Computer Based Assessment

Zeng Linghao

October 15,2020

## Part A: Data Exploration and Prepartion

1. **There are a few obvious data errors in the data. Conduct data exploration to identify the obvious data errors and correct the errors. Explain and justify your corrections. [You will use your corrected data in subsequent analysis.]**

* *Read the data*

setwd("~/Desktop/CBA/")  
churn<-read.csv(file="churn5.csv")

* *Do a quick summary of the data set*

summary(churn)

## Customer Account.Length International.Plan Voice.Mail.Plan   
## Min. : 1 Min. : 1.0 Length:2001 Length:2001   
## 1st Qu.: 501 1st Qu.: 73.0 Class :character Class :character   
## Median :1001 Median :101.0 Mode :character Mode :character   
## Mean :1001 Mean :100.9   
## 3rd Qu.:1501 3rd Qu.:127.0   
## Max. :2001 Max. :232.0   
##   
## Voice.Mail.Messages Day.Minutes Day.Calls Day.Charge   
## Min. : 0.000 Min. : 0.0 Min. : 0.0 Min. : 0.00   
## 1st Qu.: 0.000 1st Qu.:144.6 1st Qu.: 87.0 1st Qu.:24.58   
## Median : 0.000 Median :180.2 Median :101.0 Median :30.63   
## Mean : 7.696 Mean :180.6 Mean :100.3 Mean :30.70   
## 3rd Qu.:17.000 3rd Qu.:216.9 3rd Qu.:114.0 3rd Qu.:36.87   
## Max. :51.000 Max. :346.8 Max. :158.0 Max. :58.96   
## NA's :1   
## Evening.Minutes Evening.Calls Evening.Charge Night.Minutes   
## Min. : 0.0 Min. : 0.00 Min. : 0.00 Min. : 43.7   
## 1st Qu.:167.2 1st Qu.: 87.00 1st Qu.:14.21 1st Qu.:167.6   
## Median :202.2 Median :100.00 Median :17.19 Median :200.9   
## Mean :201.6 Mean : 99.84 Mean :17.13 Mean :201.2   
## 3rd Qu.:236.0 3rd Qu.:113.00 3rd Qu.:20.06 3rd Qu.:236.6   
## Max. :350.5 Max. :168.00 Max. :29.79 Max. :395.0   
##   
## Night.Calls Night.Charge International.Minutes International.Calls  
## Min. : 33.0 Min. : 1.970 Min. : 0.00 Min. : 0.000   
## 1st Qu.: 87.0 1st Qu.: 7.540 1st Qu.: 8.40 1st Qu.: 3.000   
## Median :100.0 Median : 9.040 Median :10.30 Median : 4.000   
## Mean :100.2 Mean : 9.055 Mean :10.22 Mean : 4.496   
## 3rd Qu.:113.0 3rd Qu.:10.650 3rd Qu.:12.00 3rd Qu.: 6.000   
## Max. :175.0 Max. :17.770 Max. :20.00 Max. :17.000   
##   
## International.Charge Customer.Service.Calls Churn   
## Min. : 0.00 Min. :0.000 Length:2001   
## 1st Qu.: 2.27 1st Qu.:1.000 Class :character   
## Median : 2.78 Median :1.000 Mode :character   
## Mean : 2.91 Mean :1.578   
## 3rd Qu.: 3.27 3rd Qu.:2.000   
## Max. :301.00 Max. :9.000   
##

* *Check all String data to see if it is binary*

#Internation Plan  
sum(churn$International.Plan=="Yes" | churn$International.Plan=="No")

## [1] 2001

#voice mail plan  
sum(churn$Voice.Mail.Plan=="Yes" | churn$Voice.Mail.Plan=="No")

## [1] 1999

#churn mail plan  
sum(churn$Churn=="Yes" | churn$Churn=="No")

## [1] 2001

* *From the above result, it appears that Voice.Mail.Plan contains missing data*

churn[(churn$Voice.Mail.Plan!="Yes" & churn$Voice.Mail.Plan!="No"),]

## Customer Account.Length International.Plan Voice.Mail.Plan  
## 1269 1269 58 No N  
## 2001 2001 108 Yes   
## Voice.Mail.Messages Day.Minutes Day.Calls Day.Charge Evening.Minutes  
## 1269 0 210.1 126 35.72 248.9  
## 2001 0 201.0 94 41.30 170.0  
## Evening.Calls Evening.Charge Night.Minutes Night.Calls Night.Charge  
## 1269 108 21.16 158.6 88 7.14  
## 2001 115 14.50 284.0 102 13.20  
## International.Minutes International.Calls International.Charge  
## 1269 14.4 2 3.89  
## 2001 11.1 4 2.90  
## Customer.Service.Calls Churn  
## 1269 4 No  
## 2001 1 Yes

* *For row 1269, it appears that “N” is used instead of “No” for Voice.Mail.Plan for row 2001, it appears that there is a missing values, we will aplply NA instead*

churn[1269,"Voice.Mail.Plan"]='No'  
churn[2001,"Voice.Mail.Plan"]=NA

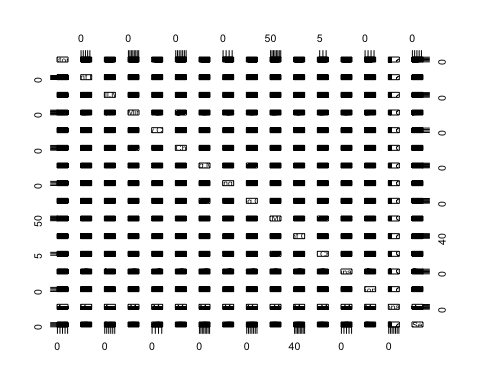
* *Let’s Turn our attention to the numerical data. First, let’s check if all the numerical attributes contain only numerical values*

churn.num<-churn[,c(1,2,5,6,7,8,9,10,11,12,13,14,15,16,17,18)]  
#check if all numeric coulumns contain only numeric numbers  
col.names<-names(churn.num)  
columns<-rep(0,16)  
for (i in 1:16){  
 columns[i]=is.numeric(churn.num[,col.names[i]])  
 print(col.names[i])  
 print(columns[i])  
}

## [1] "Customer"  
## [1] 1  
## [1] "Account.Length"  
## [1] 1  
## [1] "Voice.Mail.Messages"  
## [1] 1  
## [1] "Day.Minutes"  
## [1] 1  
## [1] "Day.Calls"  
## [1] 1  
## [1] "Day.Charge"  
## [1] 1  
## [1] "Evening.Minutes"  
## [1] 1  
## [1] "Evening.Calls"  
## [1] 1  
## [1] "Evening.Charge"  
## [1] 1  
## [1] "Night.Minutes"  
## [1] 1  
## [1] "Night.Calls"  
## [1] 1  
## [1] "Night.Charge"  
## [1] 1  
## [1] "International.Minutes"  
## [1] 1  
## [1] "International.Calls"  
## [1] 1  
## [1] "International.Charge"  
## [1] 1  
## [1] "Customer.Service.Calls"  
## [1] 1

* *In order to identify outliers, we can plot a scatter matrix and do a summary table*

plot(churn.num)

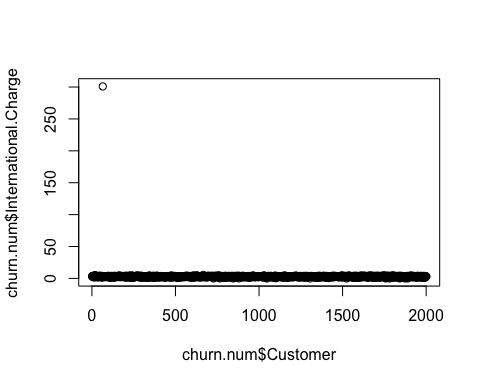


summary(churn.num)

## Customer Account.Length Voice.Mail.Messages Day.Minutes   
## Min. : 1 Min. : 1.0 Min. : 0.000 Min. : 0.0   
## 1st Qu.: 501 1st Qu.: 73.0 1st Qu.: 0.000 1st Qu.:144.6   
## Median :1001 Median :101.0 Median : 0.000 Median :180.2   
## Mean :1001 Mean :100.9 Mean : 7.696 Mean :180.6   
## 3rd Qu.:1501 3rd Qu.:127.0 3rd Qu.:17.000 3rd Qu.:216.9   
## Max. :2001 Max. :232.0 Max. :51.000 Max. :346.8   
##   
## Day.Calls Day.Charge Evening.Minutes Evening.Calls   
## Min. : 0.0 Min. : 0.00 Min. : 0.0 Min. : 0.00   
## 1st Qu.: 87.0 1st Qu.:24.58 1st Qu.:167.2 1st Qu.: 87.00   
## Median :101.0 Median :30.63 Median :202.2 Median :100.00   
## Mean :100.3 Mean :30.70 Mean :201.6 Mean : 99.84   
## 3rd Qu.:114.0 3rd Qu.:36.87 3rd Qu.:236.0 3rd Qu.:113.00   
## Max. :158.0 Max. :58.96 Max. :350.5 Max. :168.00   
## NA's :1   
## Evening.Charge Night.Minutes Night.Calls Night.Charge   
## Min. : 0.00 Min. : 43.7 Min. : 33.0 Min. : 1.970   
## 1st Qu.:14.21 1st Qu.:167.6 1st Qu.: 87.0 1st Qu.: 7.540   
## Median :17.19 Median :200.9 Median :100.0 Median : 9.040   
## Mean :17.13 Mean :201.2 Mean :100.2 Mean : 9.055   
## 3rd Qu.:20.06 3rd Qu.:236.6 3rd Qu.:113.0 3rd Qu.:10.650   
## Max. :29.79 Max. :395.0 Max. :175.0 Max. :17.770   
##   
## International.Minutes International.Calls International.Charge  
## Min. : 0.00 Min. : 0.000 Min. : 0.00   
## 1st Qu.: 8.40 1st Qu.: 3.000 1st Qu.: 2.27   
## Median :10.30 Median : 4.000 Median : 2.78   
## Mean :10.22 Mean : 4.496 Mean : 2.91   
## 3rd Qu.:12.00 3rd Qu.: 6.000 3rd Qu.: 3.27   
## Max. :20.00 Max. :17.000 Max. :301.00   
##   
## Customer.Service.Calls  
## Min. :0.000   
## 1st Qu.:1.000   
## Median :1.000   
## Mean :1.578   
## 3rd Qu.:2.000   
## Max. :9.000   
##

* *From the scatter matrix, there is a significant outlier in international charge*

plot(churn.num$International.Charge~churn.num$Customer)



summary(churn.num$International.Charge)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.00 2.27 2.78 2.91 3.27 301.00

* *We can apply the NA value to the outlier*

churn[churn[,"International.Charge"]==301,]

## Customer Account.Length International.Plan Voice.Mail.Plan  
## 65 65 41 No No  
## Voice.Mail.Messages Day.Minutes Day.Calls Day.Charge Evening.Minutes  
## 65 0 159.3 66 27.08 125.9  
## Evening.Calls Evening.Charge Night.Minutes Night.Calls Night.Charge  
## 65 75 10.7 261.9 76 11.79  
## International.Minutes International.Calls International.Charge  
## 65 11.1 5 301  
## Customer.Service.Calls Churn  
## 65 1 No

churn[65,"International.Charge"]=NA

* *Since customerID is not important, we omit that column*

churn<-churn[,-1]

* *Finally, factor the string attributes*

churn[,"Voice.Mail.Plan"]=as.factor(churn[,"Voice.Mail.Plan"])  
churn[,"International.Plan"]=as.factor(churn[,"International.Plan"])  
churn[,"Churn"]=as.factor(churn[,"Churn"])

1. **Create a summary table that displays the important information. Explain the findings.**

* *From the summary table, we are able to get a sense of out overall statistics. Several things come into mind, first of all, only about 1/10 of the total records are subscribers of International.Plan. Out of the total Calls.Churn, only about 1/10 of the Calls.Churn are Yes. Moreover, we could compare the summary table of those who are subscribers and those who are non-subscribers. The International Plan subscribers among the total subscribers are significant more than those non-subscribers. However, mean Voice.Mail.Messages of the subscribers of are only half of the mean Voice.Mail.Mesaages of the non-subscribers. Among Dauy.Minutes of the subscribers, the mean is lower than that of the non-subscribers.*

summary(churn)

## Account.Length International.Plan Voice.Mail.Plan Voice.Mail.Messages  
## Min. : 1.0 No :1809 No :1473 Min. : 0.000   
## 1st Qu.: 73.0 Yes: 192 Yes : 527 1st Qu.: 0.000   
## Median :101.0 NA's: 1 Median : 0.000   
## Mean :100.9 Mean : 7.696   
## 3rd Qu.:127.0 3rd Qu.:17.000   
## Max. :232.0 Max. :51.000   
##   
## Day.Minutes Day.Calls Day.Charge Evening.Minutes  
## Min. : 0.0 Min. : 0.0 Min. : 0.00 Min. : 0.0   
## 1st Qu.:144.6 1st Qu.: 87.0 1st Qu.:24.58 1st Qu.:167.2   
## Median :180.2 Median :101.0 Median :30.63 Median :202.2   
## Mean :180.6 Mean :100.3 Mean :30.70 Mean :201.6   
## 3rd Qu.:216.9 3rd Qu.:114.0 3rd Qu.:36.87 3rd Qu.:236.0   
## Max. :346.8 Max. :158.0 Max. :58.96 Max. :350.5   
## NA's :1   
## Evening.Calls Evening.Charge Night.Minutes Night.Calls   
## Min. : 0.00 Min. : 0.00 Min. : 43.7 Min. : 33.0   
## 1st Qu.: 87.00 1st Qu.:14.21 1st Qu.:167.6 1st Qu.: 87.0   
## Median :100.00 Median :17.19 Median :200.9 Median :100.0   
## Mean : 99.84 Mean :17.13 Mean :201.2 Mean :100.2   
## 3rd Qu.:113.00 3rd Qu.:20.06 3rd Qu.:236.6 3rd Qu.:113.0   
## Max. :168.00 Max. :29.79 Max. :395.0 Max. :175.0   
##   
## Night.Charge International.Minutes International.Calls  
## Min. : 1.970 Min. : 0.00 Min. : 0.000   
## 1st Qu.: 7.540 1st Qu.: 8.40 1st Qu.: 3.000   
## Median : 9.040 Median :10.30 Median : 4.000   
## Mean : 9.055 Mean :10.22 Mean : 4.496   
## 3rd Qu.:10.650 3rd Qu.:12.00 3rd Qu.: 6.000   
## Max. :17.770 Max. :20.00 Max. :17.000   
##   
## International.Charge Customer.Service.Calls Churn   
## Min. :0.000 Min. :0.000 No :1719   
## 1st Qu.:2.270 1st Qu.:1.000 Yes: 282   
## Median :2.780 Median :1.000   
## Mean :2.761 Mean :1.578   
## 3rd Qu.:3.248 3rd Qu.:2.000   
## Max. :5.400 Max. :9.000   
## NA's :1

summary(churn[churn[,"Churn"]=="Yes",])

## Account.Length International.Plan Voice.Mail.Plan Voice.Mail.Messages  
## Min. : 1.0 No :205 No :239 Min. : 0.000   
## 1st Qu.: 77.0 Yes: 77 Yes : 42 1st Qu.: 0.000   
## Median :104.5 NA's: 1 Median : 0.000   
## Mean :102.5 Mean : 4.649   
## 3rd Qu.:127.8 3rd Qu.: 0.000   
## Max. :225.0 Max. :45.000   
## Day.Minutes Day.Calls Day.Charge Evening.Minutes  
## Min. : 46.5 Min. : 42.0 Min. : 7.91 Min. : 75.3   
## 1st Qu.:155.0 1st Qu.: 89.0 1st Qu.:26.36 1st Qu.:176.8   
## Median :225.0 Median :104.0 Median :38.27 Median :210.6   
## Mean :211.9 Mean :102.2 Mean :36.06 Mean :212.0   
## 3rd Qu.:267.5 3rd Qu.:118.0 3rd Qu.:45.48 3rd Qu.:249.2   
## Max. :346.8 Max. :156.0 Max. :58.96 Max. :350.5   
## Evening.Calls Evening.Charge Night.Minutes Night.Calls   
## Min. : 48.0 Min. : 6.40 Min. : 47.4 Min. : 49.0   
## 1st Qu.: 88.0 1st Qu.:15.03 1st Qu.:169.6 1st Qu.: 84.0   
## Median :101.5 Median :17.89 Median :205.8 Median :100.5   
## Mean :100.4 Mean :18.02 Mean :204.8 Mean :100.2   
## 3rd Qu.:114.0 3rd Qu.:21.18 3rd Qu.:239.6 3rd Qu.:115.0   
## Max. :168.0 Max. :29.79 Max. :332.7 Max. :158.0   
## Night.Charge International.Minutes International.Calls  
## Min. : 2.130 Min. : 2.00 Min. : 1.000   
## 1st Qu.: 7.633 1st Qu.: 8.80 1st Qu.: 2.000   
## Median : 9.260 Median :10.50 Median : 4.000   
## Mean : 9.219 Mean :10.56 Mean : 4.174   
## 3rd Qu.:10.780 3rd Qu.:12.40 3rd Qu.: 5.000   
## Max. :14.970 Max. :20.00 Max. :15.000   
## International.Charge Customer.Service.Calls Churn   
## Min. :0.54 Min. :0.000 No : 0   
## 1st Qu.:2.38 1st Qu.:1.000 Yes:282   
## Median :2.84 Median :2.000   
## Mean :2.85 Mean :2.259   
## 3rd Qu.:3.35 3rd Qu.:4.000   
## Max. :5.40 Max. :9.000

summary(churn[churn[,"Churn"]=="No",])

## Account.Length International.Plan Voice.Mail.Plan Voice.Mail.Messages  
## Min. : 1.0 No :1604 No :1234 Min. : 0.000   
## 1st Qu.: 72.0 Yes: 115 Yes: 485 1st Qu.: 0.000   
## Median :101.0 Median : 0.000   
## Mean :100.6 Mean : 8.196   
## 3rd Qu.:127.0 3rd Qu.:20.000   
## Max. :232.0 Max. :51.000   
##   
## Day.Minutes Day.Calls Day.Charge Evening.Minutes  
## Min. : 0.0 Min. : 0.00 Min. : 0.00 Min. : 0.0   
## 1st Qu.:143.7 1st Qu.: 87.00 1st Qu.:24.42 1st Qu.:165.2   
## Median :177.2 Median :100.00 Median :30.12 Median :200.2   
## Mean :175.4 Mean : 99.98 Mean :29.82 Mean :199.9   
## 3rd Qu.:209.9 3rd Qu.:114.00 3rd Qu.:35.69 3rd Qu.:233.9   
## Max. :313.8 Max. :158.00 Max. :53.35 Max. :332.1   
## NA's :1   
## Evening.Calls Evening.Charge Night.Minutes Night.Calls   
## Min. : 0.00 Min. : 0.00 Min. : 43.7 Min. : 33.0   
## 1st Qu.: 86.50 1st Qu.:14.04 1st Qu.:167.3 1st Qu.: 87.0   
## Median :100.00 Median :17.02 Median :200.0 Median :100.0   
## Mean : 99.74 Mean :16.99 Mean :200.6 Mean :100.2   
## 3rd Qu.:113.00 3rd Qu.:19.89 3rd Qu.:236.2 3rd Qu.:113.0   
## Max. :164.00 Max. :28.23 Max. :395.0 Max. :175.0   
##   
## Night.Charge International.Minutes International.Calls  
## Min. : 1.970 Min. : 0.00 Min. : 0.000   
## 1st Qu.: 7.530 1st Qu.: 8.40 1st Qu.: 3.000   
## Median : 9.000 Median :10.20 Median : 4.000   
## Mean : 9.028 Mean :10.17 Mean : 4.549   
## 3rd Qu.:10.630 3rd Qu.:12.00 3rd Qu.: 6.000   
## Max. :17.770 Max. :18.90 Max. :17.000   
##   
## International.Charge Customer.Service.Calls Churn   
## Min. :0.000 Min. :0.000 No :1719   
## 1st Qu.:2.270 1st Qu.:1.000 Yes: 0   
## Median :2.750 Median :1.000   
## Mean :2.746 Mean :1.466   
## 3rd Qu.:3.240 3rd Qu.:2.000   
## Max. :5.100 Max. :8.000   
## NA's :1

## Part B: Models and Model Based Insight

1. **Execute CART and another model of your choice taught in this module. Which model perform better? Explain.**

* *Before Executing the model, we need to first do a train-test split*

train<-sample(1:nrow(churn),nrow(churn)\*0.8)  
test<--train

* *Use the previous generated train, and test row numbers to apply to the total data set*

churn.train<-churn[train,]  
churn.test<-churn[test,]

* *We need to balance the response variables*

sum(churn.train$Churn=="Yes")

## [1] 219

sum(churn.train$Churn=="No")

## [1] 1381

* *From the above code, we see the Yes and No are highly imbalanced. We need to balance the training set by duplicating the Yes so the ratio is 0.5*

churn.train.yes<-churn.train[churn.train[,"Churn"]=="Yes",]  
churn.train.no<-churn.train[churn.train[,"Churn"]=="No",]  
chosen<-sample(seq(1:nrow(churn.train.yes)),size = nrow(churn.train.no),replace = TRUE)  
churn.train.balanced<-rbind(churn.train.yes[chosen,],churn.train.no)

### RPART Model

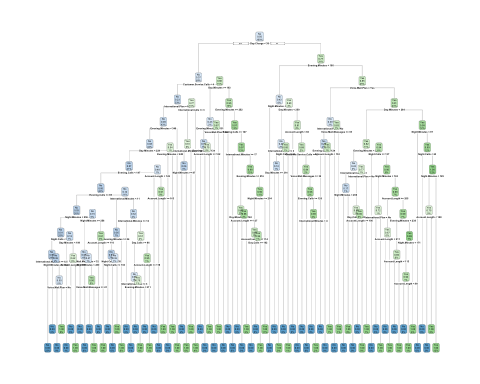
* *First, use the train set to fid the model*

library(rpart)  
tree.fit<-rpart(formula = Churn~.,data=churn.train.balanced,method = "class",control = rpart.control(minsplit = 2,cp=0))

* *Plot the resulting fit*

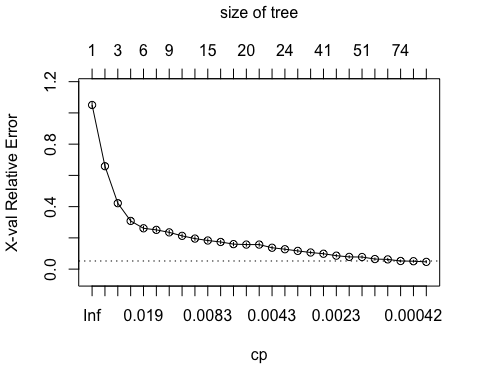
library(rpart.plot)  
rpart.plot(tree.fit)

## Warning: labs do not fit even at cex 0.15, there may be some overplotting



* *Plot the cp against the cross validated error and also use the cp table*

plotcp(tree.fit)



printcp(tree.fit)

##   
## Classification tree:  
## rpart(formula = Churn ~ ., data = churn.train.balanced, method = "class",   
## control = rpart.control(minsplit = 2, cp = 0))  
##   
## Variables actually used in tree construction:  
## [1] Account.Length Customer.Service.Calls Day.Calls   
## [4] Day.Charge Day.Minutes Evening.Calls   
## [7] Evening.Minutes International.Calls International.Minutes   
## [10] International.Plan Night.Calls Night.Minutes   
## [13] Voice.Mail.Messages Voice.Mail.Plan   
##   
## Root node error: 1381/2762 = 0.5  
##   
## n= 2762   
##   
## CP nsplit rel error xerror xstd  
## 1 0.34322954 0 1.0000000 1.050688 0.0190033  
## 2 0.23606083 1 0.6567705 0.658943 0.0178869  
## 3 0.11368573 2 0.4207096 0.422158 0.0155295  
## 4 0.02244750 3 0.3070239 0.308472 0.0137447  
## 5 0.01629254 5 0.2621289 0.261405 0.0128276  
## 6 0.01375815 7 0.2295438 0.251267 0.0126130  
## 7 0.01230992 8 0.2157857 0.235337 0.0122621  
## 8 0.01086169 9 0.2034757 0.212889 0.0117366  
## 9 0.00868936 11 0.1817524 0.195510 0.0113019  
## 10 0.00796524 14 0.1556843 0.183201 0.0109775  
## 11 0.00627565 15 0.1477190 0.173787 0.0107195  
## 12 0.00579290 18 0.1288921 0.160029 0.0103251  
## 13 0.00543085 19 0.1230992 0.157133 0.0102393  
## 14 0.00506879 21 0.1122375 0.157133 0.0102393  
## 15 0.00362056 22 0.1071687 0.136857 0.0096083  
## 16 0.00337919 23 0.1035482 0.127444 0.0092953  
## 17 0.00325851 30 0.0774801 0.116582 0.0089162  
## 18 0.00289645 32 0.0709631 0.106445 0.0085426  
## 19 0.00241371 40 0.0477915 0.098479 0.0082340  
## 20 0.00217234 43 0.0405503 0.086169 0.0077271  
## 21 0.00181028 48 0.0296886 0.078204 0.0073766  
## 22 0.00144823 50 0.0260681 0.077480 0.0073438  
## 23 0.00120685 55 0.0188269 0.064446 0.0067203  
## 24 0.00072411 58 0.0152064 0.062274 0.0066098  
## 25 0.00048274 73 0.0043447 0.052136 0.0060637  
## 26 0.00036206 76 0.0028965 0.050688 0.0059811  
## 27 0.00000000 84 0.0000000 0.046343 0.0057254

* *From the cp table and the grap, we choose a cp value in the 0.00054785 to 0.00045 range*

tree.prune<-prune(tree.fit,cp=0.00054)

* *Get the cp table*

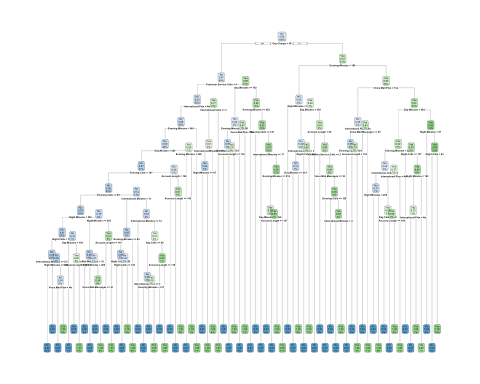
printcp(tree.prune)

##   
## Classification tree:  
## rpart(formula = Churn ~ ., data = churn.train.balanced, method = "class",   
## control = rpart.control(minsplit = 2, cp = 0))  
##   
## Variables actually used in tree construction:  
## [1] Account.Length Customer.Service.Calls Day.Calls   
## [4] Day.Charge Day.Minutes Evening.Calls   
## [7] Evening.Minutes International.Calls International.Minutes   
## [10] International.Plan Night.Calls Night.Minutes   
## [13] Voice.Mail.Messages Voice.Mail.Plan   
##   
## Root node error: 1381/2762 = 0.5  
##   
## n= 2762   
##   
## CP nsplit rel error xerror xstd  
## 1 0.34322954 0 1.0000000 1.050688 0.0190033  
## 2 0.23606083 1 0.6567705 0.658943 0.0178869  
## 3 0.11368573 2 0.4207096 0.422158 0.0155295  
## 4 0.02244750 3 0.3070239 0.308472 0.0137447  
## 5 0.01629254 5 0.2621289 0.261405 0.0128276  
## 6 0.01375815 7 0.2295438 0.251267 0.0126130  
## 7 0.01230992 8 0.2157857 0.235337 0.0122621  
## 8 0.01086169 9 0.2034757 0.212889 0.0117366  
## 9 0.00868936 11 0.1817524 0.195510 0.0113019  
## 10 0.00796524 14 0.1556843 0.183201 0.0109775  
## 11 0.00627565 15 0.1477190 0.173787 0.0107195  
## 12 0.00579290 18 0.1288921 0.160029 0.0103251  
## 13 0.00543085 19 0.1230992 0.157133 0.0102393  
## 14 0.00506879 21 0.1122375 0.157133 0.0102393  
## 15 0.00362056 22 0.1071687 0.136857 0.0096083  
## 16 0.00337919 23 0.1035482 0.127444 0.0092953  
## 17 0.00325851 30 0.0774801 0.116582 0.0089162  
## 18 0.00289645 32 0.0709631 0.106445 0.0085426  
## 19 0.00241371 40 0.0477915 0.098479 0.0082340  
## 20 0.00217234 43 0.0405503 0.086169 0.0077271  
## 21 0.00181028 48 0.0296886 0.078204 0.0073766  
## 22 0.00144823 50 0.0260681 0.077480 0.0073438  
## 23 0.00120685 55 0.0188269 0.064446 0.0067203  
## 24 0.00072411 58 0.0152064 0.062274 0.0066098  
## 25 0.00054000 73 0.0043447 0.052136 0.0060637

* *Plot the resulting tree*

rpart.plot(tree.prune)

## Warning: labs do not fit even at cex 0.15, there may be some overplotting



* *Do a confusion matrix on the test set*

library(caret)

## Loading required package: lattice

## Loading required package: ggplot2

pred<-predict(tree.prune,newdata = churn.test,type="class")  
confusion.rpart<-confusionMatrix(pred,churn.test$Churn)  
confusion.rpart

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction No Yes  
## No 325 22  
## Yes 13 41  
##   
## Accuracy : 0.9127   
## 95% CI : (0.8807, 0.9385)  
## No Information Rate : 0.8429   
## P-Value [Acc > NIR] : 2.691e-05   
##   
## Kappa : 0.6501   
##   
## Mcnemar's Test P-Value : 0.1763   
##   
## Sensitivity : 0.9615   
## Specificity : 0.6508   
## Pos Pred Value : 0.9366   
## Neg Pred Value : 0.7593   
## Prevalence : 0.8429   
## Detection Rate : 0.8105   
## Detection Prevalence : 0.8653   
## Balanced Accuracy : 0.8062   
##   
## 'Positive' Class : No   
##

### Logistic Regression

* *Fit the logistic regression model*

glm.fit<-glm(Churn~Account.Length+International.Plan+Voice.Mail.Plan+Day.Minutes+Day.Calls+Evening.Calls+Evening.Charge+Night.Minutes+Night.Calls+International.Minutes+International.Calls+Customer.Service.Calls,data = churn.train.balanced,family = binomial)

* *See the result summary*

summary(glm.fit)

##   
## Call:  
## glm(formula = Churn ~ Account.Length + International.Plan + Voice.Mail.Plan +   
## Day.Minutes + Day.Calls + Evening.Calls + Evening.Charge +   
## Night.Minutes + Night.Calls + International.Minutes + International.Calls +   
## Customer.Service.Calls, family = binomial, data = churn.train.balanced)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.96284 -0.71468 -0.09365 0.81246 2.86833   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -8.2988520 0.6061472 -13.691 < 2e-16 \*\*\*  
## Account.Length 0.0032653 0.0011975 2.727 0.006398 \*\*   
## International.PlanYes 2.7701428 0.1600867 17.304 < 2e-16 \*\*\*  
## Voice.Mail.PlanYes -1.0511945 0.1281390 -8.204 2.33e-16 \*\*\*  
## Day.Minutes 0.0165604 0.0009691 17.088 < 2e-16 \*\*\*  
## Day.Calls 0.0078230 0.0023023 3.398 0.000679 \*\*\*  
## Evening.Calls 0.0025657 0.0022932 1.119 0.263218   
## Evening.Charge 0.0999494 0.0115788 8.632 < 2e-16 \*\*\*  
## Night.Minutes 0.0018139 0.0010134 1.790 0.073464 .   
## Night.Calls -0.0036580 0.0024475 -1.495 0.135025   
## International.Minutes 0.0451558 0.0171983 2.626 0.008650 \*\*   
## International.Calls -0.0444937 0.0192381 -2.313 0.020734 \*   
## Customer.Service.Calls 0.7510210 0.0374833 20.036 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 3820.6 on 2755 degrees of freedom  
## Residual deviance: 2628.1 on 2743 degrees of freedom  
## (6 observations deleted due to missingness)  
## AIC: 2654.1  
##   
## Number of Fisher Scoring iterations: 5

* *Use VIF() from the car pacakge to detec multicolineairty, and leave out the variables one by one until the VIF scores for all variables are low*

library(car)

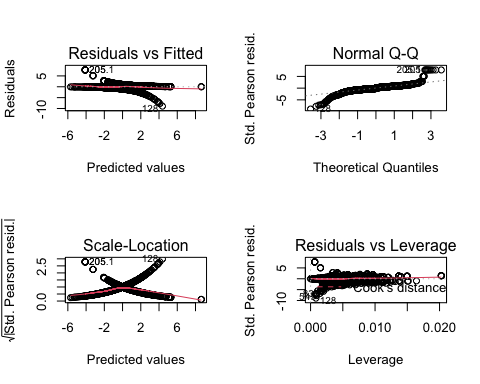
## Loading required package: carData

vif(glm.fit)

## Account.Length International.Plan Voice.Mail.Plan   
## 1.022476 1.186083 1.034149   
## Day.Minutes Day.Calls Evening.Calls   
## 1.357975 1.016853 1.010690   
## Evening.Charge Night.Minutes Night.Calls   
## 1.128208 1.047221 1.010959   
## International.Minutes International.Calls Customer.Service.Calls   
## 1.016548 1.038089 1.524052

* \*\*Plot the diagonsitic plot\*

par(mfrow=c(2,2))  
plot(glm.fit)



* *Use the model to make prediction on a test set. And make a confusion matrix from the prediction.*

library(caret)  
glm.probs=predict(glm.fit,churn.test,type="response")  
glm.pred=rep("No",nrow(churn.test))  
glm.pred[glm.probs>.5]="Yes"  
confusion.log<-confusionMatrix(as.factor(glm.pred),churn.test$Churn)  
confusion.log

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction No Yes  
## No 263 20  
## Yes 75 43  
##   
## Accuracy : 0.7631   
## 95% CI : (0.7184, 0.8039)  
## No Information Rate : 0.8429   
## P-Value [Acc > NIR] : 1   
##   
## Kappa : 0.3399   
##   
## Mcnemar's Test P-Value : 3.02e-08   
##   
## Sensitivity : 0.7781   
## Specificity : 0.6825   
## Pos Pred Value : 0.9293   
## Neg Pred Value : 0.3644   
## Prevalence : 0.8429   
## Detection Rate : 0.6559   
## Detection Prevalence : 0.7057   
## Balanced Accuracy : 0.7303   
##   
## 'Positive' Class : No   
##

* *Let’s compare the confusion matrix of the two models*

confusion.rpart

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction No Yes  
## No 325 22  
## Yes 13 41  
##   
## Accuracy : 0.9127   
## 95% CI : (0.8807, 0.9385)  
## No Information Rate : 0.8429   
## P-Value [Acc > NIR] : 2.691e-05   
##   
## Kappa : 0.6501   
##   
## Mcnemar's Test P-Value : 0.1763   
##   
## Sensitivity : 0.9615   
## Specificity : 0.6508   
## Pos Pred Value : 0.9366   
## Neg Pred Value : 0.7593   
## Prevalence : 0.8429   
## Detection Rate : 0.8105   
## Detection Prevalence : 0.8653   
## Balanced Accuracy : 0.8062   
##   
## 'Positive' Class : No   
##

confusion.log

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction No Yes  
## No 263 20  
## Yes 75 43  
##   
## Accuracy : 0.7631   
## 95% CI : (0.7184, 0.8039)  
## No Information Rate : 0.8429   
## P-Value [Acc > NIR] : 1   
##   
## Kappa : 0.3399   
##   
## Mcnemar's Test P-Value : 3.02e-08   
##   
## Sensitivity : 0.7781   
## Specificity : 0.6825   
## Pos Pred Value : 0.9293   
## Neg Pred Value : 0.3644   
## Prevalence : 0.8429   
## Detection Rate : 0.6559   
## Detection Prevalence : 0.7057   
## Balanced Accuracy : 0.7303   
##   
## 'Positive' Class : No   
##

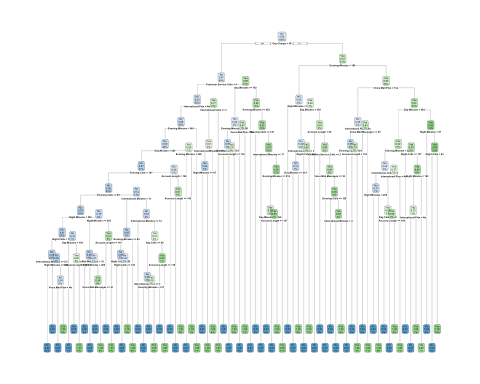
* *From the confusion matrix, altough the logistics regression has a little bit lower False Negative but significantly higher False Positives, so the better model s the rpart model*

1. **Based on findings from the models in 3, explain and advise the Telecom company management.**

* \*\*From the rpart fit\*

rpart.plot(tree.prune)

## Warning: labs do not fit even at cex 0.15, there may be some overplotting



* *Use the variable importance to find the variables that improve the purity the most*

tree.prune$variable.importance

## Day.Minutes Customer.Service.Calls Day.Charge   
## 319.49835 311.28556 310.01820   
## Evening.Minutes International.Plan Evening.Charge   
## 239.64197 238.89089 227.95449   
## International.Minutes International.Charge Night.Calls   
## 134.04130 125.80266 109.81534   
## Evening.Calls Night.Minutes Day.Calls   
## 109.79502 108.98140 99.28923   
## Night.Charge International.Calls Voice.Mail.Messages   
## 95.80665 90.03778 86.35745   
## Account.Length Voice.Mail.Plan   
## 82.95285 58.46749

* *From the result, we see that the number of minutes talked during the day, and the duration of the customer service calls, as well as whether the customer is a international plan subscriber has the most influence on the outcome*
* *We can also use the logistics regression result to detect how some factors affect the odds*

summary(glm.fit)

##   
## Call:  
## glm(formula = Churn ~ Account.Length + International.Plan + Voice.Mail.Plan +   
## Day.Minutes + Day.Calls + Evening.Calls + Evening.Charge +   
## Night.Minutes + Night.Calls + International.Minutes + International.Calls +   
## Customer.Service.Calls, family = binomial, data = churn.train.balanced)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.96284 -0.71468 -0.09365 0.81246 2.86833   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -8.2988520 0.6061472 -13.691 < 2e-16 \*\*\*  
## Account.Length 0.0032653 0.0011975 2.727 0.006398 \*\*   
## International.PlanYes 2.7701428 0.1600867 17.304 < 2e-16 \*\*\*  
## Voice.Mail.PlanYes -1.0511945 0.1281390 -8.204 2.33e-16 \*\*\*  
## Day.Minutes 0.0165604 0.0009691 17.088 < 2e-16 \*\*\*  
## Day.Calls 0.0078230 0.0023023 3.398 0.000679 \*\*\*  
## Evening.Calls 0.0025657 0.0022932 1.119 0.263218   
## Evening.Charge 0.0999494 0.0115788 8.632 < 2e-16 \*\*\*  
## Night.Minutes 0.0018139 0.0010134 1.790 0.073464 .   
## Night.Calls -0.0036580 0.0024475 -1.495 0.135025   
## International.Minutes 0.0451558 0.0171983 2.626 0.008650 \*\*   
## International.Calls -0.0444937 0.0192381 -2.313 0.020734 \*   
## Customer.Service.Calls 0.7510210 0.0374833 20.036 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 3820.6 on 2755 degrees of freedom  
## Residual deviance: 2628.1 on 2743 degrees of freedom  
## (6 observations deleted due to missingness)  
## AIC: 2654.1  
##   
## Number of Fisher Scoring iterations: 5

* *From the coefficients of the various factors, we do see account lenghth, international plan subscription, number of minutes during the day, amount of evening charges, and amount od night call durations, as well as the international calls have a positive impact on the odds of churn. A one unit increase in one of these factors, given the other factors are constant, will increase the avergae odds of yes on churn. In constrast, for attributes having negative coefficients, the customer is a voice mail plan subscriber, the amound to night calls, and the amount of international calls have a negative impact on the outcome. That is, a one unit increase in one of these variables, given the others are constant, will have a negative impact on the odds of being Yes on churn on average.*

## Part C: Advanced Concepts

1. **The cross validation error in your optimal CART is reported in the rpart package cp table. *If the outcome variable is continuous, this is fine. But if the outcome variable is categorical, an important information is missing.* Explain the last two sentences (in Bold) above.**

* *Since regression tree utilizes mse as a metrics for test error, but classification generally utilizes purity measures such as gini/entropy. At times when regression tree will see an improvement in a split beacuse of improvement in mse, the classification tree might see no improvement because the purity stays the same.AS a result, regression will make a split but classification won’t. Even if the cross validated error is lower for regression compared with classification, the prediction based on class will be the same.*

1. **How did CART deal with the missing value(s) if any, in the dataset?**

* *rpart package will try to find 5 X variables that will result in the best split at each node as a back up plan.rpart will rank them based on the best primary split. If there is a missing variable, rpart will use the surroagte variables so that the split could be made.*