|  |  |
| --- | --- |
| Group 7 pROJECT REPORT  CUSTOMER CHURN PREDICTION | Prajwal C N & Kunal Sharma  BUSINESS ANALYTICS MIS 64036 |

ABSTRACT

Churn is a problem for telecom companies because it is more difficult to acquire new customers than it is to keep existing customers from leaving. Customer Churn Modeling has received a lot of attention recently, as there are signs that existing customers generate a large portion of corporate profit. Companies are also very interested in identifying consumers who are likely to become churn, and they often use Data Mining methods to help them do so. We recognized customers who are likely to churn in this project and provided sufficient intercession to encourage them to remain based on available data.

INTRODUCTION

Customer retention and acquisition are important factors that have a direct impact on a company's profitability and striking a balance between the two is not easy. Since the churn rate influences the company's sales, customer retention is likely to be a key factor.

Customers can switch service providers for a variety of reasons, including network problems, poor customer service, and a high monthly plan. These problems can be addressed by offering discounts or improved service, among other things, to keep customers from switching service providers.

We can use this knowledge to derive patterns and forecast future outcomes in modern times when we have the analytical capabilities to interpret and analyze complex data and extract useful information.

The aim of this project is to use a predictive model to evaluate data and identify trends to predict when an established customer will switch service providers. Many predictive models, such as Nave bias, K nearest neighbor, and regression, can be used to perform our study. Here, we'll use logistic regression to construct our model.

Overview of Data

ABC wireless company has provided the following data from which we can infer:

* Demographics
  + State
  + Account length
  + Area code
  + International plan
  + Voice-mail plan
* Calling Behaviour
  + Number of messages
  + Total day minutes, Total day calls, Total day charge
  + Total evening minutes, Total evening calls and Total evening charges
  + Total night minutes, Total night calls and Total night charges
  + Total International minutes, Total International calls and Total International charges
  + Number of calls to customer service

**Data Preprocessing**

#### Importing the Churn dataset

Churn\_Data <- read\_csv("Churn\_Train.csv")  
  
# Inspecting data  
head(Churn\_Data)

## # A tibble: 6 x 20  
## state account\_length area\_code international\_plan voice\_mail\_plan  
## <chr> <dbl> <chr> <chr> <chr>   
## 1 NV 125 area\_code\_510 no no   
## 2 HI 108 area\_code\_415 no no   
## 3 DC 82 area\_code\_415 no no   
## 4 HI NA area\_code\_408 no yes   
## 5 OH 83 area\_code\_415 no no   
## 6 MO 89 area\_code\_415 no no   
## # ... with 15 more variables: number\_vmail\_messages <dbl>,  
## # total\_day\_minutes <dbl>, total\_day\_calls <dbl>, total\_day\_charge <dbl>,  
## # total\_eve\_minutes <dbl>, total\_eve\_calls <dbl>, total\_eve\_charge <dbl>,  
## # total\_night\_minutes <dbl>, total\_night\_calls <dbl>,  
## # total\_night\_charge <dbl>, total\_intl\_minutes <dbl>, total\_intl\_calls <dbl>,  
## # total\_intl\_charge <dbl>, number\_customer\_service\_calls <dbl>, churn <chr>

*Examining the dataset*

## Rows: 3,333  
## Columns: 20  
## $ state <chr> "NV", "HI", "DC", "HI", "OH", "MO", "NC"~  
## $ account\_length <dbl> 125, 108, 82, NA, 83, 89, 135, 28, 86, 6~  
## $ area\_code <chr> "area\_code\_510", "area\_code\_415", "area\_~  
## $ international\_plan <chr> "no", "no", "no", "no", "no", "no", "no"~  
## $ voice\_mail\_plan <chr> "no", "no", "no", "yes", "no", "no", "no~  
## $ number\_vmail\_messages <dbl> 0, 0, 0, 30, 0, 0, 0, 0, 0, 0, 0, NA, 32~  
## $ total\_day\_minutes <dbl> 2013.4, 291.6, 300.3, 110.3, 337.4, 178.~  
## $ total\_day\_calls <dbl> 99, 99, 109, 71, 120, 81, 81, 87, 115, 1~  
## $ total\_day\_charge <dbl> 28.66, 49.57, 51.05, 18.75, 57.36, 30.38~  
## $ total\_eve\_minutes <dbl> 1107.6, 221.1, 181.0, 182.4, 227.4, NA, ~  
## $ total\_eve\_calls <dbl> 107, 93, 100, 108, 116, 74, 114, 92, 112~  
## $ total\_eve\_charge <dbl> 14.93, 18.79, 15.39, 15.50, 19.33, 19.86~  
## $ total\_night\_minutes <dbl> 243.3, 229.2, 270.1, 183.8, 153.9, 131.9~  
## $ total\_night\_calls <dbl> 92, 110, 73, 88, 114, 120, 82, 112, 95, ~  
## $ total\_night\_charge <dbl> 10.95, 10.31, 12.15, 8.27, 6.93, 5.94, 9~  
## $ total\_intl\_minutes <dbl> 10.9, 14.0, 11.7, 11.0, 15.8, 9.1, 10.3,~  
## $ total\_intl\_calls <dbl> 7, 9, 4, 8, 7, 4, 6, 3, 7, 6, 7, NA, 4, ~  
## $ total\_intl\_charge <dbl> 2.94, 3.78, 3.16, 2.97, 4.27, 2.46, 2.78~  
## $ number\_customer\_service\_calls <dbl> 0, 2, 0, 2, 0, 1, 1, 3, 2, 4, 1, NA, 3, ~  
## $ churn <chr> "no", "yes", "yes", "no", "yes", "no", "~

*Summary statistics of dataset*

summary(Churn\_Data)

## state account\_length area\_code international\_plan  
## Length:3333 Min. :-209.00 Length:3333 Length:3333   
## Class :character 1st Qu.: 72.00 Class :character Class :character   
## Mode :character Median : 100.00 Mode :character Mode :character   
## Mean : 97.32   
## 3rd Qu.: 127.00   
## Max. : 243.00   
## NA's :501   
## voice\_mail\_plan number\_vmail\_messages total\_day\_minutes total\_day\_calls  
## Length:3333 Min. :-10.000 Min. : 0.0 Min. : 0.0   
## Class :character 1st Qu.: 0.000 1st Qu.: 149.3 1st Qu.: 87.0   
## Mode :character Median : 0.000 Median : 190.5 Median :101.0   
## Mean : 7.333 Mean : 418.9 Mean :100.3   
## 3rd Qu.: 16.000 3rd Qu.: 237.8 3rd Qu.:114.0   
## Max. : 51.000 Max. :2185.1 Max. :165.0   
## NA's :200 NA's :200 NA's :200   
## total\_day\_charge total\_eve\_minutes total\_eve\_calls total\_eve\_charge  
## Min. : 0.00 Min. : 0.0 Min. : 0.0 Min. : 0.00   
## 1st Qu.:24.45 1st Qu.: 170.5 1st Qu.: 87.0 1st Qu.:14.14   
## Median :30.65 Median : 209.9 Median :100.0 Median :17.09   
## Mean :30.63 Mean : 324.3 Mean :100.1 Mean :17.08   
## 3rd Qu.:36.84 3rd Qu.: 257.6 3rd Qu.:114.0 3rd Qu.:20.00   
## Max. :59.64 Max. :1244.2 Max. :170.0 Max. :30.91   
## NA's :200 NA's :301 NA's :200 NA's :200   
## total\_night\_minutes total\_night\_calls total\_night\_charge total\_intl\_minutes  
## Min. : 23.2 Min. : 33.0 Min. : 1.040 Min. : 0.00   
## 1st Qu.:167.3 1st Qu.: 87.0 1st Qu.: 7.530 1st Qu.: 8.50   
## Median :201.4 Median :100.0 Median : 9.060 Median :10.30   
## Mean :201.2 Mean :100.1 Mean : 9.054 Mean :10.23   
## 3rd Qu.:235.3 3rd Qu.:113.0 3rd Qu.:10.590 3rd Qu.:12.10   
## Max. :395.0 Max. :175.0 Max. :17.770 Max. :20.00   
## NA's :200 NA's :200 NA's :200   
## total\_intl\_calls total\_intl\_charge number\_customer\_service\_calls  
## Min. : 0.00 Min. :0.000 Min. :0.000   
## 1st Qu.: 3.00 1st Qu.:2.300 1st Qu.:1.000   
## Median : 4.00 Median :2.780 Median :1.000   
## Mean : 4.47 Mean :2.762 Mean :1.561   
## 3rd Qu.: 6.00 3rd Qu.:3.270 3rd Qu.:2.000   
## Max. :20.00 Max. :5.400 Max. :9.000   
## NA's :301 NA's :200 NA's :200   
## churn   
## Length:3333   
## Class :character   
## Mode :character   
##   
##   
##   
##

*Data cleaning and Exploratory Data Analysis*

From glimpse we can see that, Some of the character variables can be converted into factors, So Converting character variables to factors.

Churn\_Data <- Churn\_Data %>% mutate\_if(is.character, as.factor)

From summary we can see that, Churn\_Data dataset has both NA and negative values, So investigating and handling further.

# Checking NULL values in the dataset at column level.  
colSums(is.na(Churn\_Data))

## state account\_length   
## 0 501   
## area\_code international\_plan   
## 0 0   
## voice\_mail\_plan number\_vmail\_messages   
## 0 200   
## total\_day\_minutes total\_day\_calls   
## 200 200   
## total\_day\_charge total\_eve\_minutes   
## 200 301   
## total\_eve\_calls total\_eve\_charge   
## 200 200   
## total\_night\_minutes total\_night\_calls   
## 200 0   
## total\_night\_charge total\_intl\_minutes   
## 200 200   
## total\_intl\_calls total\_intl\_charge   
## 301 200   
## number\_customer\_service\_calls churn   
## 200 0

# Checking Negative values in the dataset at column level.  
sapply(Churn\_Data %>% select\_if(is.numeric), function(x) {  
 sum(x < 0, na.rm = TRUE)  
})

## account\_length number\_vmail\_messages   
## 51 201   
## total\_day\_minutes total\_day\_calls   
## 0 0   
## total\_day\_charge total\_eve\_minutes   
## 0 0   
## total\_eve\_calls total\_eve\_charge   
## 0 0   
## total\_night\_minutes total\_night\_calls   
## 0 0   
## total\_night\_charge total\_intl\_minutes   
## 0 0   
## total\_intl\_calls total\_intl\_charge   
## 0 0   
## number\_customer\_service\_calls   
## 0

Since account length, and other numeric variables has few negative values, assuming them as erroneous values, we cannot void them because their corresponding churn value is “no” which means they are still associated with the provider.

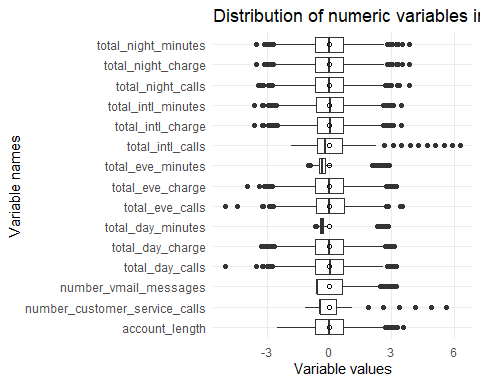
Churn\_Data <-  
 Churn\_Data %>% mutate\_if(is.numeric, function(x) {  
 ifelse(x < 0, abs(x), x)  
 })

From the above plot, we see there are outliers in the data, in order to impute NA values, we have several techniques such as mean, median, KNN imputation and linear regression. Since there are many outliers in the data, its not feasible to do mean imputation. Hence using **median imputation** technique.

imputation\_model <- preProcess(Churn\_Data %>% select\_if(is.numeric),method = "medianImpute")  
data <- predict(imputation\_model, Churn\_Data %>% select\_if(is.numeric))  
  
Churn\_Data <- Churn\_Data %>% select(setdiff(names(Churn\_Data), names(data))) %>% cbind(data)

**Visualizing distribution of Churn numeric variable.**

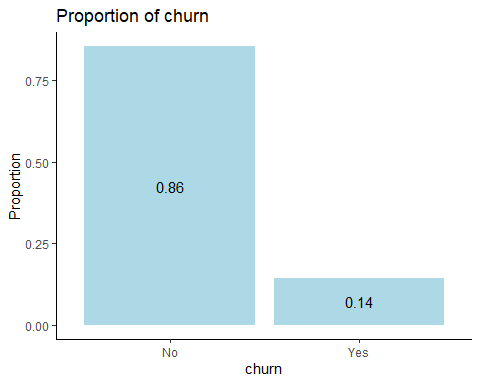
Churn\_Data %>% select\_if(is.numeric) %>% mutate\_all(scale) %>% gather("features","values") %>% na.omit() %>%   
 ggplot(aes(x = features, y = values)) +  
 geom\_boxplot(show.legend = FALSE) +  
 stat\_summary(fun = mean, geom = "point", pch = 1) + # Add average to the boxplot  
 scale\_y\_continuous(name = "Variable values", minor\_breaks = NULL) +  
 scale\_fill\_brewer(palette = "Set1") +  
 coord\_flip() +   
 theme\_minimal() +  
 labs(x = "Variable names") +  
 ggtitle(label = "Distribution of numeric variables in Churn dataset")



From above box plot we can see that, most of the variables are not normally distributed and there are many outliers in the data.

**Visualizing distribution of Churn categorical variable.**

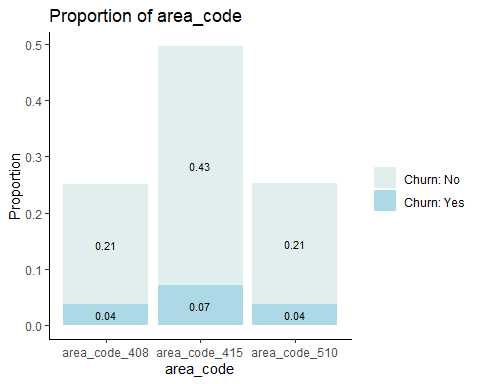
ggplot(Churn\_Data, aes(x=churn, y=..prop..,group = 1)) +   
 geom\_bar(fill="light blue") +  
 theme\_classic() +   
 geom\_text(aes(label=round(..prop..,2)),stat = "count",  
 position = position\_stack(vjust=0.5)) +   
 labs(y = 'Proportion', title = "Proportion of churn") +  
 scale\_x\_discrete(labels = c("No","Yes"))



From the above graph we can see that only around 14% of population are churned and rest 86% are retained in the telecom network.

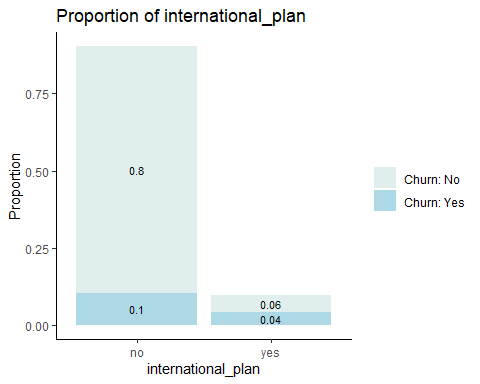
**Proportion of area\_code**

as.data.frame(prop.table(table(Churn\_Data[c("area\_code","churn")]))) %>%   
 ggplot(aes(x=area\_code,y=Freq,fill=churn)) + geom\_col() +   
 geom\_text(aes(label=round(Freq,2)),position = position\_stack(vjust = 0.5),size=2.8) +   
 theme\_classic() + labs( y = 'Proportion', title = "Proportion of area\_code") +   
 theme(legend.title = element\_blank()) +   
 scale\_fill\_manual(labels = c("Churn: No","Churn: Yes"),   
 values = c("azure2","light blue"))



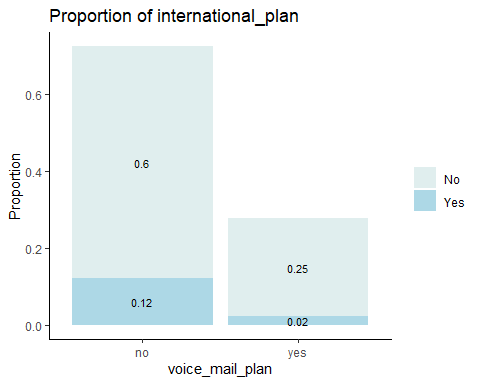
**Proportion of international\_plan**

as.data.frame(prop.table(table(Churn\_Data[c("international\_plan","churn")]))) %>%   
 ggplot(aes(x=international\_plan,y=Freq,fill=churn)) + geom\_col() +   
 geom\_text(aes(label=round(Freq,2)),position = position\_stack(vjust = 0.5),size=2.8) +   
 theme\_classic() + labs( y = 'Proportion', title = "Proportion of international\_plan") +   
 theme(legend.title = element\_blank()) +   
 scale\_fill\_manual(labels = c("Churn: No","Churn: Yes"),   
 values = c("azure2","light blue"))



**Proportion of voice\_mail\_plan**

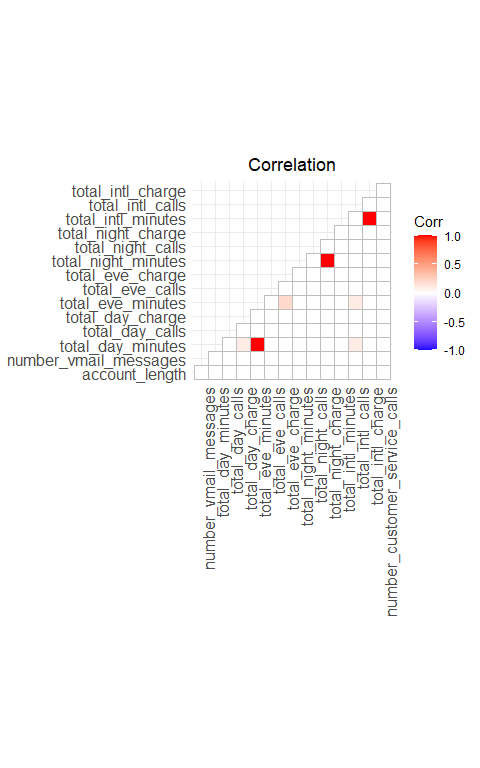
as.data.frame(prop.table(table(Churn\_Data[c("voice\_mail\_plan","churn")]))) %>%   
 ggplot(aes(x=voice\_mail\_plan,y=Freq,fill=churn)) + geom\_col() +   
 geom\_text(aes(label=round(Freq,2)),position = position\_stack(vjust = 0.5),size=2.8) +   
 theme\_classic() + labs( y = 'Proportion', title = "Proportion of international\_plan") +   
 theme(legend.title = element\_blank()) +   
 scale\_fill\_manual(labels = c("No","Yes"),   
 values = c("azure2","light blue"))



**Correlation**

The image below will assist us in determining the variables’ correlation.

Churn\_Data\_cor <- round(cor(Churn\_Data %>% select\_if(is.numeric)), 1)  
  
ggcorrplot(Churn\_Data\_cor, title = "Correlation", type = "lower") +  
 theme(plot.title = element\_text(hjust = 0.5),  
 axis.text.x = element\_text(angle = 90))



Total minutes and total charge for the day, evening, night, and international are strongly linked, we may omit them since they can cause “multi-collinearity” issue.

**Model Strategy**

The task of using a classifier to divide an example into two categories is known as binary classification. Since the target variable in this data is categorical, and outcome for this model is a likelihood or probability of odds between 0 and 1, so we will using both **logistic regression** and **decision tree** to solve this problem and comparing both models performance matrix and choosing better one.

**Logistic Regression**

#### Pre-Processing of data

**Splitting dataset into training (80%) and validation (20%) sets**

The training set will be used to fit our model which we will be testing over the testing set.

set.seed(12)  
index <- createDataPartition(Churn\_Data$churn, p=0.8, list=FALSE)  
Churn\_Data\_train\_df <- Churn\_Data[index,]  
Churn\_Data\_test\_df <- Churn\_Data[-index,]

**Scaling train and test churn datasets**

scaling <- preProcess(Churn\_Data\_train\_df %>% select\_if(is.numeric), method = c("center", "scale"))  
Churn\_Data\_train\_norm <- predict(scaling, Churn\_Data\_train\_df %>% select\_if(is.numeric))  
Churn\_Data\_test\_norm <- predict(scaling, Churn\_Data\_test\_df %>% select\_if(is.numeric))  
  
Churn\_Data\_train\_norm$churn <- Churn\_Data\_train\_df$churn  
Churn\_Data\_test\_norm$churn <- Churn\_Data\_test\_df$churn

#### Model Construction

Model\_1 <- glm(churn ~ ., data = Churn\_Data\_train\_norm , family= "binomial")

summary(Model\_1)

##   
## Call:  
## glm(formula = churn ~ ., family = "binomial", data = Churn\_Data\_train\_norm)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.1055 -0.5099 -0.3508 -0.2013 3.1185   
##   
## Coefficients: (3 not defined because of singularities)  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.300038 0.079590 -28.899 < 2e-16 \*\*\*  
## account\_length 0.018545 0.062143 0.298 0.76538   
## number\_vmail\_messages 0.127134 0.177494 0.716 0.47382   
## total\_day\_minutes -2.292991 1.250524 -1.834 0.06671 .   
## total\_day\_calls 0.046819 0.061769 0.758 0.44847   
## total\_day\_charge 0.853647 0.117549 7.262 3.81e-13 \*\*\*  
## total\_eve\_minutes 2.184432 1.242318 1.758 0.07869 .   
## total\_eve\_calls -0.041535 0.061678 -0.673 0.50068   
## total\_eve\_charge 0.068285 0.194636 0.351 0.72571   
## total\_night\_minutes 1.835634 47.612807 0.039 0.96925   
## total\_night\_calls 0.038145 0.062195 0.613 0.53967   
## total\_night\_charge -1.703082 47.611535 -0.036 0.97147   
## total\_intl\_minutes -5.955075 16.386684 -0.363 0.71630   
## total\_intl\_calls -0.186343 0.066424 -2.805 0.00503 \*\*   
## total\_intl\_charge 6.191261 16.384772 0.378 0.70553   
## number\_customer\_service\_calls 0.689509 0.056961 12.105 < 2e-16 \*\*\*  
## area\_code\_area\_code\_408 0.005171 0.075775 0.068 0.94559   
## area\_code\_area\_code\_415 -0.025656 0.075043 -0.342 0.73244   
## area\_code\_area\_code\_510 NA NA NA NA   
## international\_plan\_no -0.603710 0.048133 -12.542 < 2e-16 \*\*\*  
## international\_plan\_yes NA NA NA NA   
## voice\_mail\_plan\_no 0.560696 0.181780 3.084 0.00204 \*\*   
## voice\_mail\_plan\_yes NA NA NA NA   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 2209 on 2666 degrees of freedom  
## Residual deviance: 1740 on 2647 degrees of freedom  
## AIC: 1780  
##   
## Number of Fisher Scoring iterations: 6

Now we can infer from the summary of the model, the significant variables, p values, test statistics etc..

**Predict values using based on Model\_1.**

pred\_probs <- predict(object = Model\_1,Churn\_Data\_test\_norm, type = "response")  
  
# Finding accuracy for the model  
# Function to find the accuracy, based on probability(0.5 - 0.9)   
sequence1 <- data.frame(pred\_cutoff = seq(0.5,0.9,0.1), pred\_accuracy = rep(0,5))  
  
for (i in 1:5){  
 Model\_11 <- as.factor(ifelse(pred\_probs > sequence1$pred\_cutoff[i], "yes", "no"))  
 sequence1[i,2] <- confusionMatrix(Model\_11,Churn\_Data\_test\_df$churn )$overall[1]  
}  
  
# Shows the probability with its accuracy  
sequence1

## pred\_cutoff pred\_accuracy  
## 1 0.5 0.8678679  
## 2 0.6 0.8693694  
## 3 0.7 0.8663664  
## 4 0.8 0.8603604  
## 5 0.9 0.8588589

# *Assigning labels based on maximum probability prediction*

Model\_Pre\_lables <- as.factor(ifelse(pred\_probs>sequence1$pred\_cutoff[which.max(sequence1$pred\_accuracy)] ,"yes","no"))

#### Performance Metrics

Confusion matrix for churn model.

confusionMatrix(Model\_Pre\_lables,Churn\_Data\_test\_norm$churn)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction no yes  
## no 563 80  
## yes 7 16  
##   
## Accuracy : 0.8694   
## 95% CI : (0.8414, 0.894)  
## No Information Rate : 0.8559   
## P-Value [Acc > NIR] : 0.1746   
##   
## Kappa : 0.2258   
##   
## Mcnemar's Test P-Value : 1.171e-14   
##   
## Sensitivity : 0.9877   
## Specificity : 0.1667   
## Pos Pred Value : 0.8756   
## Neg Pred Value : 0.6957   
## Prevalence : 0.8559   
## Detection Rate : 0.8453   
## Detection Prevalence : 0.9655   
## Balanced Accuracy : 0.5772   
##   
## 'Positive' Class : no   
##

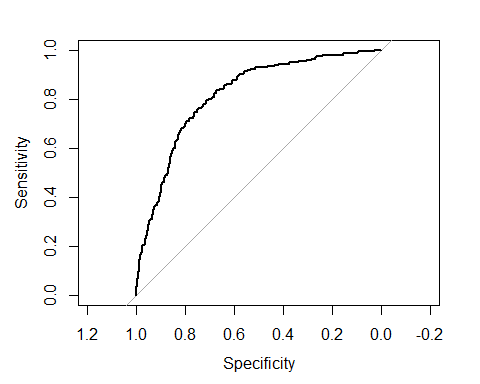
From the above confusion matrix we can see that, **Accuracy** -> 0.869 **Sensitivity** -> 0.9877 **Specificity** -> 0.1667.

#### ROC Curve of the model 1

roc(Churn\_Data\_test\_df$churn, pred\_probs)

##   
## Call:  
## roc.default(response = Churn\_Data\_test\_df$churn, predictor = pred\_probs)  
##   
## Data: pred\_probs in 570 controls (Churn\_Data\_test\_df$churn no) < 96 cases (Churn\_Data\_test\_df$churn yes).  
## Area under the curve: 0.7973

plot.roc(roc(Churn\_Data\_test\_df$churn, pred\_probs))

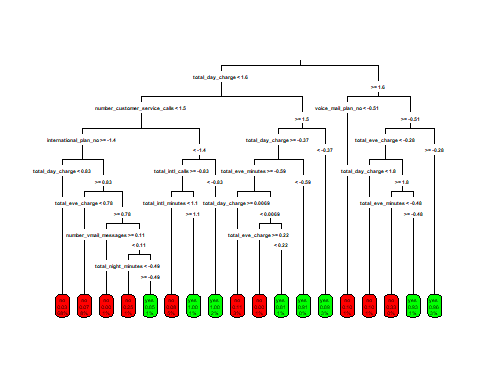
From the above analysis we see Area under curve (AUC) of the model is 0.7973.

**Decision Tree Classifier**

#### Constructing decision tree model on above partitioned data

#### Model Construction

Model\_2 <- rpart(churn ~ ., data = Churn\_Data\_train\_norm, method = "class")  
  
rpart.plot(Model\_2, type = 3, box.palette = c("red", "green"), fallen.leaves = TRUE)



From above, we can see the summary plot of the model where each variable is split into branches or nodes based on **Entropy value**. To use entropy to determine the optimal features is split upon, the algorithm calculates the change in homogeneity that would result from a split on each possible feature which is a measure known as **information gain**.

Predict values using based on Model\_2.

pred\_labels <- predict(object = Model\_2,Churn\_Data\_test\_norm, type = "class")  
pred\_probs <- predict(object = Model\_2,Churn\_Data\_test\_norm)

#### 

#### Performance Metrics

Confusion matrix for significant variable model.

confusionMatrix(pred\_labels,Churn\_Data\_test\_norm$churn)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction no yes  
## no 563 38  
## yes 7 58  
##   
## Accuracy : 0.9324   
## 95% CI : (0.9106, 0.9503)  
## No Information Rate : 0.8559   
## P-Value [Acc > NIR] : 5.285e-10   
##   
## Kappa : 0.6837   
##   
## Mcnemar's Test P-Value : 7.744e-06   
##   
## Sensitivity : 0.9877   
## Specificity : 0.6042   
## Pos Pred Value : 0.9368   
## Neg Pred Value : 0.8923   
## Prevalence : 0.8559   
## Detection Rate : 0.8453   
## Detection Prevalence : 0.9024   
## Balanced Accuracy : 0.7959   
##   
## 'Positive' Class : no   
##

From above confusion metric we can see that, **Accuracy** -> 0.9324 **Sensitivity** -> 0.9877 **Specificity** -> 0.6042

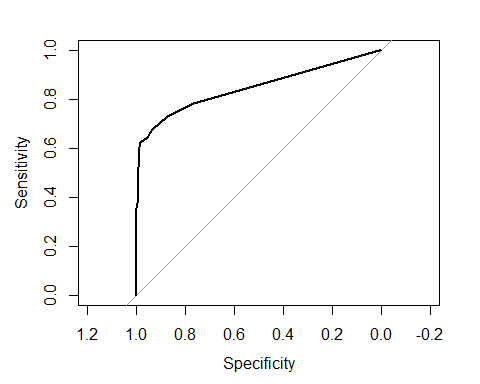
#### 

#### AUC of the model 2

roc(Churn\_Data\_test\_df$churn, pred\_probs[,2])

##   
## Call:  
## roc.default(response = Churn\_Data\_test\_df$churn, predictor = pred\_probs[, 2])  
##   
## Data: pred\_probs[, 2] in 570 controls (Churn\_Data\_test\_df$churn no) < 96 cases (Churn\_Data\_test\_df$churn yes).  
## Area under the curve: 0.847

plot.roc(roc(Churn\_Data\_test\_df$churn, pred\_probs[,2]))

From the above analysis we see Area under curve (AUC) of the model is 0.847.

**Conclusion**

From logistics regression and decision tree models we found that, AUC and accuracy values of decision tree are higher. Hence choosing decision tree as best model for future predictions.

**Predicting Model based on Customers\_To\_Predict data**

load("C:/Users/prajw/Downloads/Customers\_To\_Predict.RData")  
  
Customers\_To\_Predict <- Customers\_To\_Predict %>% select(-state) %>% fastDummies::dummy\_cols(., remove\_selected\_columns = TRUE)  
Customers\_To\_Predict <- as.data.frame(scale(Customers\_To\_Predict))  
predict\_labels <- predict(object = Model\_2, Customers\_To\_Predict, type = "class")  
  
  
Customers\_To\_Predict <- Customers\_To\_Predict %>% mutate(Churn\_Prob = predict\_labels)  
  
table(Customers\_To\_Predict$Churn\_Prob)

##   
## no yes   
## 903 97

We're using a data set that contains a list of customers for whom we need to forecast future churn. We were able to predict that out of 1000 customers 97 customers moving from ABC wireless to another network.

**Contributions**

|  |  |
| --- | --- |
| **NAME** | **CONTRIBUTION** |
| Prajwal C N  pchamben@kent.edu | Model Building, Model Performance, Predictions and Results, Data Cleaning, Data Exploration, Documentation and presentation |
| Kunal Sharma  Ksharm11@kent.edu | Model Building, Model Performance, Predictions and Results, Data Cleaning, Data Exploration, Documentation and presentation |