     1  610
```

4.3 Decision Tree

```
# Create a decision tree with the most relevant factors
# related to customer churn

tree <- ctree(Exited~Complain+Age+Point.Earned, training)
plot(tree)
```



5. Share the Results of the Analysis

5.1 Key Takeaways

a. Whether or not a customer submitted a complaint was the most significant factor in customer churn.

b. Other factors related to customer churn included the age of the customer and the number of points the customer had earned.

c. Interestingly, there does not appear to be a relationship between the customer satisfaction rating nor the length of tenure and customer churn.

d. If a customer submits a complaint, is over 36 years old, and has earned few points, they are more likely to churn. On the other hand, customers who have never issued a complaint are far less likely to churn.

5.2 Actionable Steps

a. Because there is little correlation between the customer satisfaction rating and customer churn, it would be beneficial to issue more extensive customer surveys to assess their needs, concerns, and general satisfaction.

b. Customers are far more likely to churn if they had previously submitted a complaint. That may imply that their complaints were not adequately resolved. To retain such customers, we should assess the most common complaints and make sure the customers are satisfied with the resolutions.