MRA_R_Script.R

Kalyan

Wed Sep 14 11:11:05 2016

```
## Set the working directory to read the input data file
setwd("D:/datafiles/")

## Read the input Employee data file
CellData = read.csv("Cellphone.csv", header = TRUE)
CellData$Churn = as.factor(CellData$Churn)

set.seed(7689)
library(caret)
## Warning: package 'caret' was built under R version 3.2.5
## Loading required package: lattice
## Loading required package: ggplot2
## Warning: package 'ggplot2' was built under R version 3.2.4
train <- createDataPartition(CellData$Churn,list=FALSE,times=1,p=0.7)

CellData.dev <- CellData[train,]
CellData.holdout <- CellData[-train,]

## Verify the data is evenly distributed across the partitions
prop.table(table(CellData$Churn))
## 
##         0         1 
## 0.8550855 0.1449145
prop.table(table(CellData.dev$Churn))
## 
##         0         1 
## 0.8547558 0.1452442
prop.table(table(CellData.holdout$Churn))
## 
##         0         1 
## 0.8558559 0.1441441
# Dim function to get number of rows and columns from the data frame
dim(CellData)
## [1] 3333   11
## Get the column names
names(CellData)
##  [1] "Churn"           "AccountWeeks"    "ContractRenewal"
##  [4] "DataPlan"        "DataUsage"       "CustServCalls"  
##  [7] "DayMins"         "DayCalls"        "MonthlyCharge"  
## [10] "OverageFee"      "RoamMins"
## Get Class detail for each column
sapply(CellData, class)
##           Churn    AccountWeeks ContractRenewal        DataPlan 
##        "factor"       "integer"       "integer"       "integer" 
##       DataUsage   CustServCalls         DayMins        DayCalls 
##       "numeric"       "integer"       "numeric"       "integer" 
##   MonthlyCharge      OverageFee        RoamMins 
##       "numeric"       "numeric"       "numeric"
## Structure of data set
str(CellData)
## 'data.frame':    3333 obs. of  11 variables:
##  $ Churn          : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ AccountWeeks   : int  128 107 137 84 75 118 121 147 117 141 ...
##  $ ContractRenewal: int  1 1 1 0 0 0 1 0 1 0 ...
##  $ DataPlan       : int  1 1 0 0 0 0 1 0 0 1 ...
##  $ DataUsage      : num  2.7 3.7 0 0 0 0 2.03 0 0.19 3.02 ...
##  $ CustServCalls  : int  1 1 0 2 3 0 3 0 1 0 ...
##  $ DayMins        : num  265 162 243 299 167 ...
##  $ DayCalls       : int  110 123 114 71 113 98 88 79 97 84 ...
##  $ MonthlyCharge  : num  89 82 52 57 41 57 87.3 36 63.9 93.2 ...
##  $ OverageFee     : num  9.87 9.78 6.06 3.1 7.42 ...
##  $ RoamMins       : num  10 13.7 12.2 6.6 10.1 6.3 7.5 7.1 8.7 11.2 ...
# Summarize the dataset
summary(CellData)
##  Churn     AccountWeeks   ContractRenewal     DataPlan     
##  0:2850   Min.   :  1.0   Min.   :0.0000   Min.   :0.0000  
##  1: 483   1st Qu.: 74.0   1st Qu.:1.0000   1st Qu.:0.0000  
##           Median :101.0   Median :1.0000   Median :0.0000  
##           Mean   :101.1   Mean   :0.9031   Mean   :0.2766  
##           3rd Qu.:127.0   3rd Qu.:1.0000   3rd Qu.:1.0000  
##           Max.   :243.0   Max.   :1.0000   Max.   :1.0000  
##    DataUsage      CustServCalls      DayMins         DayCalls    
##  Min.   :0.0000   Min.   :0.000   Min.   :  0.0   Min.   :  0.0  
##  1st Qu.:0.0000   1st Qu.:1.000   1st Qu.:143.7   1st Qu.: 87.0  
##  Median :0.0000   Median :1.000   Median :179.4   Median :101.0  
##  Mean   :0.8165   Mean   :1.563   Mean   :179.8   Mean   :100.4  
##  3rd Qu.:1.7800   3rd Qu.:2.000   3rd Qu.:216.4   3rd Qu.:114.0  
##  Max.   :5.4000   Max.   :9.000   Max.   :350.8   Max.   :165.0  
##  MonthlyCharge      OverageFee       RoamMins    
##  Min.   : 14.00   Min.   : 0.00   Min.   : 0.00  
##  1st Qu.: 45.00   1st Qu.: 8.33   1st Qu.: 8.50  
##  Median : 53.50   Median :10.07   Median :10.30  
##  Mean   : 56.31   Mean   :10.05   Mean   :10.24  
##  3rd Qu.: 66.20   3rd Qu.:11.77   3rd Qu.:12.10  
##  Max.   :111.30   Max.   :18.19   Max.   :20.00  
## Contract Renewal on Churn
chisq.test(CellData$ContractRenewal,CellData$Churn)
## 
##  Pearson's Chi-squared test with Yates' continuity correction
## 
## data:  CellData$ContractRenewal and CellData$Churn
## X-squared = 222.57, df = 1, p-value < 2.2e-16
```

INTERPRETATION:- ContractRenewal and Churn are highly significance

```
## DataPlan on Churn
chisq.test(CellData$DataPlan,CellData$Churn)
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  CellData$DataPlan and CellData$Churn
## X-squared = 34.132, df = 1, p-value = 5.151e-09
```

INTERPRETATION:- DataPlan and Churn are highly significance

```
## Correlation between AccountWeeks and Churn
cor(CellData$AccountWeeks, as.integer(CellData$Churn))
## [1] 0.01654074
```

INTERPRETATION:- AccountWeeks and Churn ARE NOT CORRELATED

```
## Correlation between Data Usage and Churn
cor(CellData$DataUsage, as.integer(CellData$Churn))
## [1] -0.08719451
```

INTERPRETATION:- DataUsage and Churn ARE NOT CORRELATED

```
## Correlation between Customer Service Call Count and Churn
cor(CellData$CustServCalls, as.integer(CellData$Churn))
## [1] 0.20875
```

INTERPRETATION:- DataUsage and Churn ARE LESS CORRELATED

```
## Correlation between Day Minutes and Churn
cor(CellData$DayMins, as.integer(CellData$Churn))
## [1] 0.2051508
```

INTERPRETATION:- DayMins and Churn ARE LESS CORRELATED

```
## Correlation between Day Calls and Churn
cor(CellData$DayCalls, as.integer(CellData$Churn))
## [1] 0.01845931
```

INTERPRETATION:- DayCalls and Churn ARE NOT CORRELATED

```
## Correlation between Monthly Charges and Churn
cor(CellData$MonthlyCharge, as.integer(CellData$Churn))
## [1] 0.07231271
```

INTERPRETATION:- MonthlyCharge and Churn ARE LESS CORRELATED

```
## Correlation between Overage Fee and Churn
cor(CellData$OverageFee, as.integer(CellData$Churn))
## [1] 0.09281243
```

INTERPRETATION:- OverageFee and Churn ARE LESS CORRELATED

```
## Correlation between Roaming Minutes and Churn
cor(CellData$RoamMins, as.integer(CellData$Churn))
## [1] 0.06823878
```

INTERPRETATION:- RoamMins and Churn ARE LESS CORRELATED

```
## Logistic Regression
CellData.glm<-
glm(Churn~CustServCalls+ContractRenewal+DataPlan+DataUsage+MonthlyCharge+Roam
Mins, data=CellData.dev, family="binomial"(link="logit"))

summary(CellData.glm)
##
## Call:
## glm(formula = Churn ~ CustServCalls + ContractRenewal + DataPlan +
##     DataUsage + MonthlyCharge + RoamMins, family = binomial(link =
"logit"),
##     data = CellData.dev)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0243  -0.5114  -0.3468  -0.2047   3.0317
##
```

```
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)    
## (Intercept)     -5.472501   0.508392 -10.764  < 2e-16 ***
## CustServCalls    0.486783   0.045982  10.586  < 2e-16 ***
## ContractRenewal -2.166641   0.173544 -12.485  < 2e-16 ***
## DataPlan        -1.680367   0.652044  -2.577  0.00996 ** 
## DataUsage       -0.505837   0.228818  -2.211  0.02706 *  
## MonthlyCharge    0.080704   0.007137  11.308  < 2e-16 ***
## RoamMins         0.071027   0.027223   2.609  0.00908 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 1934.3  on 2333  degrees of freedom
## Residual deviance: 1523.0  on 2327  degrees of freedom
## AIC: 1537
## 
## Number of Fisher Scoring iterations: 6
```

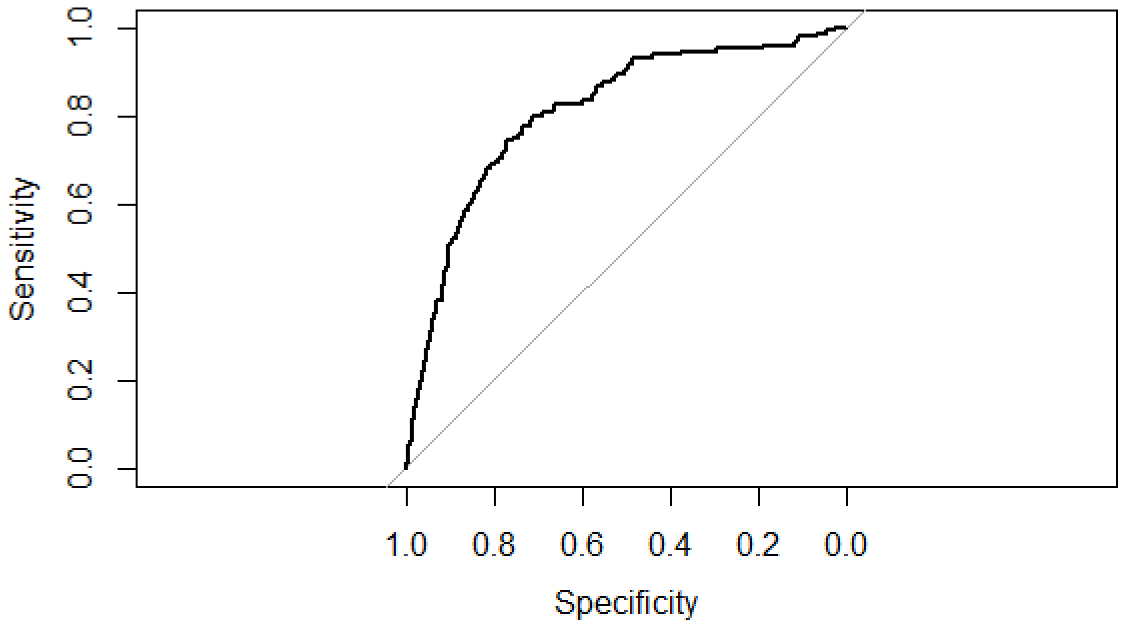
INTERPRETATION:- All the above variables are Significant/Highly Significant

```
confint(CellData.glm)
## Waiting for profiling to be done...
##                       2.5 %      97.5 %
## (Intercept)     -6.48542739 -4.49132189
## CustServCalls    0.39723463  0.57767037
## ContractRenewal -2.50862797 -1.82764401
## DataPlan        -2.98753756 -0.42780440
## DataUsage       -0.95352144 -0.05541682
## MonthlyCharge    0.06691287  0.09490650
## RoamMins         0.01805628  0.12483407
## odds ratios only

exp(coef(CellData.glm))
##     (Intercept)   CustServCalls ContractRenewal        DataPlan 
##     0.004200713     1.627073633     0.114561732     0.186305576 
##       DataUsage   MonthlyCharge        RoamMins 
##     0.603000357     1.084050101     1.073609817 

#developing prediction on the Testing dataset
CellData.holdout$PredictChurn<-predict.glm(CellData.glm,
newdata=CellData.holdout,type="response")

## evaluating the model
library(SDMTools)
## Warning: package 'SDMTools' was built under R version 3.2.5
## 
## Attaching package: 'SDMTools'
## The following objects are masked from 'package:caret':
## 
```

```
##     sensitivity, specificity

confusion.matrix(CellData.holdout$Churn, CellData.holdout$PredictChurn,
threshold = 0.5)
##     obs
## pred   0   1
##    0 820 115
##    1  35  29
```

INTERPRETATION:- This Confusion Matrix shows True Positive and False Positive

```
## attr(,"class")
## [1] "confusion.matrix"
#Predictive ability of the model using ROC curve
library(pROC)
## Warning: package 'pROC' was built under R version 3.2.5
## Type 'citation("pROC")' for a citation.
##
## Attaching package: 'pROC'
## The following object is masked from 'package:SDMTools':
##
```



```
##     auc
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
CellData.holdout.roc<-roc(Churn~PredictChurn, data=CellData.holdout)

attributes(CellData.holdout.roc)
## $names
##  [1] "percent"            "sensitivities"      "specificities"
##  [4] "thresholds"         "direction"          "cases"
##  [7] "controls"           "fun.sesp"           "auc"
## [10] "call"               "original.predictor" "original.response"
## [13] "predictor"          "response"           "levels"
##
## $class
## [1] "roc"
plot(roc(Churn~PredictChurn, data=CellData.holdout))
```

```
##
## Call:
## roc.formula(formula = Churn ~ PredictChurn, data = CellData.holdout)
##
## Data: PredictChurn in 855 controls (Churn 0) < 144 cases (Churn 1).
## Area under the curve: 0.8118

CellData.holdout.roc$auc
## Area under the curve: 0.8118
```

INTERPRETATION:- The AUC more than 80% shows the success prediction of actual versus Predicted Model

```
library(BCA)
## Warning: package 'BCA' was built under R version 3.2.5
CellData.holdout$Churn <- as.factor(CellData.holdout$Churn)
lift.chart("CellData.glm", data=CellData.holdout, targLevel=1, trueResp =
.28)
## [1] 0.1441441
```