ABC wireless Inc.

Churn Group Project

MIS 64036 Business Analytics

Courtney Osentoski, Mercy Nani & Monalisa Singh
11-25-2017

	Monalisa	Mercy	Courtney
Project Goal		Mercy summarized this for the project.	
Overview of Data Exploration	We all explored the data.	We all explored the data.	We all explored the data.
Data Cleaning	We all took a part in data cleaning.	We all took a part in data cleaning.	We all took a part in data cleaning.
Model Strategy	This was part of our tournaments we completed for each week and we were all involved in this but Monalisa wrote most of this for the paper.	This was part of our tournaments we completed for each week and we were all involved in this.	This was part of our tournaments we completed for each week and we were all involved in this.
Estimation of Model's Performance	Monalisa helped explain our model performance though the paper. We did check this every time we submitted our model for the tournaments.	We did check this every time we submitted our model for the tournaments.	We did check this every time we submitted our model for the tournaments.
Insights and Conclusion	We all looked through the model to see what insights we could derive from it.	We all looked through the model to see what insights we could derive from it.	We all looked through the model to see what insights we could derive from it. Courtney wrote most of this for the paper.
PowerPoint	We all are helped on this but Monalisa and Mercy contributed more in the construction of the PowerPoint.	We all are helped on this but Monalisa and Mercy contributed more in the construction of the PowerPoint.	We all are helped on this but Monalisa and Mercy contributed more in the construction of the PowerPoint.
Tournament	Monalisa and Courtney helped with the construction of the R code and Mercy helped test it.	Monalisa and Courtney helped with the construction of the R code and Mercy helped test it.	Monalisa and Courtney helped with the construction of the R code and Mercy helped test it.

Telecom Churn Project

Project Goal

Customer churn is when a company loses its customers to its competitors. It is costlier to acquire a new customer than it is to retain the current customer, which is especially true in the telecom business. Most telecom companies suffer from voluntary churn. Companies from this sector often have their customer service department attempt to win back defecting clients, because recovered long-term customers can be worth much more to a company than newly recruited clients.

Churn rate has a strong impact on the lifetime value of the customer because it affects the length of service and the future revenue of the company so being able to predict customers who are likely to churn is valuable. The theme of this project is to enable churn reduction using analytics.

Overview of data, including data exploration analysis and data cleaning

For this data each row represents a subscribing telephone customer. The Churn Train dataset contains 3,333 rows and 20 columns. Each column contains customer attributes such as state, account length, area code, type of plan, number of voicemail messages, total call minutes used during different times of day, charges incurred for services, and whether or not the customer is still a customer. Looking at the head command below we can see that State, Area Code, International Plan, Voicemail Plan, and Churn are not numeric values.

On a closer look at the churn data we noticed a lot of missing values which we initially imputed with KNN.

	state	account_length	area_code	international_plan	voice_mail_plan	number_vmail_messages	total_day_minutes	total_day_calls	total_day_charg
1	NV	0.578151570	area_code_510	no	no	-0.5330444	2.5457683	-0.06645076	-0.212215
2	HI	0.223055900	area_code_415	no	no	-0.5330444	-0.2033275	-0.06645076	2.042049
3	DC	-0.320031595	area_code_415	no	no	-0.5330444	-0.1894367	0.43256706	2.201605
4	HI	-0.299143615	area_code_408	no	yes	1.6478133	-0.4927984	-1.46370066	-1.280592
5	OH	-0.299143615	area_code_415	no	no	-0.5330444	-0.1302013	0.98148666	2.881873
6	MO	-0.173815731	area_code_415	no	no	-0.5330444	-0.3835882	-0.96468284	-0.026785
7	NC	0.787031376	area_code_415	no	no	-0.5330444	-0.3467058	-0.96468284	0.396899
8	PA	-1.447982547	area_code_415	no	no	-0.5330444	-0.4003529	-0.66527215	-0.219761

When analyzing the data, we noticed there were only three distinct area codes associated with all of the various states which doesn't make any sense since all states should have multiple area codes. We dealt with this later on.

```
 n_distinct(ba_imputed$area_code)
L] 3
```

We noticed there were negative values in the number_vmail_messages column which did not make a lot of sense as shown in the picture above.

Calls and minutes were divided as per day,evening,night,international indicating some correlation between them. We planned to run correlation between these attributes to see if we can drop any of the highly correlated columns.

total_day_minutes	total_day_calls	total_day_charge	total_eve_minutes	total_eve_calls	total_eve_charge	total_night_minutes	total_night_cal
2.5457683	-0.06645076	-0.212215152	2.446951757	0.345297467	-0.501443236	0.83483170	-0.41432

We used the library caret and used the nearZeroVar function and found that the number_vmail_messages had the least variance. So we got rid of that column which also helped us neglect the negative values in that column which seemed insignificant.

```
> View(ba_imputed)
> nzv<-nearZeroVar(ba_imputed)
> nzv
[1] 6
>
> ba_imputed$number_vmail_messages<-NULL
```

As you can see below, we removed the number_vmail_messages attribute. This is what our data looks like now.

	state	account_length	area_code	international_plan	voice_mail_plan	total_day_minutes	total_day_calls	total_day_charge	total_eve_minutes	tota
1	NV	0.578151570	area_code_510	no	no	2.5457683	-0.06645076	-0.212215152	2.446951757	
2	HI	0.223055900	area_code_415	no	no	-0.2033275	-0.06645076	2.042049477	-0.322241198	
3	DC	-0.320031595	area_code_415	no	no	-0.1894367	0.43256706	2.201605271	-0.447503056	
4	HI	-0.299143615	area_code_408	no	yes	-0.4927984	-1.46370066	-1.280592123	-0.443129824	
5	OH	-0.299143615	area_code_415	no	no	-0.1302013	0.98148666	2.881873554	-0.302561654	
6	MO	-0.173815731	area_code_415	no	no	-0.3835882	-0.96468284	-0.026785446	-0.294627361	
7	NC	0.787031376	area_code_415	no	no	-0.3467058	-0.96468284	0.396899872	-0.310058623	
8	PA	-1.447982547	area_code_415	no	no	-0.4003529	-0.66527215	-0.219761710	-0.507791182	

As we can see here, KNN imputation shows a lot of negative values for most of the attributes in the dataframe. We plan to deal with this using the median impute method.
We read the the data frame again, removed the number_vmail_messages and ran the median impute. This is what our data looks like now.

	state	account_length	area_code	international_plan	voice_mail_plan	total_day_minutes	total_day_calls	total_day_charge	total_eve_minutes	tota
1	NV	125	area_code_510	no	no	2013.4	99	28.66	1107.6	
2	HI	108	area_code_415	no	no	291.6	99	49.57	221.1	
3	DC	82	area_code_415	no	no	300.3	109	51.05	181.0	
4	HI	100	area_code_408	no	yes	110.3	71	18.75	182.4	
5	OH	83	area_code_415	no	no	337.4	120	57.36	227.4	
6	MO	89	area_code_415	no	no	178.7	81	30.38	209.9	
7	NC	135	area_code_415	no	no	201.8	81	34.31	225.0	
8	PA	28	area_code_415	no	no	168.2	87	28.59	161.7	

howing 1 to 9 of 3,333 entries

Console D:/RStudio/MonaProjects/

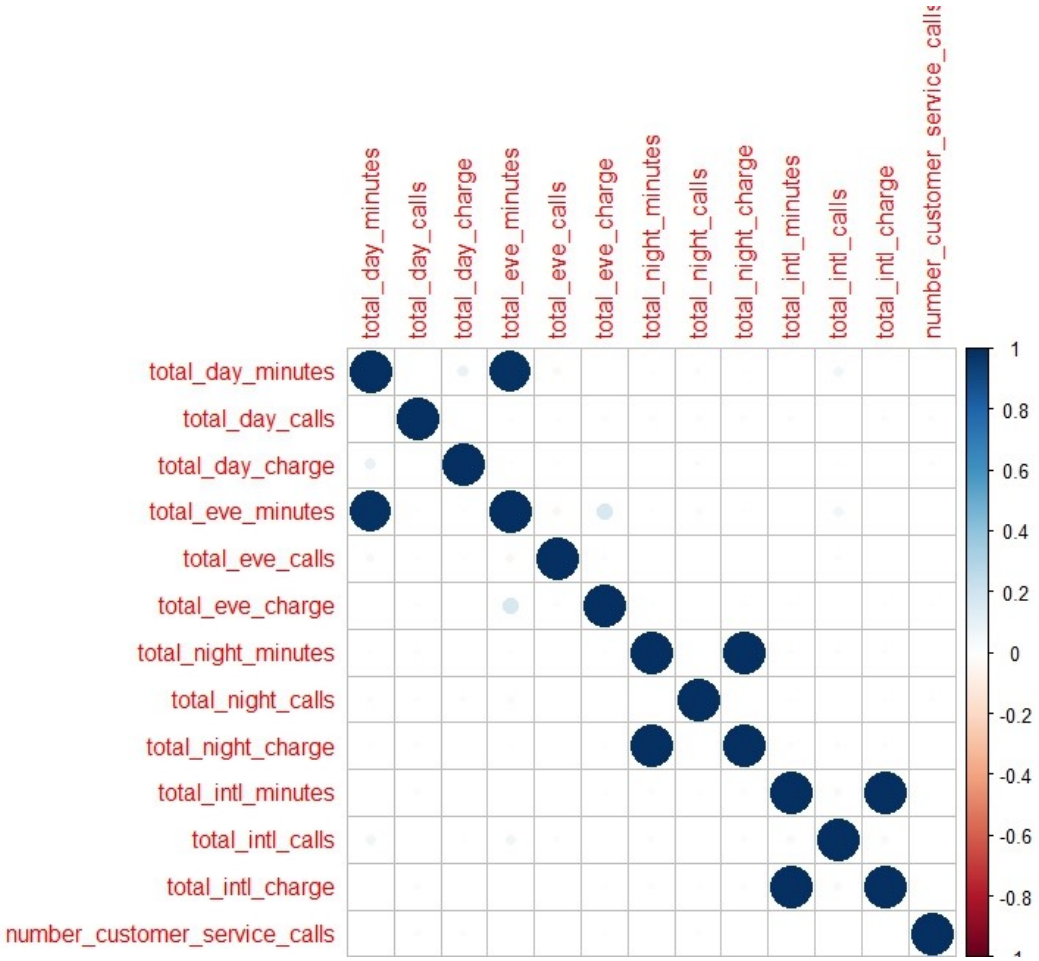
```
[1] 3
> preproc<-preProcess(ct, method = c("medianImpute"))
> ba_imputed<-predict(preproc,ct)
> summary(ba_imputed)
     state       account_length            area_code    international_plan voice_mail_plan
 WV     : 106   Min.   :-209.00   area_code_408: 838   no :3010           no :2411
 MN     :  84   1st Qu.:  77.00   area_code_415:1655   yes: 323           yes: 922
 NY     :  83   Median : 100.00   area_code_510: 840
 AL     :  80   Mean   :  97.72
 OH     :  78   3rd Qu.: 122.00
 OR     :  78   Max.   : 243.00
 (Other):2824
```

Our data frame looks a lot better here.

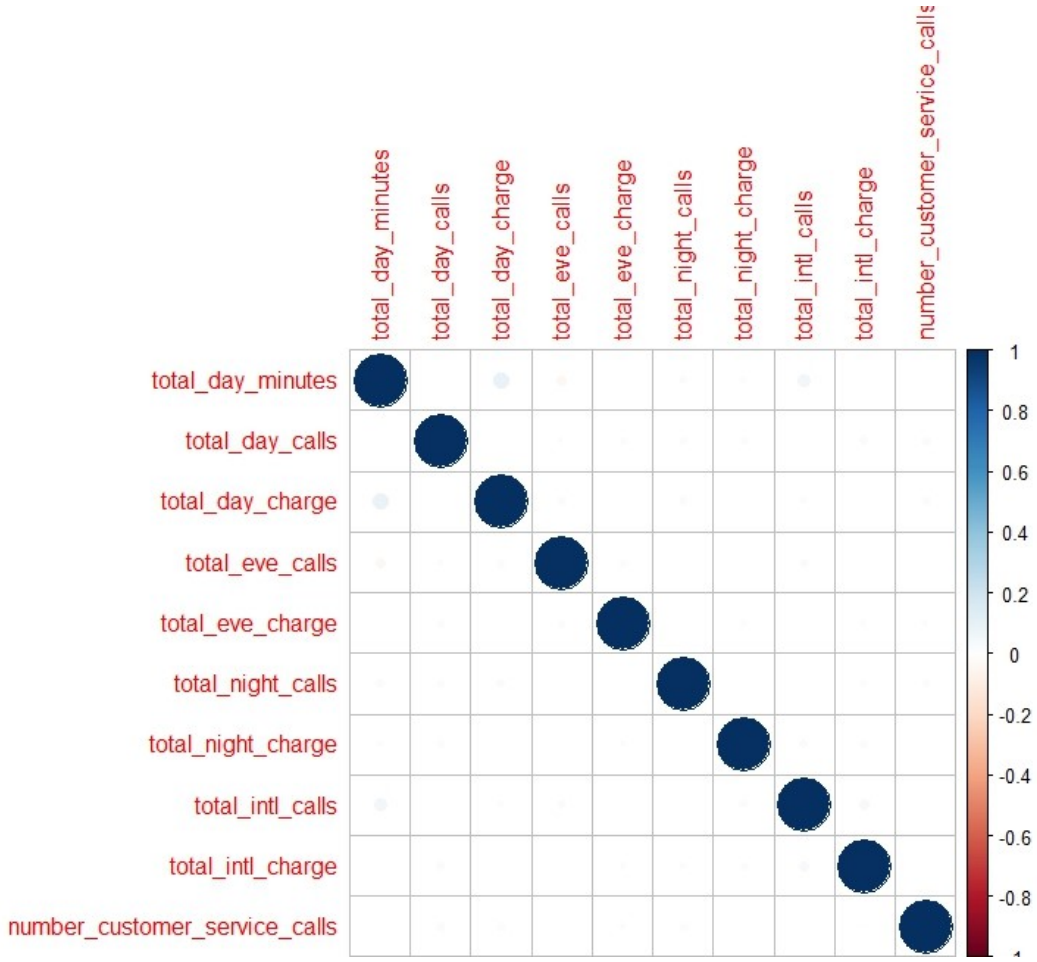
For building our first model, this is the data set we used. We also realised that the data cleaning will be done simultaneously according to the different models we make and the significance they give for the various models.
When working on our second Decision Tree model we looked at the correlation of all the numeric columns in the data set of the first Decision Tree. As you can see below there are some columns that are highly correlated to each other based on the dark blue circles:

1. Total_eve_minutes and Total_day_minutes
2. Total_night_minutes and Total_night_charge
3. Total_intl_minutes and Total_intl_charge

Based off of this graph, we decided to remove the total_eve_minutes, total_night_minutes, and total_intl_minutes columns.



Once we removed these columns, we ran another corrplot on the remaining numeric columns to see if any other columns were correlated. We can see from below total_day_minutes and total_day_charge are slightly correlated based on the small light blue circles. We removed total_day_minutes and the accuracy of our model did not change so we kept all of these correlated columns removed for the Final Decision Tree model.


total_day_minutes
total_day_calls
total_day_charge
total_eve_calls
total_eve_charge
total_night_calls
total_night_charge
total_intl_calls
total_intl_charge
number_customer_service_calls
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8

Building the Initial models.

1. The first model we built was using the data frame with only the number_vmail_messages not attribute not present in the dataframe. It gave us the following results.

This model was a J48 decision tree model which we created using the RWeka package in R. We used this model to see how well the instances were being correctly classified and to see the RMSE model.

The good part about this model was that it gave a lot of parameters including the confusion matrix.

```
> ctf_Impnzv<-nearZeroVar(ctf_Imp)
> ctf_Impnzv
[1] 6
> ctf_Imp$number_vmail_messages<-NULL
> modelv<-J48(churn~., data=ctf_Imp)
> summary(modelv)

=== Summary ===

Correctly Classified Instances        3174               95.2295 %
Incorrectly Classified Instances       159                4.7705 %
Kappa statistic                          0.7801
Mean absolute error                      0.0861
Root mean squared error                  0.2075
Relative absolute error                 34.7307 %
Root relative squared error             58.9508 %
Total Number of Instances             3333

=== Confusion Matrix ===

    a    b   <-- classified as
 2843    7 |    a = no
  152  331 |    b = yes
```

We combined State, area_code and account_length into one attribute called “Location_acclength” and removed these independent attributes to see what effect it creates on the model.

```
=== Summary ===

Correctly Classified Instances        3204               96.1296 %
Incorrectly Classified Instances       129                3.8704 %
Kappa statistic                          0.8267
Mean absolute error                      0.073
Root mean squared error                  0.1911
Relative absolute error                 29.4492 %
Root relative squared error             54.2837 %
Total Number of Instances             3333

=== Confusion Matrix ===

    a    b   <-- classified as
 2843    7 |    a = no
  122  361 |    b = yes
```

We can see here that the correctly classified instances increased in the model.

2. The next model was a linear model. We built this model using the same data as in the J48 model and it gave us decent results.

```
> library(ISLR)
> ctf_Imp3<-ctf_Imp2
> ctf_Imp3$churn<-as.numeric(ctf_Imp3$churn)
> modelv3<-lm(churn~., data=ctf_Imp3)
> summary(modelv3)

Call:
lm(formula = churn ~ ., data = ctf_Imp3)

Residuals:
    Min      1Q  Median      3Q     Max
-0.6492  0.0000  0.0000  0.0000  1.0317

Coefficients:
                                          Estimate Std. Error t value Pr(>|t|)
(Intercept)                              4.629e-01  3.580e-01   1.293   0.1965
international_planyes                    2.769e-01  4.658e-02   5.945 4.85e-09 ***
voice_mail_planyes                      -7.583e-02  2.947e-02  -2.573   0.0103 *
total_day_minutes                        5.895e-04  6.044e-04   0.975   0.3298
total_day_calls                          3.172e-04  8.070e-04   0.393   0.6944
total_day_charge                        -1.214e-04  3.568e-03  -0.034   0.9729
total_eve_minutes                       -1.262e-03  1.214e-03  -1.039   0.2992
total_eve_calls                         -2.597e-04  8.004e-04  -0.325   0.7457
total_eve_charge                         2.359e-02  1.430e-02   1.650   0.0995 .
total_night_minutes                     -2.015e-01  2.139e-01  -0.942   0.3465
total_night_calls                        7.549e-04  6.920e-04   1.091   0.2758
total_night_charge                       4.488e+00  4.753e+00   0.944   0.3454
total_intl_minutes                       1.689e+00  1.386e+00   1.219   0.2234
total_intl_calls                        -8.079e-03  6.519e-03  -1.239   0.2158
total_intl_charge                       -6.194e+00  5.134e+00  -1.207   0.2281
number_customer_service_calls            6.715e-02  1.062e-02   6.325 5.17e-10 ***
Location_acclengthAK_area_code_408_100   1.988e-02  3.740e-01   0.053   0.9576
Location_acclengthAK_area_code_408_110  -1.549e-01  4.612e-01  -0.336   0.7372
Location_acclengthAK_area_code_408_127  -8.694e-02  4.581e-01  -0.190   0.8495
Location_acclengthAK_area_code_408_141   4.482e-02  4.592e-01   0.098   0.9223
Location_acclengthAK_area_code_408_36    7.165e-02  4.631e-01   0.155   0.8771
Location_acclengthAK_area_code_408_50    1.536e-02  4.585e-01   0.034   0.9733
Location_acclengthAK_area_code_408_52   -1.196e-01  4.588e-01  -0.261   0.7944
```

```
Residual standard error: 0.3217 on 563 degrees of freedom
Multiple R-squared:  0.859,     Adjusted R-squared:  0.1652
F-statistic: 1.238 on 2769 and 563 DF,  p-value: 0.0007463

> anova(modelv3)
Analysis of Variance Table

Response: churn
                              Df  Sum Sq Mean Sq  F value    Pr(>F)
international_plan             1  27.887 27.8874 269.5251 < 2.2e-16 ***
voice_mail_plan                1   4.442  4.4423  42.9334 1.280e-10 ***
total_day_minutes              1   0.038  0.0380   0.3671 0.5448317
total_day_calls                1   0.067  0.0665   0.6429 0.4229971
total_day_charge               1  14.277 14.2766 137.9802 < 2.2e-16 ***
total_eve_minutes              1   3.709  3.7085  35.8420 3.814e-09 ***
total_eve_calls                1   0.002  0.0024   0.0234 0.8784597
total_eve_charge               1   0.053  0.0531   0.5132 0.4740531
total_night_minutes            1   0.813  0.8132   7.8596 0.0052294 **
total_night_calls              1   0.000  0.0000   0.0000 0.9964533
total_night_charge             1   0.035  0.0351   0.3396 0.5602836
total_intl_minutes             1   1.315  1.3146  12.7052 0.0003956 ***
total_intl_calls               1   1.126  1.1258  10.8808 0.0010330 **
total_intl_charge              1   0.045  0.0448   0.4330 0.5108040
number_customer_service_calls  1  18.446 18.4460 178.2758 < 2.2e-16 ***
Location_acclength          2754 282.499  0.1026   0.9914 0.5583243
Residuals                    563  58.253  0.1035
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We used the linear model to see if there is any linearity explained by the attributes. However, we decided to try some more models to get a better estimate from the data.

3. We built a logistic regression model for a better estimate of performance of the model. For this model we removed the "Location_accountlength" attribute too because they had too many levels in them as we can see in the picture above but we kept area_code and account_length.

```
> Predictedd_Values<-predict (churnmodel, newdata= ctf2_imp, type="response")
> roc(ctf2_imp$churn, Predictedd_Values)

Call:
roc.default(response = ctf2_imp$churn, predictor = Predictedd_Values)

Data: Predictedd_Values in 2850 controls (ctf2_imp$churn no) < 483 cases (ctf2_imp$churn yes).
Area under the curve: 0.8166
>
```

4. Our **FINAL** model was a Decision Tree model because it gave us the Probabilities of Yes and No which is exactly what we needed. We built a classification tree for this problem.
    a. Our first Decision Tree model was based on the same data frame as the logistic regression model. This is the one we submitted for the Tournament1.

```
 confmatrix <- print(table(ctf1$predict, ctf1$churn, dnn=c("Predicted", "Actual")))
         Actual
redicted   no  yes
     no  2813  177
     yes   37  306
 Accuracy <- print((confmatrix[2,2]+confmatrix[1,1])/sum(confmatrix) * 100)
1] 93.57936
 Sensitivity<-print(confmatrix[2,2]/(confmatrix[2,2]+confmatrix[1,2])*100)
1] 63.35404
 Specificity<-print(confmatrix[1,1]/(confmatrix[1,1]+confmatrix[2,1])*100)
1] 98.70175
```

b. Our second Decision Tree model was based on the same data from the first Decision Tree model plus some removed the columns day, eve, night and intl minutes based on our correlation analysis, as shown above. We used this for the Tournament Test 2.

```
> confmatrix <- print(table(ctf_Imp_cor5$predict2, ctf_Imp_cor5$churn, dnn=c("Predicted", "Actual")))
         Actual
Predicted   no  yes
      no  2813  177
      yes   37  306
> corrplot(cor(ctf_Imp_cor5[,5:13]))
> Accuracy <- print((confmatrix[2,2]+confmatrix[1,1])/sum(confmatrix) * 100)
[1] 93.57936
> Sensitivity<-print(confmatrix[2,2]/(confmatrix[2,2]+confmatrix[1,2])*100)
[1] 63.35404
> Specificity<-print(confmatrix[1,1]/(confmatrix[1,1]+confmatrix[2,1])*100)
[1] 98.70175
```

c. Our third and our **FINAL** decision tree model had a little variation. For this one, we reloaded the data set and did the median impute and NearZeroVar and then removed the number_vmail_messages, (day,eve,night,intl)minutes and the area_code and kept the state attribute. We did this because the area_code were only of 3 distinct types for all states as mentioned previously so it would be better to identify the customers likely to churn according to the states.

```
> ctf_Imp_cor5$predict2 <- predict(treeee, data = ctf_Imp_cor5, type ="class")
> View(ctf_Imp_cor5)
> confmatrix <- print(table(ctf_Imp_cor5$predict2, ctf_Imp_cor5$churn, dnn=c("Predicted", "Actual")))
         Actual
Predicted   no  yes
      no  2835  162
      yes   15  321
> Accuracy <- print((confmatrix[2,2]+confmatrix[1,1])/sum(confmatrix) * 100)
[1] 94.68947
> Sensitivity<-print(confmatrix[2,2]/(confmatrix[2,2]+confmatrix[1,2])*100)
[1] 66.45963
> Specificity<-print(confmatrix[1,1]/(confmatrix[1,1]+confmatrix[2,1])*100)
[1] 99.47368
>
```

Estimation of Model's performance

1. In the J48 model we saw 96% correctly classified instances. However, that did not give us any insight about the model's performance. It was surprising to see such a high correctly classified rate because the data we used for the J48 had minimal data cleaning done over it. The result looked too good to be true.

2. In the Linear model, we performed the analysis of variance as shown below.

```
Residual standard error: 0.3217 on 563 degrees of freedom
Multiple R-squared:  0.859,     Adjusted R-squared:  0.1652
F-statistic: 1.238 on 2769 and 563 DF,  p-value: 0.0007463

> anova(modelv3)
Analysis of Variance Table

Response: churn
                              Df  Sum Sq Mean Sq  F value    Pr(>F)
international_plan             1  27.887 27.8874 269.5251 < 2.2e-16 ***
voice_mail_plan                1   4.442  4.4423  42.9334 1.280e-10 ***
total_day_minutes              1   0.038  0.0380   0.3671 0.5448317
total_day_calls                1   0.067  0.0665   0.6429 0.4229971
total_day_charge               1  14.277 14.2766 137.9802 < 2.2e-16 ***
total_eve_minutes              1   3.709  3.7085  35.8420 3.814e-09 ***
total_eve_calls                1   0.002  0.0024   0.0234 0.8784597
total_eve_charge               1   0.053  0.0531   0.5132 0.4740531
total_night_minutes            1   0.813  0.8132   7.8596 0.0052294 **
total_night_calls              1   0.000  0.0000   0.0000 0.9964533
total_night_charge             1   0.035  0.0351   0.3396 0.5602836
total_intl_minutes             1   1.315  1.3146  12.7052 0.0003956 ***
total_intl_calls               1   1.126  1.1258  10.8808 0.0010330 **
total_intl_charge              1   0.045  0.0448   0.4330 0.5108040
number_customer_service_calls  1  18.446 18.4460 178.2758 < 2.2e-16 ***
Location_acclength          2754 282.499  0.1026   0.9914 0.5583243
Residuals                    563  58.253  0.1035
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This model looked pretty good with an 85% R square and most of the attributes were significant as shown by the p-value. We still felt the need for a better model as we were not able to use this model over our test set and it gave no idea about the customers who were likely to churn.

3. The logistic regression model gave a decent estimate of the performance.

```
> Predictedd_Values<-predict (churnmodel, newdata= ctf2_imp, type="response")
> roc(ctf2_imp$churn, Predictedd_Values)

Call:
roc.default(response = ctf2_imp$churn, predictor = Predictedd_Values)

Data: Predictedd_Values in 2850 controls (ctf2_imp$churn no) < 483 cases (ctf2_imp$churn yes).
Area under the curve: 0.8166
>
```

The Area Under The Curve was .8166 which was good but not good enough. We needed a better estimate for our churn prediction.

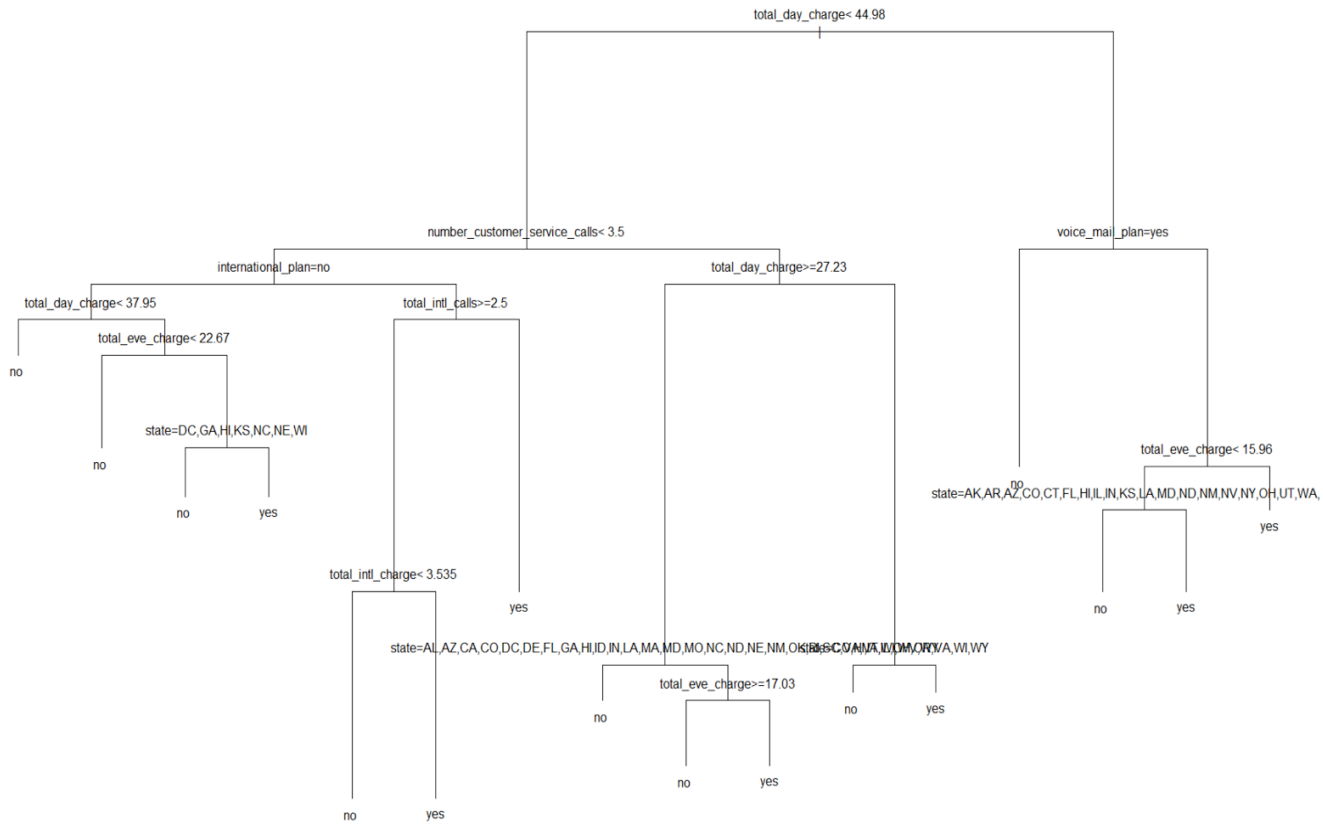
4. For our **Final Decision Tree model**, the estimate for the Area Under the Curve came out pretty good with all the changes that we did in data(removing area_code and keeping state, etc) and also got a higher estimate of accuracy.

```
> roc(ctf_Imp_cor5$churn, ctf_Imp_cor5$predict1[,2])

Call:
roc.default(response = ctf_Imp_cor5$churn, predictor = ctf_Imp_cor5$predict1[,     2])

Data: ctf_Imp_cor5$predict1[, 2] in 2850 controls (ctf_Imp_cor5$churn no) < 483 cases (ctf_Imp_cor5$churn yes).
Area under the curve: 0.8885
```

We got the probabilities of yes for churn using this model when we tested it on the data that was available. Also, plotting the decision tree gave us some insights about the data which the other models unable to which we will discuss in the next section.



Insights and conclusions

Our decision tree above can give us some interesting insights into seeing which variables are helpful to predict if a customer is going to churn. There are 71 nodes, so we will only go through the first few.

- If a customer has a total day charge > $44.98 and has a voicemail plan, then they are not going to churn. This contains 52 observations.
- If a customer has a total day charge > $44.98, has no voicemail plan, and has total evening charge > $15.96 will churn. This contains 96 observations.
- If a customer has a total day charge > $44.98, has no voicemail plan, has total evening charge < $15.96 will churn, and lives in AK,AR,CO,CT,FL,HI,IL,IN,KS,LA,MD,ND,NM,NV,NY,OH,UT,WA then they will not churn. This contains 84 observations.
- If a customer has a total day charge > $44.98, has no voicemail plan, has total evening charge <$15.96 will churn, and does not lives in AK,AR,CO,CT,FL,HI,IL,IN,KS,LA,MD,ND,NM,NV,NY,OH,UT,WA then they will churn. This contains 58 observations.
- If a customer has a total day charge < $44.98, number of customer service calls < 3.5, no international plan, and total day charge < $37.95 then they are not going to churn. This contains 2,262 observations.

Variable importance

total_day_charge	number_customer_service_calls	state	total_eve_charge
26	14	13	10
international_plan	total_intl_charge	total_intl_calls	voice_mail_plan
9	9	8	7
total_night_calls	total_night_charge	account_length	
2	1	1	

By looking at the summary, we can determine that total_day_charge is the most important variable. Variable important for regression trees is based on the primary splits and the surrogate splits. The variable importance below adds up to 100. Total_day_charge, number_customer_service_calls, state, and total_eve_charge are the four top variables in our model to predict churn.

Our decision tree model has been our best model throughout this project of predicting churn. We have accomplished this through data exploration and data cleaning throughout the course. By completing the three tournaments our model has been improved. Our final model has an AOC of .8885 and an accuracy of 94.69%. Based on our discussions in class, this model performs well. *ABC Wireless Inc.* can predict the churn of their current customers with high confidence in our model. This will save them money from trying to recruit new customers and be able to retain their current customers.