**CHURN PREDICTION /ANALYSIS**

**DONE USING R AND TABLEAU**

Prepared By:

*Deepak Narayanan Muralidharan*

**Churn Prediction /Analysis on the given set of dataset**

**Objective of the Project:**

To obtain a Logistic Regression Model of the Insurancedata which includes various attributes of customers mentioned in the dataset. This model will allow us to predict customers who will stick on with Allianz or in turn renew their existing service plans (Churn=0) and customers who will not (Churn=1). Therefore, this model will help us to take a decision to retain the customers (churned) who will look for some other Insurance companies.

**Approach taken for this Project:**

First of all to start with, there are around 21 variables in the given dataset. Before building a model we need to identify the most significant variables out of this 21. So, first we need to do some preprocessing of the data. As part of pre-processing we need to make sure that the data contains only numeric values and the data is fully available. Out of these 21 variables, there were 3 variables (Phone number, Area Code, etc.) which can be removed as they are not relevant for our prediction at this moment. Also, there were some data missing in certain rows. So those rows were omitted (using na.omit function in R). Then the significant variables can be found out from this data. The following steps where carried out:

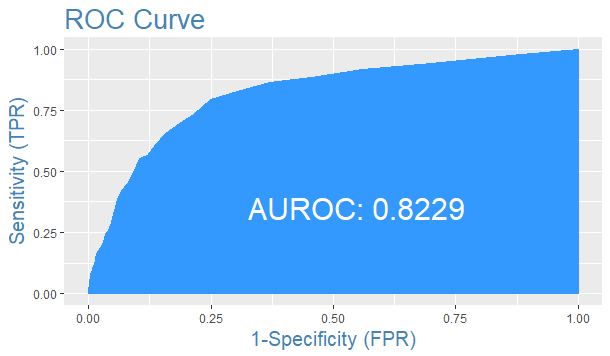
* To start with, the data was divided into 2 sets namely Training dataset and Test dataset with a proportion of 80:20.
* Created fit0 taking “Churn” as the depended value and rest as the independent values
* Similarly Fit1 was created
* The iteration continued and created Fit2 where we got the significant variables. This was done using step function.
* Hence finalized the model Fit 2.
* The next step is to predict the values for the training data using predict function. The predicted values are the probabilities. Using view () function we can see the probability values
* Converted these predicted values to “0” and “1”.
* The threshold needs to be set and the advisable one was 0.5
* Got the best accuracy rate with this threshold. Hence proceeded
* So based on the predicted values, compared it with the actual values using Table function.
* Post this, a confusion matrix was created and the accuracy was 86% for 0.5 threshold
* Plotted ROC Curve to determine the area under the curve.
* Plotted a KS Plot as well (just to understand percentage of responders).

Also, the aim was to visualize this in Tableau using “R Integration with Tableau” feature. The following steps were carried out:

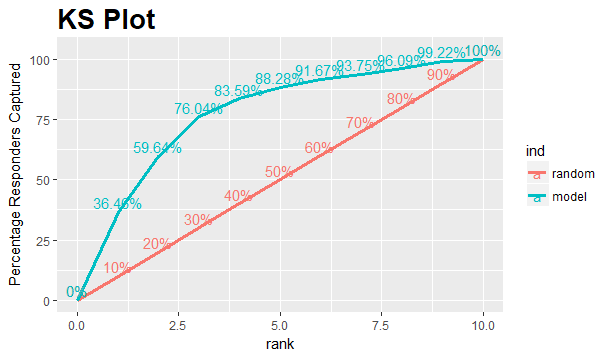
* Using the finalized model in R (fit2), created one named Final\_model (Prediction) in Tableau. This was created with the help of confusion matrix
* Created a graph using variables Churn (Actual Churn), Predicted Churn(Final\_model) ,Accuracy and Customer ID to identify the churn
* For the Odds sheet, created a filter with all the measure values & threshold parameter.
* A model created with R code with all variables (significant ones).

**Outcome & Observations:**

*ROC Chart:*

**

*Ks\_Plot:*

**

*Actual Vs Predicted Value Table:*

        Predicted

Actual    0    1

     0   2220   63

     1   307   77

**Conclusion:**

Based on our predicted Model Fit2, We can conclude that 307 customers will not churn and there is a change of 63 customers to get churned. This is based on an accuracy of 86.13%. Therefore the Telecom Company (Service Provider) should contact these 63 customers to retain them by providing any attractive offers.

R codes used for this Project:

|  |
| --- |
| install.packages("caret") |
|  | install.packages("pbkrtest") |
|  | library(caret) |
|  | library(ResourceSelection) |
|  | library(pbapply) |
|  | library(survey) |
|  | library(data.table) |
|  | library(InformationValue) |
|  |  |
|  |  |
|  | churndata$State=NULL |
|  | churndata$`Area Code`=NULL |
|  | churndata$Phone=NULL |
|  | na.omit(churndata) |
|  | summary(churndata) |
|  | set.seed(1234) |
|  | inTrain=createDataPartition(churndata$Churn,p=0.8,list=FALSE) |
|  | Training=churndata[inTrain,] |
|  | Testing=churndata[-inTrain,] |
|  | fit0=glm(Churn ~.,data = Training,family = binomial(link = "logit") ) |
|  | summary(fit0) |
|  | library(MASS) |
|  | install.packages("e1071") |
|  | library(e1071) |
|  | step= stepAIC(fit0,direction = "both") |
|  |  |
|  | fit1= glm(Churn ~ `Account Length` + `VMail Message` + `Day Mins` + `Eve Mins` + |
|  | `CustServ Calls` + `Int'l Plan` + `VMail Plan` + `Night Charge` + |
|  | `Intl Calls` + `Intl Charge`,data = Training,family = binomial(link = "logit") ) |
|  | summary(fit1) |
|  |  |
|  | fit2= update(fit1,.~.-`Account Length`,data = Training) |
|  | summary(fit2) |
|  |  |
|  | pred=predict(fit2,newdata = Training[,-1],type = "response" ) |
|  | View(pred) |
|  | pred1=ifelse(pred<0.5,0,1) |
|  | View(pred1) |
|  | table(Training$Churn,pred1,dnn = list('Actual','Predicted')) |
|  | confusionMatrix(table(Training$Churn,pred1,dnn = list('Actual','Predicted'))) |
|  |  |
|  |  |
|  | testpred=predict(fit2,newdata = Testing[,-1],type = "response" ) |
|  | View(testpred) |
|  | testpred1=ifelse(testpred<0.5,0,1) |
|  | View(testpred1) |
|  | confusionMatrix(table(Testing$Churn,testpred1,dnn = list('Actual','Predicted'))) |
|  |  |
|  |  |
|  |  |
|  | plotROC(actuals = Training$Churn,predictedScores = as.numeric(fitted(fit2))) |
|  | ks\_plot(actuals = Training$Churn,predictedScores = as.numeric(fitted(fit2))) |
|  |  |
|  | install.packages("Rserve") |
|  | library(Rserve) |
|  | Rserve() |

Thank-you