Customer Churn

Predictions

Susan Kight
7-11-2019

Descriptive Analysis

The churn dataset supplied was reviewed and descriptive analytics performed to prepare appropriately for modeling. The distributions of each feature against the target were explored as well as correlations between features.

Distribution of Each Feature in Dataset

Brief exploratory analysis was requested and therefore descriptions of each feature have been provided below that guided that cleaning and transformations that were performed during the pre-processing of each model. Appendix 1 has all graphs related to the distribution of each feature.

The target feature 'result' shows an overwhelming proportion of customers stay and only a much smaller frequency 'LEAVE'. This smaller frequency are the churners that we want to predict and change their final decision on so that they decide to remain with the company instead. They would be incentivized with the $100 discount. Approximately 9.5% of the data is marked as churn (LEAVE).

With COLLEGE, there is an equal balance of those who are college educated and those who are not. But we are seeing greater proportional frequency for those that 'STAY' which is not surprising given the low volume of those that 'LEAVE' in this dataset.

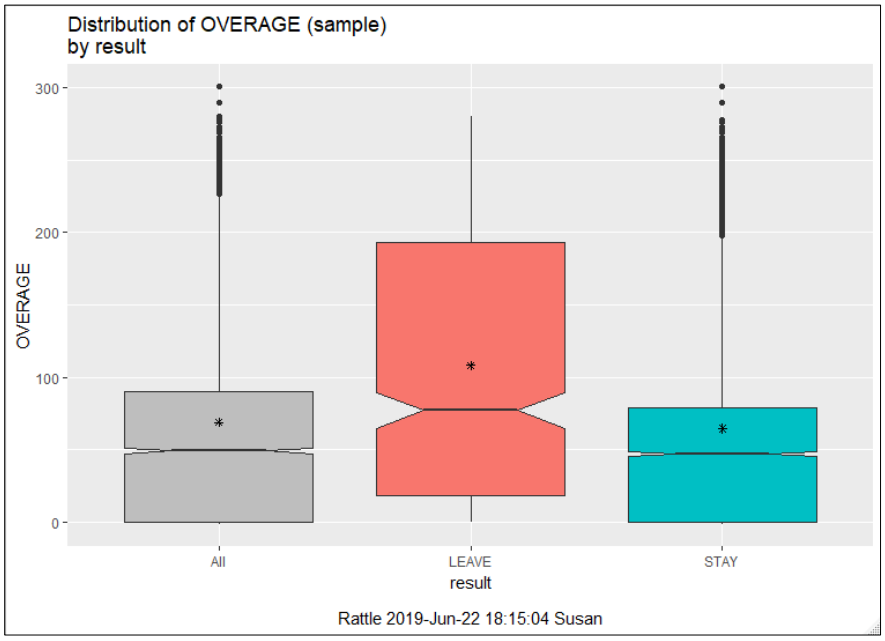
REPORTED_SATISFACTION shows large frequency for 'very_unsat' and the smallest frequency for those who are 'sat'. There is very similar proportion of those wanting to 'LEAVE' irrespective of reported satisfaction, only 'avg' sees a lower proportion for 'LEAVE'. This is quite concerning as logically you would expect anyone very satisfied to have the lowest desire to leave out of all the population.

USAGE_LEVEL is another feature showing similar proportions for 'LEAVE' despite the level of usage. There is an exception for 'very_high 'which has the lowest 'LEAVE' proportion. If customers are actively using the service, they may be more likely to be stay as it works for them. It is important to note that this is a self-reported usage level.

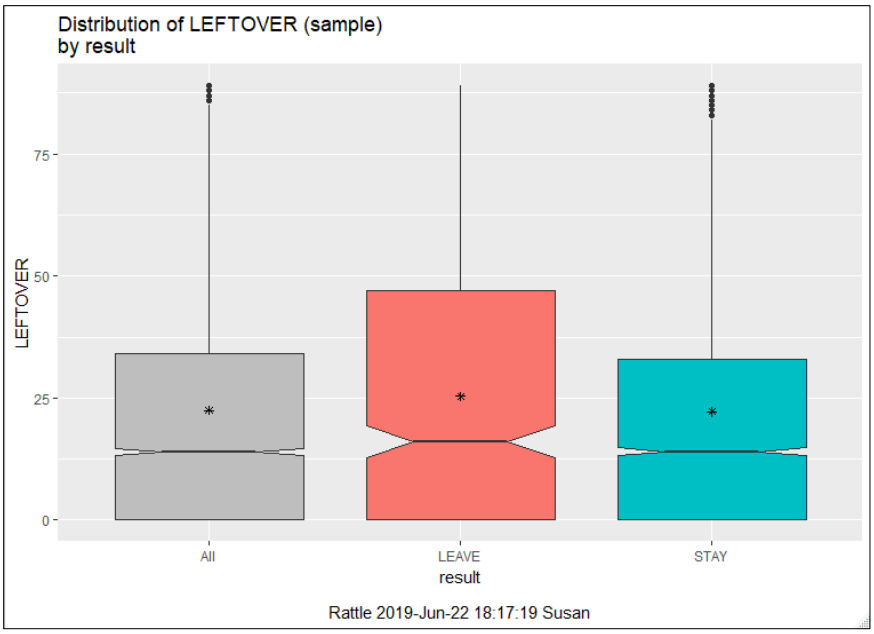
CONSIDERING_CHANGE_OF_PLAN has the largest frequency for 'considering' and this level together with 'no' and 'perhaps' has the greatest proportions of 'LEAVE' compared to the levels which is interesting. Logically you would not expect to see one of the largest proportions to leave for a group of the population not even considering changing their plan. Would be good to learn more about this feature in terms of how it was asked to customers and how far in advance of a contract ending.

INCOME has a skewed distribution as one would expect and will need to be transformed by natural log before running certain models such as logistic regression. Those who have higher ICNCOME have a greater likelihood for 'LEAVE'. This feature appears to be annual income.

OVERAGE gives data on average overcharges per month and has a skewed distribution that will need transforming by natural log during pre-processing for certain models. With greater OVERAGE we are seeing a greater proportion 'LEAVE'. What's interesting to see about this feature is that there are some outliers at the higher end values of the distribution that will also need to be handled by replacing with the median during pre-processing stage prior to running certain models so they are not impacted by these.



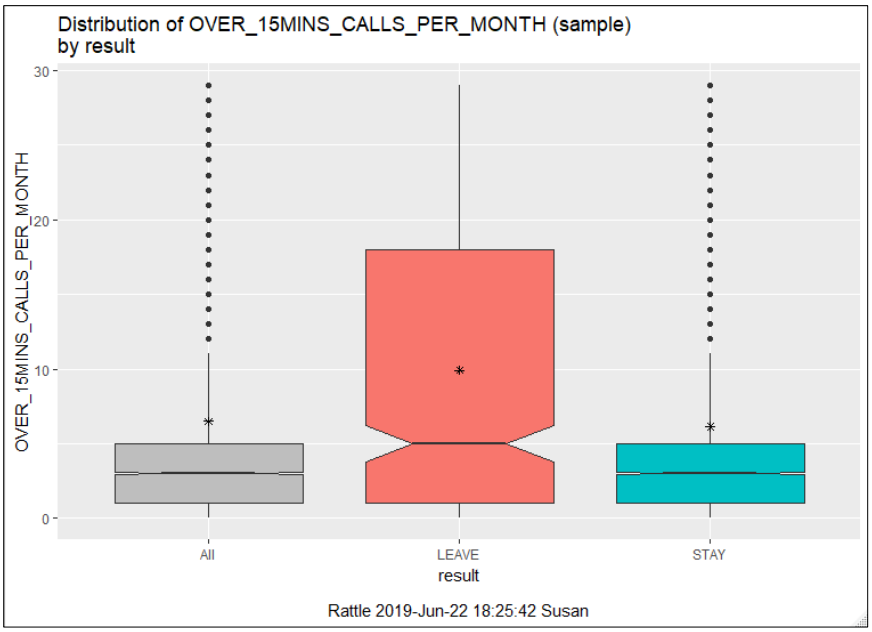
LEFTOVER gives data on average percentage leftover minutes per month and is another numeric feature with a skewed distribution. The larger the LEFTOVER, the more likely they are to 'LEAVE'. This feature will need to be transformed by a natural log during pre-processing for certain numeric models. LEFTOVER also appears to have outliers at the high end of its distribution that will need to be resolved by replacing with the median prior to running specific models so they are not impacted by these.



HOUSE shows a single peak and then levels off within its distribution. The smaller values of HOUSE, the greater proportion 'LEAVE'. This will need to be transformed by natural log before running logistic regression and other numeric models.

HANDSET_PRICE has a skewed distribution and shows for a large proportion of the higher HANDSET_PRICE it tends to have customers more likely to 'LEAVE'. This will need to be transformed by natural log before running logistic regression and other numeric models.

OVER_15MINS_CALLS_PER_MONTH gives the average number of greater than 15 mins calls per month shows a very unusual distribution with two strong peaks for the lower values and then suddenly dropping and leveling off from 10 onwards. Overall from 7 onwards there is a greater tendency to 'LEAVE'. There is a skewed distribution being displayed and therefore will need to be transformed by natural log before running logistic regression and other numeric models. This feature has several outliers that will need to be handled by replacing with this median so they do not negatively impact some of the model I will be performing.



AVERAGE_CALL_DURATION is another feature displaying a skewed distribution and will therefore need to be transformed by natural log as mentioned for some of the other numeric features. Only at a small number of specific points does 'LEAVE' perform proportionally higher than 'STAY'.

It is important to note that for the numeric features above, they will require rescaling 0 to 1 for some of the models such as ANN. I will discuss the pre-processing for each model within the specific model sections.

There were no missing values in the original data.

Correlated Features

To complete the descriptive analysis of the features within the churn dataset, I did review for correlation. The below are the results of those that are highly correlated, for the correlation matrix please see appendix 2:

- OVERAGE with OVER_15MINS_CALLS_PER_MONTH = 0.750767521
- LEFTOVER with AVERAGE_CALL_DURATION = -0.633073112
- HANDSET_PRICE with INCOME = 0.729021615

During modeling, I will attempt running each model numerous times to see which one feature from each of the 3 bullets is more beneficial to remove by reviewing the validation error to which has the lowest

value. Not all variations of pre-processing will be discussed in this report and only the final model for each model type will be described in depth and have its results presented.

Modeling

For each model type, the following were used to ensure each model could be reviewed and compared fairly:

- Partitioned: 70/15/15
- Seed: 61

Also, to make target easier to understand it was changed to the leave result displaying values as Yes (Churn) and No for those that were STAY.

Each model type had its own pre-processing performed which will be described in each of the sections below.

Logistic Regression

The pre-processing for the logistic regression model included outliers being replaced with median for LEFTOVER, OVERAGE and OVER_15MINS_CALLS_PER_MONTH. All numeric features were transformed by natural log. Once transformed, any features (OVERAGE, LEFTOVER and OVER_15MINS_PER_CALLS_PER_MONTH with missing values discovered were replaced with the median where there was still a slight difference in mean and median. Final step included placing all numeric features on the same scale using 0 to 1.

Please note, no features were excluded from the original dataset as I included all in the initial model and then optimized by removing statistically insignificant features as the tuning parameter. The results from the initial model are shown in appendix 3 for how each feature performed with its P-value for statistical significance. The initial model performed 9.4% overall error's and had an AUC of 0.5953 in the validation set.

In reviewing the initial model's results, I then removed the statistically insignificant features that were COLLEGE, REPORTED_SATISFACTION, REPORTED_USAGE_LEVEL, CONSIDERING_CHANGE_OF_PLAN, HANDSET_PRICE, AVERAGE_CALL_DURATION, and LEFTOVER. The optimized model was ran and results show statistical significance for all remaining features – please see appendix 4.

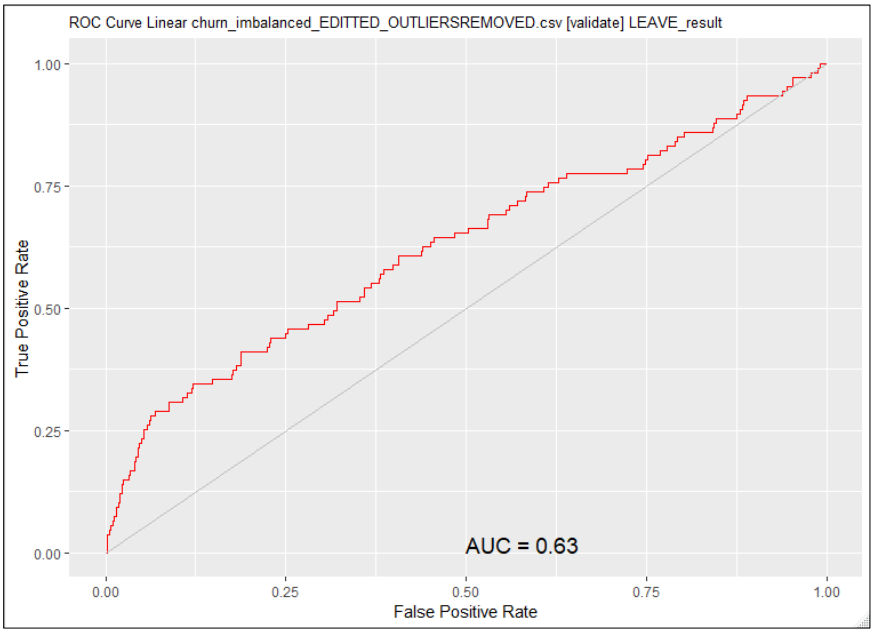
The coeffient values on the optimized model are shown below together with explainations:

```
Coefficients:
                                          Estimate Std. Error z value Pr(>|z|)
(Intercept)                               -0.41217    1.54868  -0.266  0.79013
IMD_RLG_INCOME                             0.27102    0.08451   3.207  0.00134 **
IMD_RLG_HOUSE                             -0.61242    0.08432  -7.263 3.79e-13 ***
IMD_IMD_RLG_OVERAGE                        0.58272    0.11484   5.074 3.89e-07 ***
IMD_IMD_RLG_OVER_15MINS_CALLS_PER_MONTH    0.26845    0.06032   4.451 8.57e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- As INCOME increases, the probability of churning ('Yes' to leave) goes up 0.27.
- As HOUSE increases, the probability of churning goes down.
- As OVERAGE increases, the probability of churning goes up by the most significant amount (0.58).
- As OVER_15MINS_PER_CALLS_PER_MONTH increases, the probability of churning goes up.

The errors were then reviewed on the model for the validation set. This came to 9.4% overall validation error. I did notice on the validation confusion matrix that no predictions were being made for yes. See appendix 5 for the confusion matrix on the validation set. I did compare the number of Yes predictions with some of the other versions of the model that I ran when experimenting with different pre-processing (e.g. not fixing outliers) and the validation error was worse with still no positive predictions being made.

Validation AUC and ROC curve were also reviewed. The AUC was 0.6267 and the ROC curve is shown below:



The validation AUC value is better than random guessing as it is above 0.5. It is okay value overall but would have preferred a higher number to indicate a higher quality of the classifier. The ROC plot does show some early signs of strong performance with a steep climb up but then it does taper off as false positive rate increases.

Decision Tree

The pre-processing for the decision tree model type included removing INCOME and OVERAGE which are highly correlated features. Most of the original data was used as it was, and I checked for no missing values for which there were not any. Knowing the tree would initially be over fitted and then pruned back using the results of the model to find the lowest xerror and using its associated complexity for the resulting optimal tree, I felt confident with this approach for parameter tuning. Also knowing the tree would select based on information gain and entropy I was not concerned about the splits it would make. I did attempt several other pre-processing methods, but this performed best with the lowest validation error initially and within the resulting optimized pruned back decision tree.

The settings for the tree were min. split 0, max. depth 30, min. bucket 1, and complexity at 0. The below displays the results of the initial overfitted tree. The initial tree had an error rate of 15.7% and AUC 0.5393 on the validation set. Irrespective of the pre-processing attempted all decision tree models had the lowest xerror at the root node.

```
Root node error: 504/5307 = 0.094969

n= 5307

          CP nsplit rel error xerror     xstd
1  0.00308642      0 1.0000000 1.0000 0.042376
2  0.00297619     15 0.9325397 1.0575 0.043446
3  0.00277778     35 0.8670635 1.0655 0.043590
4  0.00238095     41 0.8492063 1.0794 0.043841
5  0.00198413     61 0.7956349 1.2143 0.046168
6  0.00182540    113 0.6904762 1.2262 0.046364
7  0.00165344    181 0.5357143 1.2817 0.047261
8  0.00158730    193 0.5099206 1.3512 0.048342
9  0.00148810    262 0.3690476 1.3671 0.048583
10 0.00132275    307 0.2916667 1.4643 0.050013
11 0.00113379    330 0.2599206 1.5218 0.050824
12 0.00099206    338 0.2500000 1.6706 0.052809
13 0.00079365    429 0.1488095 1.7063 0.053263
14 0.00076313    466 0.1130952 1.7143 0.053363
15 0.00066138    487 0.0892857 1.7679 0.054025
16 0.00056689    563 0.0198413 1.7679 0.054025
17 0.00049603    575 0.0079365 1.7738 0.054098
18 0.00001000    586 0.0000000 1.7738 0.054098
```

As a result, the usual strategy of minimizing xerror is not that valuable to prune the tree back. Instead, I gently increased complexity as the tuning parameter and looked for the lowest validation error to find an optimal tree. The below are the results of the various attempts.

Complexity	Validation Error	Validation AUC	Notes
0.002	10%	0.5779	Tree too complex to interpret
0.0021	10%	0.5779	Tree too complex to interpret

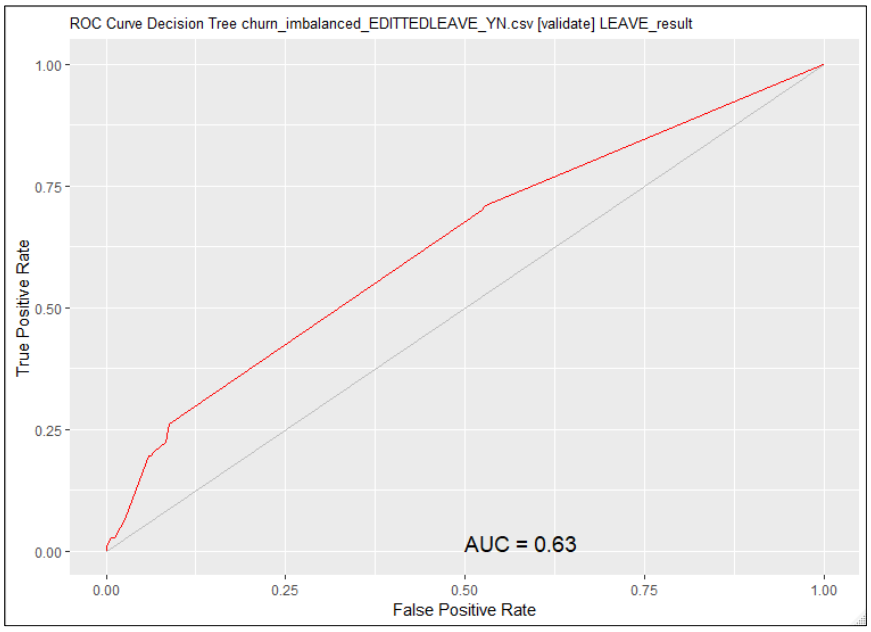
0.0028	10.1%	0.5849	Tree too complex to interpret
0.0029	10.1%	0.5849	Tree too complex to interpret
0.003	**10.2%**	**0.6278**	**Tree is interpretable and has leaf nodes for each class**
0.0031	9.4%	0.5000	No splits on the tree
0.0032	9.4%	0.5000	No splits on the tree
0.004	9.4%	0.5000	No splits on the tree
0.005	9.4%	0.5000	No splits on the tree

At the various complexities, despite the validation error being lowest at the higher complexities attempted, there were no splits on the decision tree, or it was too complex to interpret so therefore they could not be used. In the end the optimal tree was selected as the only interpretable tree and for the highest AUC – complexity of 0.003.

The validation error for the tree is given above (10.2%), to see the confusion matrix please see appendix 6. For the decision tree plot, see please appendix 7. Classification rules from the decision tree for Yes leaf nodes include:

1. HOUSE less than $596,900, and OVER_15MINS_CALLS_PER_MONTH greater than and equal to 7.5, and REPORTED_USAGE_LEVEL is average or very little, and REPORTED_SATISFACTION is average or satisfied, and COLLEGE is no. This leaf node only had 8 observations which is not even 1% of the data, the probability is 0.75.
2. HOUSE less than $596,900, and OVER_15MINS_CALLS_PER_MONTH greater than and equal to 7.5, and REPORTED_USAGE_LEVEL is high, little, or very high, and HOUSE greater than and equal to $181,400 and HOUSE is less than $565,300, and HOUSE less than $467,400 and HOUSE less than $246,900, and HANDSET_PRICE is less than $229.5, and HOUSE is greater than and equal to $230,300. This leaf node is also small with 12 observations and a probability of 0.92. This had the highest entropy of all Yes majority class leaf nodes.
3. HOUSE is less than $596,900, and OVER_15MINS_CALLS_PER_MONTH is greater than and equal to 7.5, and REPORTED_USAGE_LEVEL is high, little, or very high, and HOUSE is less than and equal to $181,400, and HOUSE is less than $565,300, and HOUSE less than $467,400, and HOUSE is less than $246,900, and HANDSET_PRICE less than $229.5, and HOUSE less than $230,300, and REPORTED_SATISFACTION is unsatisfactory, very satisfactory, or very unsatisfactory. This majority Yes class leaf node again had 0% of the data and probability of 0.6.
4. HOUSE $596,900 and OVER_15MINS_CALLS_PER_MONTH is greater than and equal to 7.5, and REPORTED_USAGE_LEVEL=high, little, or very high, and HOUSE is greater than and equal to $181,400, and HOUSE is less than $565,300, and HOUSE is greater than and equal to $467,400, and HANDSET_PRICE is greater than and equal to $261.5, and LEFTOVER is less than 58, and OVER_15MINS_CALLS_PER_MONTH less than 24.5. This leaf node with majority class Yes had 1% of data and probability of 0.72.

The validation ROC and AUC were also analyzed in depth. The AUC is 0.6278 for the validation set which again better than random guessing and in general an okay number although it would have been preferable to be higher stronger performance at predicting. Please find the ROC curve below:



The ROC curve above would have been more preferred to have been straight up to more steeply at the start of the curve, so the predictions are right early on. Nevertheless, the model is still more robust than random guessing.

SVM - RBF

For SVM model type I ran Polynomial 1, Polynomial 2 and RBF models and fine-tuned them using the complexity parameter. Of all optimized models, the best performing model with the highest AUC on the validation set proved to be RBF. The results were as follows:

	P=1	P=2	RBF
Complexity	C=1	C=0.5	**C=4**
Validation AUC	0.6001	0.6433	**0.6464**
Validation Error	9.4%	9.3%	**9.2%**

Not only did the model have the highest AUC but also the lowest validation error. The RBF model will be described further below. To see the details for the weaker models - Polynomial 1 and Polynomial 2 please see appendix 8 together with a summary of the respective model's performance.

For this model all features were included and no features were removed – not even highly correlated. The only pre-processing was to rescale the numeric features 0 to 1. It is important to note there was no missing values for features and the outliers were left as they were and not transformed in anyway.

Please note, I did run several models with different pre-processing (highly correlated features removed, numeric featured transformed by natural log with missing values for numeric features filled with the median etc.) but they did not perform better than this model on validation AUC and validation error.

The complexities were changed as the tuning parameter and the results recorded are shown below.

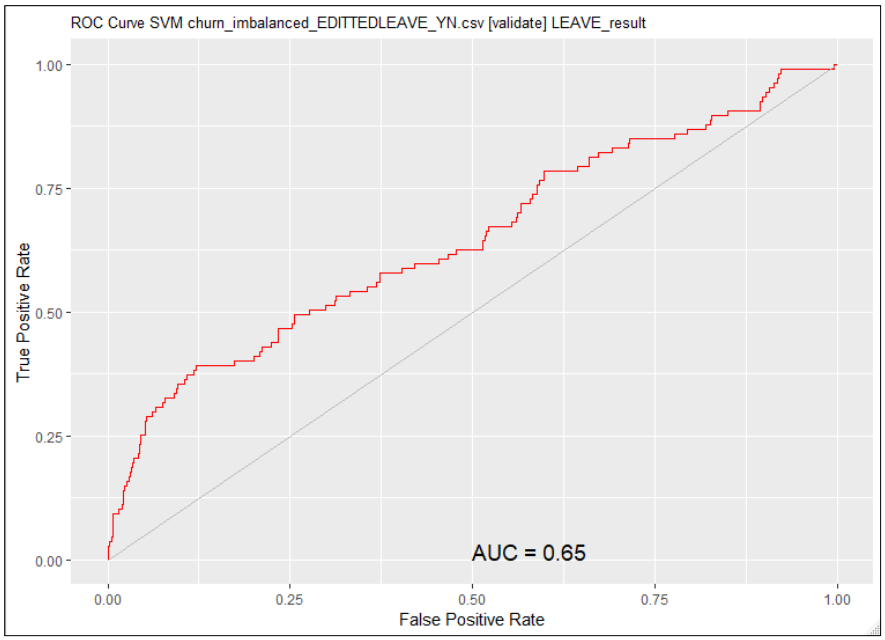
Complexity	C=.1	C=1	C=10
Validation Error	9.4%	9.4%	9.5%
Validation AUC	0.6460	0.6460	0.6494

Complexity	C=.2	C=.3	C=.4	C=.5	C=.6	C=.7	C=.8	C=.9
Validation Error	9.4%	9.4%	9.4%	9.4%	9.4%	9.4%	9.4%	9.4%
Validation AUC	0.6461	0.6462	0.6461	0.6460	0.6459	0.6460	0.6459	0.6461

Complexity	C=2	C=3	**C=4**	C=5	C=6	C=7	C=8
Validation Error	9.4%	9.4%	**9.2%**	9.2%	9.3%	9.3%	9.4%
Validation AUC	0.6463	0.6463	**0.6464**	0.6464	0.6465	0.6477	0.6490

Optimal model at C=4 was selected as it has the lowest validation error at 9.2% and the higher end of AUC values. For going to a greater complexity of C=10, I did not think it was worth the slight AUC improvement and the validation error was also significantly worse. In reviewing the validation confusion matrix, the model is making a small number of predictions for yes. To see the validation confusion matrix please see appendix 9.

The validation ROC curve and AUC are shown below and reviewed in further detail.



The validation ROC curve looks extremely strong at the start. The validation UC is also an okay value and performing better than random guessing. Ideally it could be a lot stronger in the northwest of the ROC curve plot and closer to perfection at 1.

Neural Network

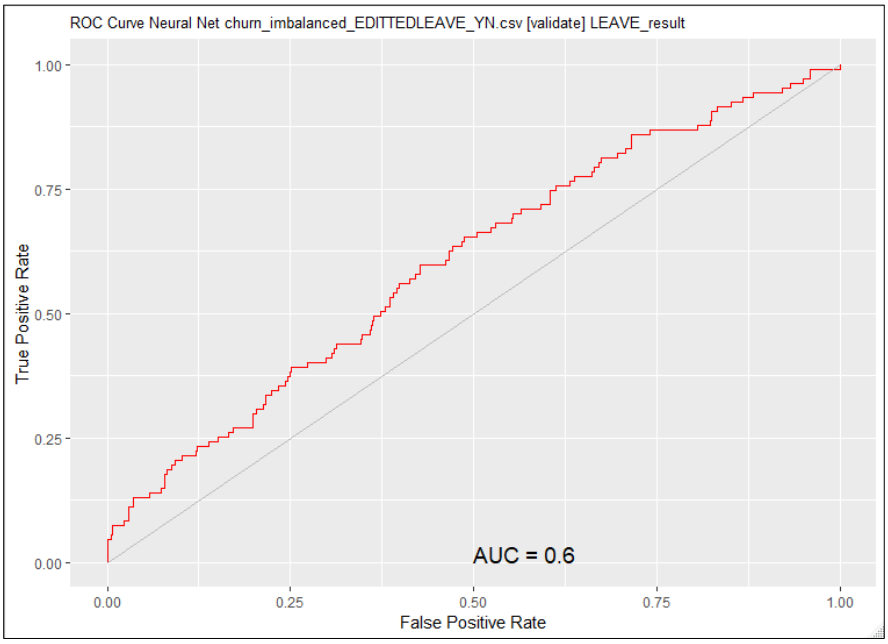
For the neural network model type pre-processing included transforming all numeric features by natural log due to their skewed distributions. For the missing values that then appeared for OVERAGE, HOUSE and OVER_15MINS_CALLS_PER_MONTH after performing natural log, they were replaced with the median as there was still a slight difference between the mean and median values. Then numeric features rescaled 0 to 1 to be on the same scale as the final step performed. I did attempt various other pre-processing for example with highly correlated features removed and no transformations by log applied but they did not perform as well on validation AUC and validation error.

The grid search below shows the various iterations of the model that I ran tuning the parameter for number of hidden nodes.

Number of Hidden Layer Nodes	Validation Error	Validation AUC
2	**9.1%**	**0.6030**
3	10%	0.6270
4	10.7%	0.5503
5	10.4%	6002
6	10.6%	0.6004
7	10.6%	0.5894
8	10.4%	0.5800
9	10.9%	0.5925
10	11.1%	0.6029

The optimal neural network was selected at 2 hidden nodes as it has the highest AUC of 0.6030 and gave a validation error of 9.1%. The confusion matrix is showing predictions for yes. To view the validation confusion matrix please see appendix 10 for details.

The validation ROC curve and AUC were also reviewed. Validation AUC is 0.6030 which is better than random guessing but still not as high as one would expect from a robust strong model. The quality of the model is okay. The ROC curve is provided below.



The curve line is the weakest so far and a steeper incline would have been preferred, especially at the start.

Random Forest

For this model type, I again attempted numerous pre-processing such as natural logging numeric features with skewed distributions and rescaling 0 to 1, removing highly correlated features etc. to see which combination provided the highest AUC and lowest error on the validation set. Removing outliers was not attempted as this model is not impacted by outliers. I found including all features performed the best. I did attempt a 50:50 sample size but it was the worst performance, so I ultimately decided to go with a model where no sample size restrictions were set. Number of trees was set to 500.

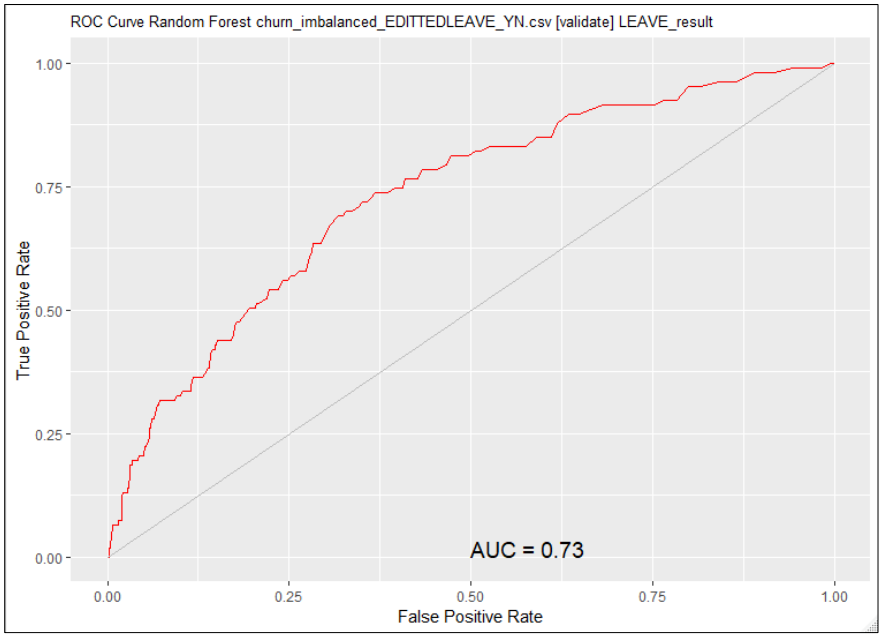
Number of variables was set as the tuning parameter and the results are below.

Variables	3	4	5
Validation Error	**9.6%**	9.8%	10%
Validation AUC	**0.7300**	0.7291	0.7301

The optimal model was selected with 3 variables and not 5 variables as the validation error was lowest at 9.6% and the additional complexity in the model was not worth the minimal improvement in AUC of 0.001 for 5 variables.

The validation confusion matrix details can be found in appendix 11 and does make Yes predictions. For a brief summary on the text view confusion matrix, variable importance and error rate please see appendix 12.

The validation AUC for this model is 0.7300 which is fairly high quality. This is the highest of the models so fat. The ROC curve is shared below.



The validation ROC curve does look strong showing extremely steep incline at the start.

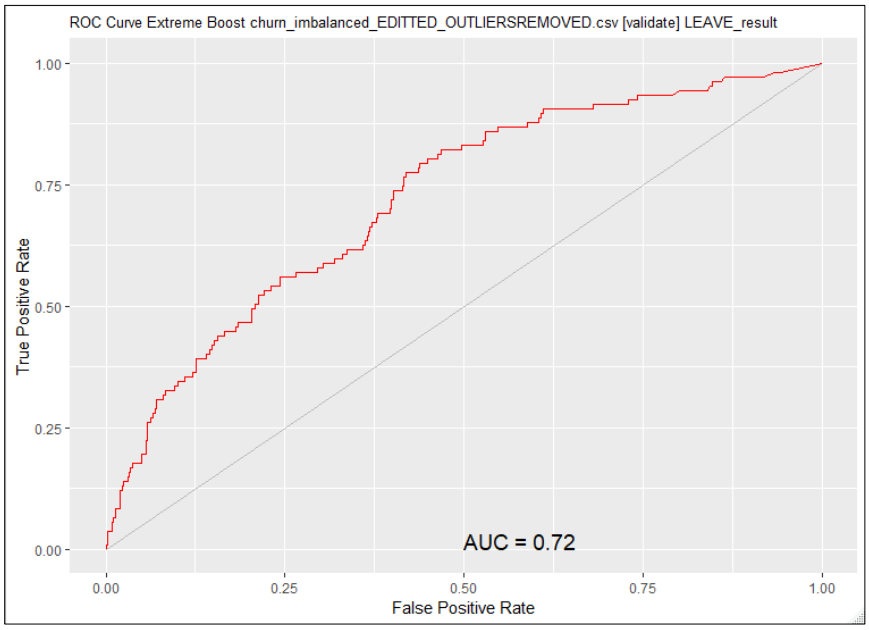
Adaptive Boosting

The adaptive boosting model type was attempted in various ways but including all features and outliers replaced with median performed the strongest. I attempted other pre-processing without outliers resolved, highly correlated removed, numeric features natural logged and rescaled 0 to 1 – all performing worse using validation AUC and validation error.

All default settings were selected for Adaptative boosting model which include trees 50, max. depth 6, min. split 20, complexity 0.01, X Val 10, iterations 50. There are tuning parameters such as number of trees but in class we were advised to stick to the default options, so no tuning of parameters was applied.

The resulting model achieved 9.5% validation error. For the validation confusion matrix please see appendix 13, it shows very minimal Yes predictions. To view a brief summary of the training error plot, variable importance plot see appendix 14.

The validation AUC was also analyzed, and it showed the model's quality at 0.7224 which is robust and doing a fairly good job at predicting. The ROC curve is below.



As with the random forest model type, this curve is fairly strong although it would have been desirable to have it even stronger steeper incline.

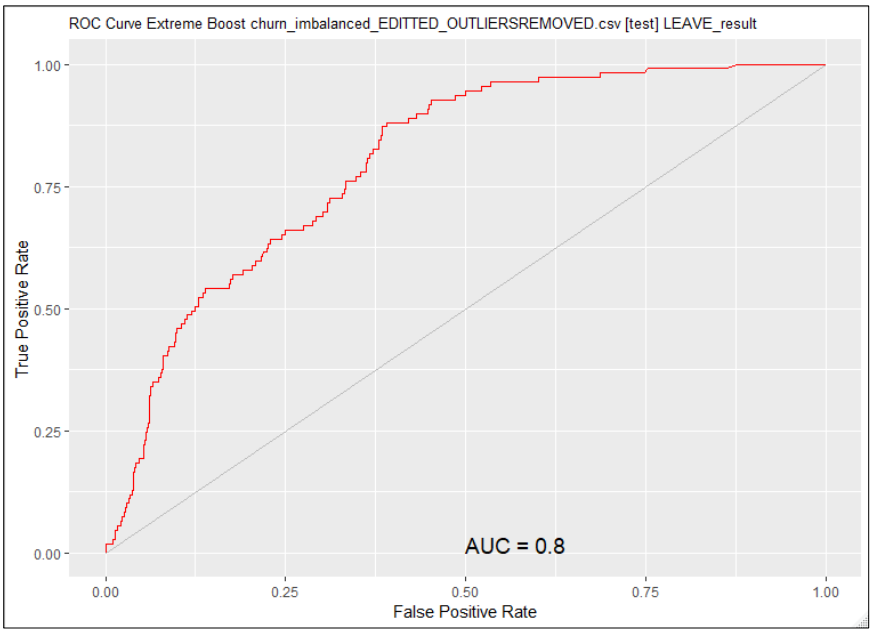
Best Overall Model

The evaluation criterion to select the best performing model out of all 6 models is the ROC curve and its associated AUC on the test set. Below is a quick summary of the various models that were ran and the AUC for each.

Model Type	AUC
SVM RBF	0.5828
Neural Network	0.6537
Logistic Regression	0.7130
Decision Tree	0.7230

Random Forest	0.7683
Adaptive Boosting	**0.7997**

The best overall model that has the highest ROC curve was adaptive boosting with an AUC of 0.7997. The test ROC curve is displayed below:



This plot does show robust performance at predicting and is the strongest curve seen so far. It is performing exceptionally well compared to random guessing and has a promising steep incline at the start of the curve.

Profit curve

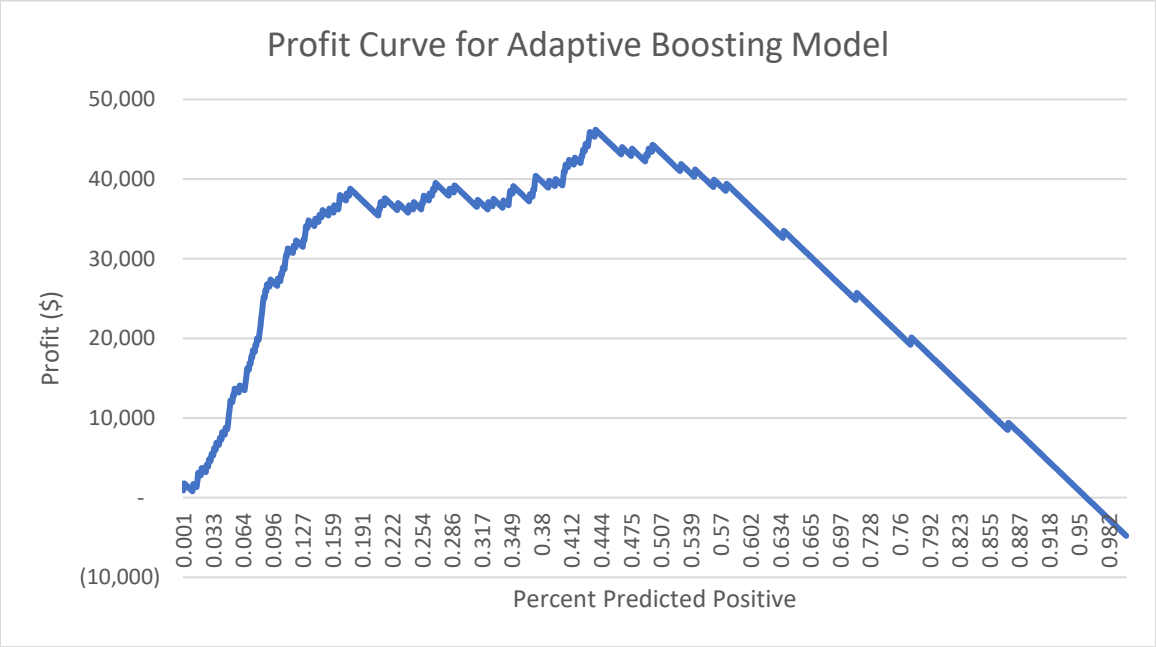
A profit curve was then created in Excel based on the testing data using the probabilities exported from Rattle that were sorted by probability largest to smallest as Yes represented leave (churner). For each instance the percentage of the list predicted as positive was also calculated. Finally, the expected profit was calculated for the cost and benefit details shared. The below highlights what was shared in order to calculate the profit:

Assume that the cost for an intervention (say, a discount given to a predicted churner) is $100, that all interventions are successful, and that retaining a customer is worth $1000. Based on this result, give a specific recommendation for how many people should be given the discount.

This was put into a cost matrix before completing the expected profit calculation in Excel. For target class one (churner) the profit per customer is $1000-100 = $900 so the profit on a churner who decides to stay is $900. A churner who does leave is -$100 as this is the discount offered to customers to entice them to stay.

		Predicted	
		Y	N
Actual	Y	$900	0
	N	-$100	0

The below profit curve line graph was built based on the cumulative profit using the above cost and benefit details.

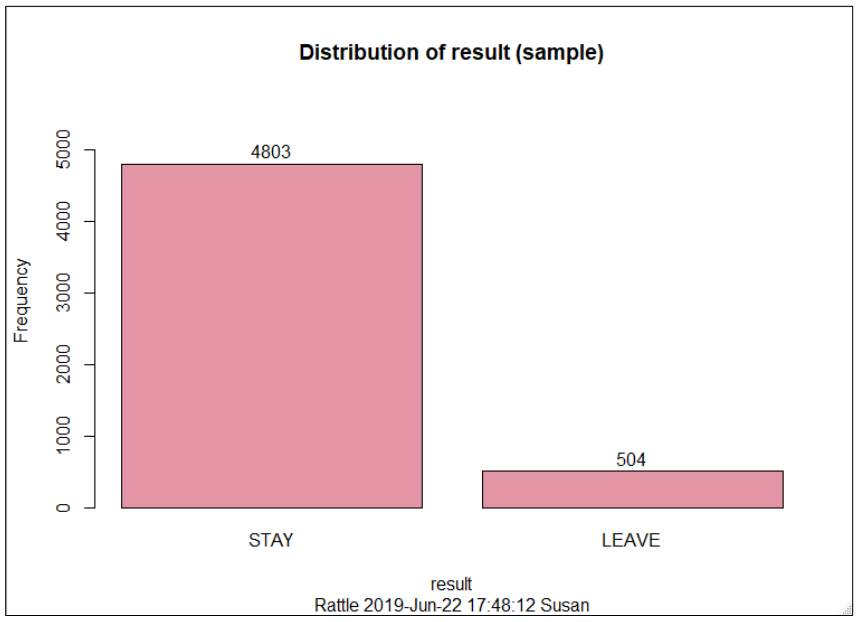


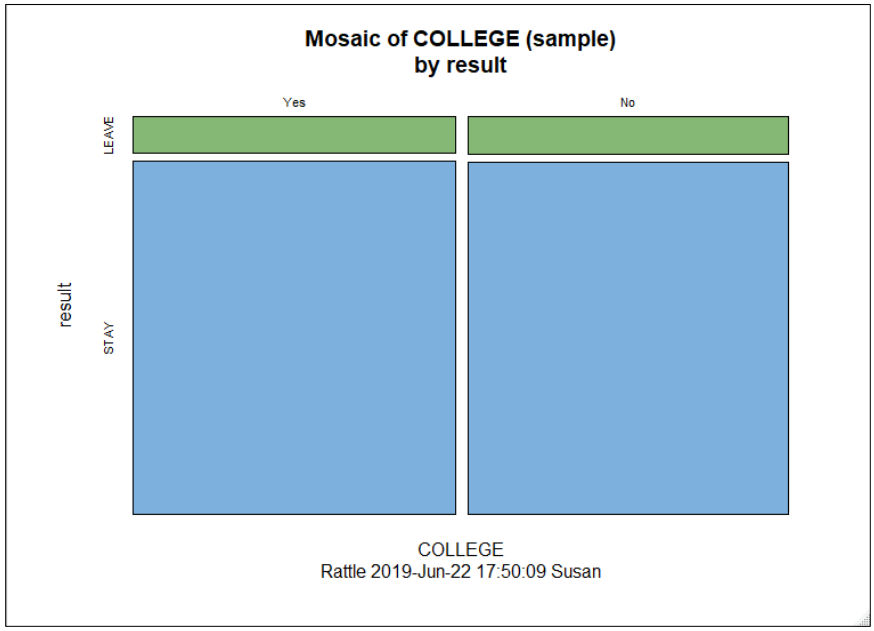
In general, the profit curve is as expected in terms of the overall shape and tail end of the profit curve going into negative profit. The maximum profit is reached at $46,200. In reviewing the profit curve, I recommend targeting the top 43.76% (498 out of 1138) of the decreasing scored customers to be given the discount.

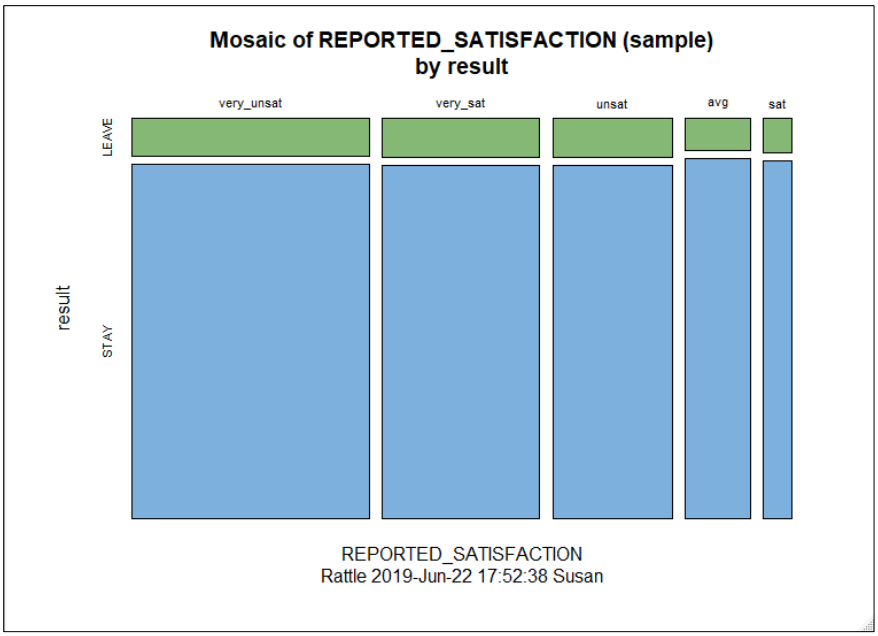
In conclusion, with this dataset and the fine tuning each of the model type the best model using ROC curve as the evaluation criterion proved to be adaptive boosting. Applying the churn business case, a profit curve was calculated and plotted to recommend exactly how many should receive the discount. For the next discounted campaign, the top 43.76% should be targeted.

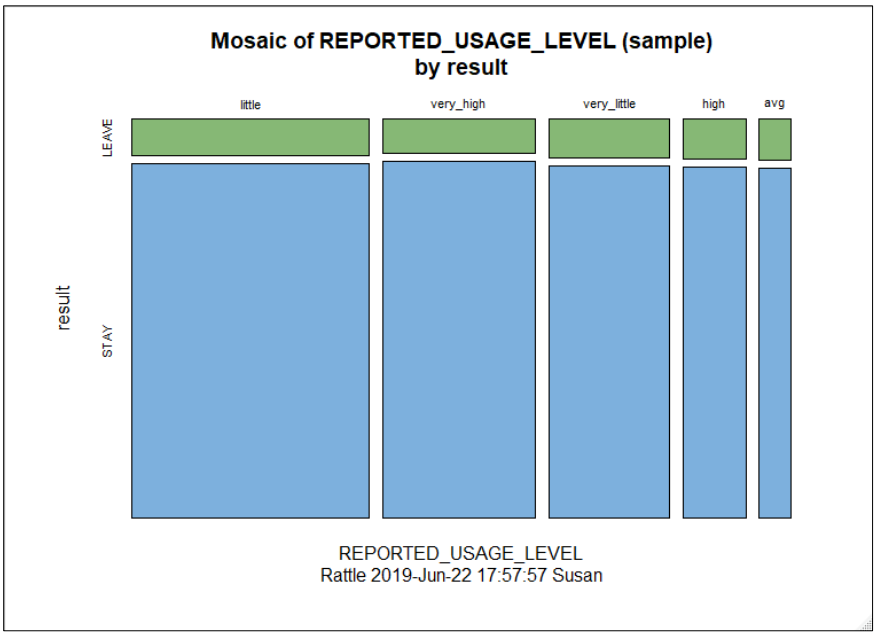
Appendix 1

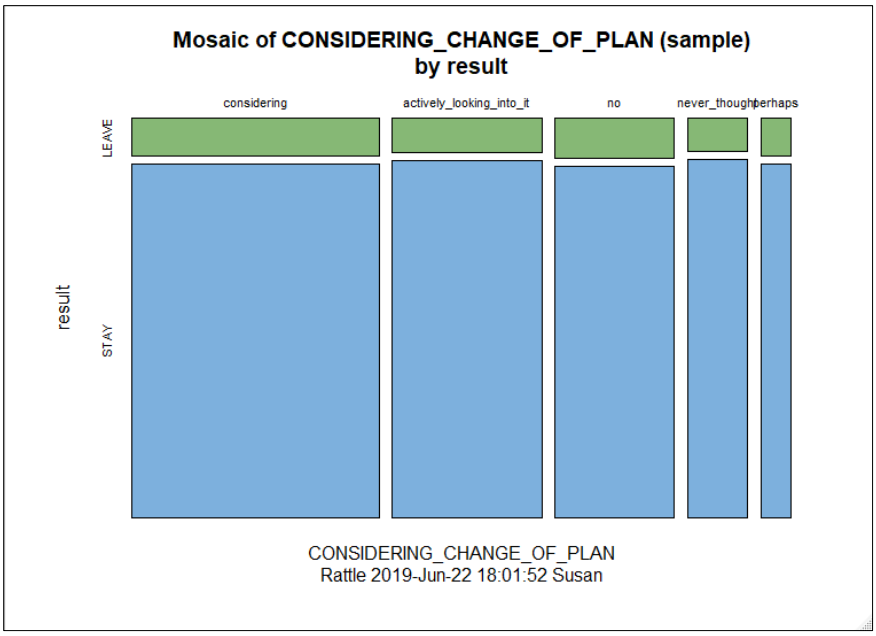
Distributions of each feature from Churn dataset.

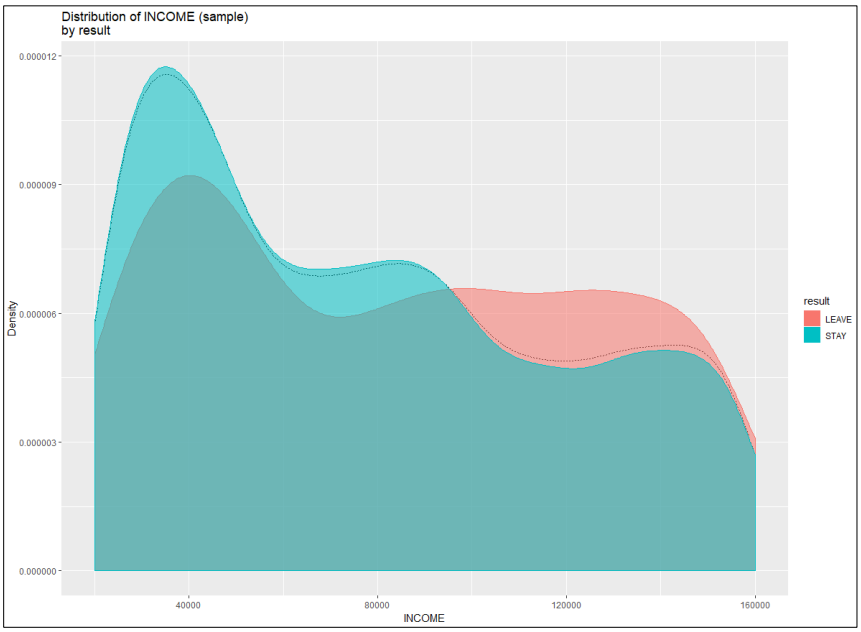


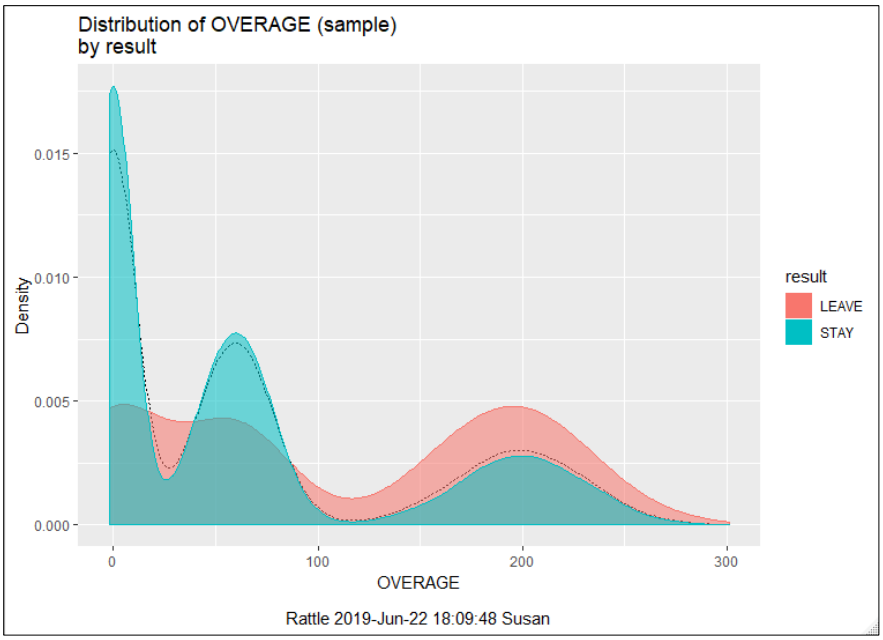


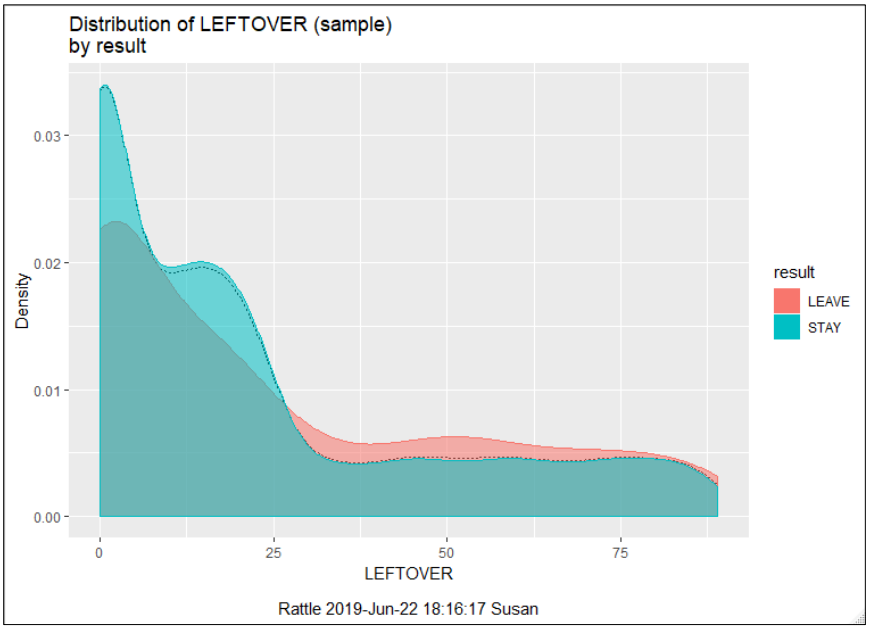

Mosaic of REPORTED_SATISFACTION (sample)
by result
very_unsat
very_sat
unsat
avg
sat
LEAVE
STAY
result
REPORTED_SATISFACTION
Rattle 2019-Jun-22 17:52:38 Susan

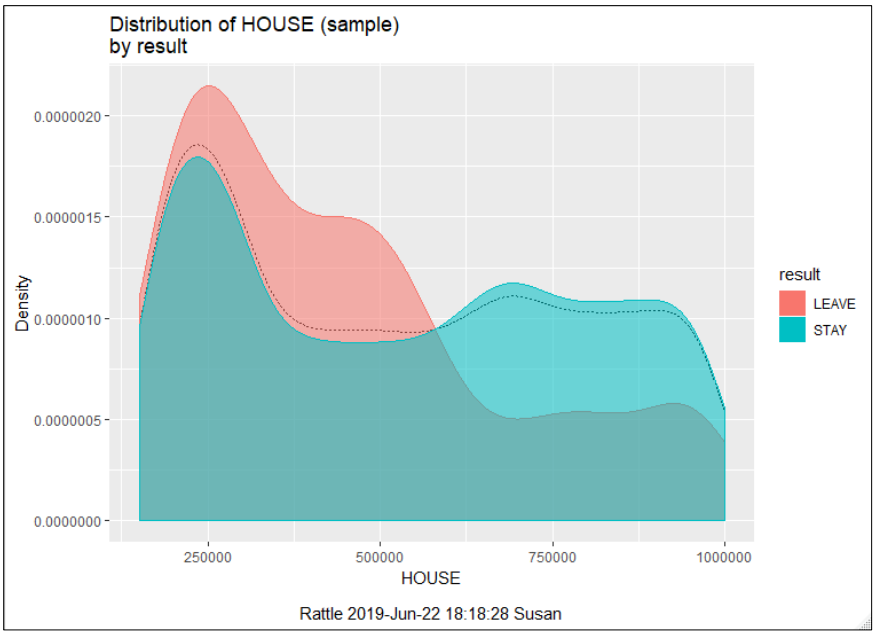

Mosaic of REPORTED_USAGE_LEVEL (sample)
by result
little
very_high
very_little
high
avg
LEAVE
STAY
result
REPORTED_USAGE_LEVEL
Rattle 2019-Jun-22 17:57:57 Susan

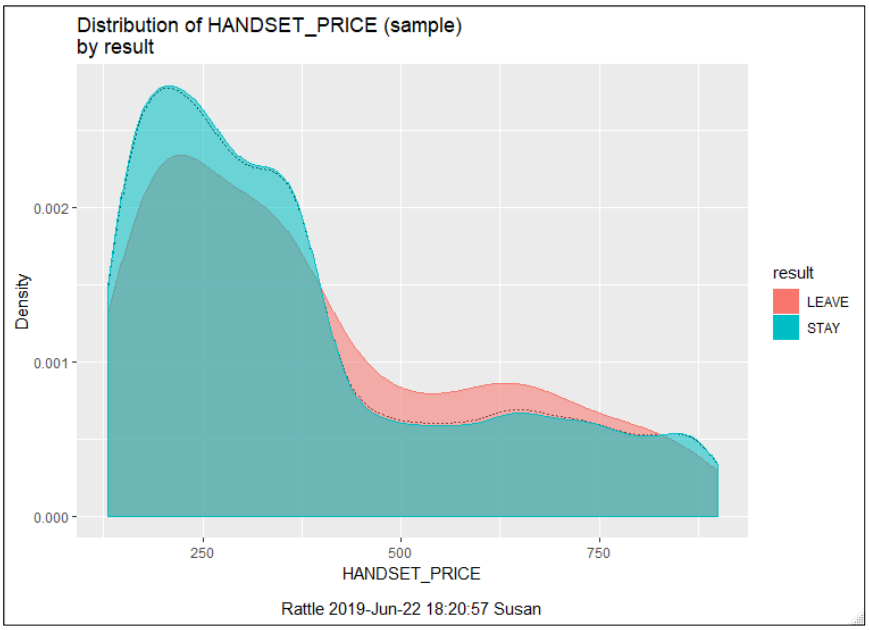

Mosaic of CONSIDERING_CHANGE_OF_PLAN (sample)
by result
considering
actively_looking_into_it
no
never_thought
perhaps
LEAVE
STAY
result
CONSIDERING_CHANGE_OF_PLAN
Rattle 2019-Jun-22 18:01:52 Susan

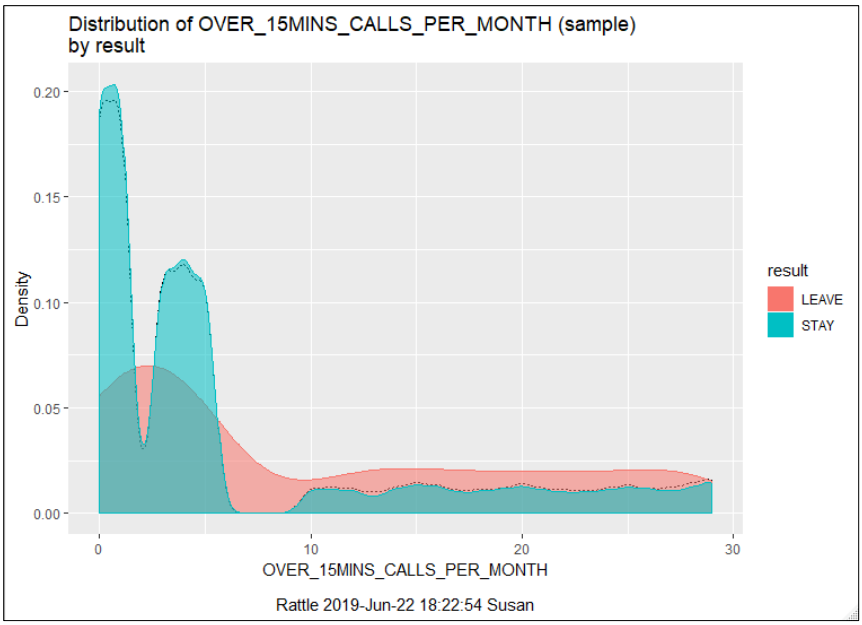

Distribution of INCOME (sample)
by result
0.000012
0.000009
0.000006
0.000003
0.000000
Density
40000
80000
120000
160000
INCOME
result
LEAVE
STAY

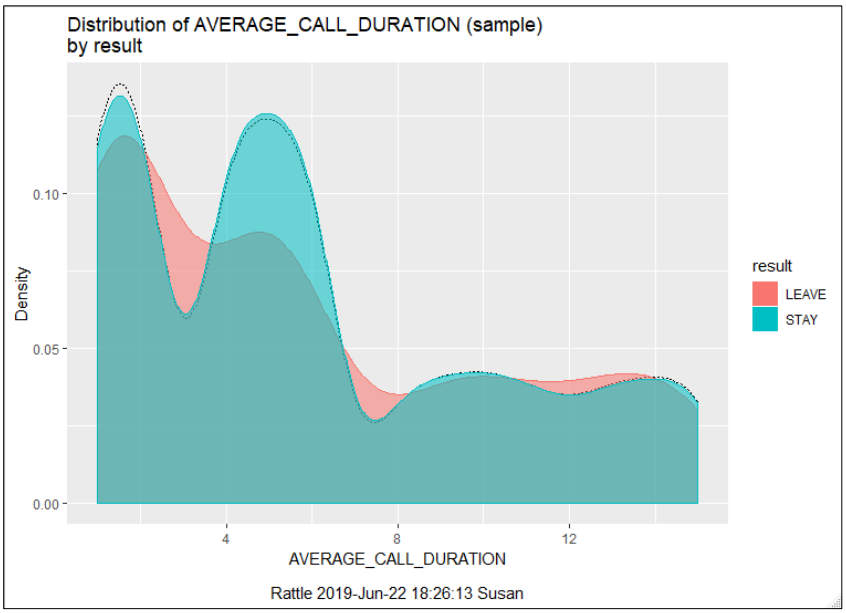

Distribution of OVERAGE (sample)
by result
Density
0.015
0.010
0.005
0.000
0
100
200
300
OVERAGE
result
LEAVE
STAY
Rattle 2019-Jun-22 18:09:48 Susan


Distribution of LEFTOVER (sample)
by result
Density
0.03
0.02
0.01
0.00
0
25
50
75
LEFTOVER
result
LEAVE
STAY
Rattle 2019-Jun-22 18:16:17 Susan


Distribution of HOUSE (sample)
by result
Density
0.0000020
0.0000015
0.0000010
0.0000005
0.0000000
250000
500000
750000
1000000
HOUSE
result
LEAVE
STAY
Rattle 2019-Jun-22 18:18:28 Susan

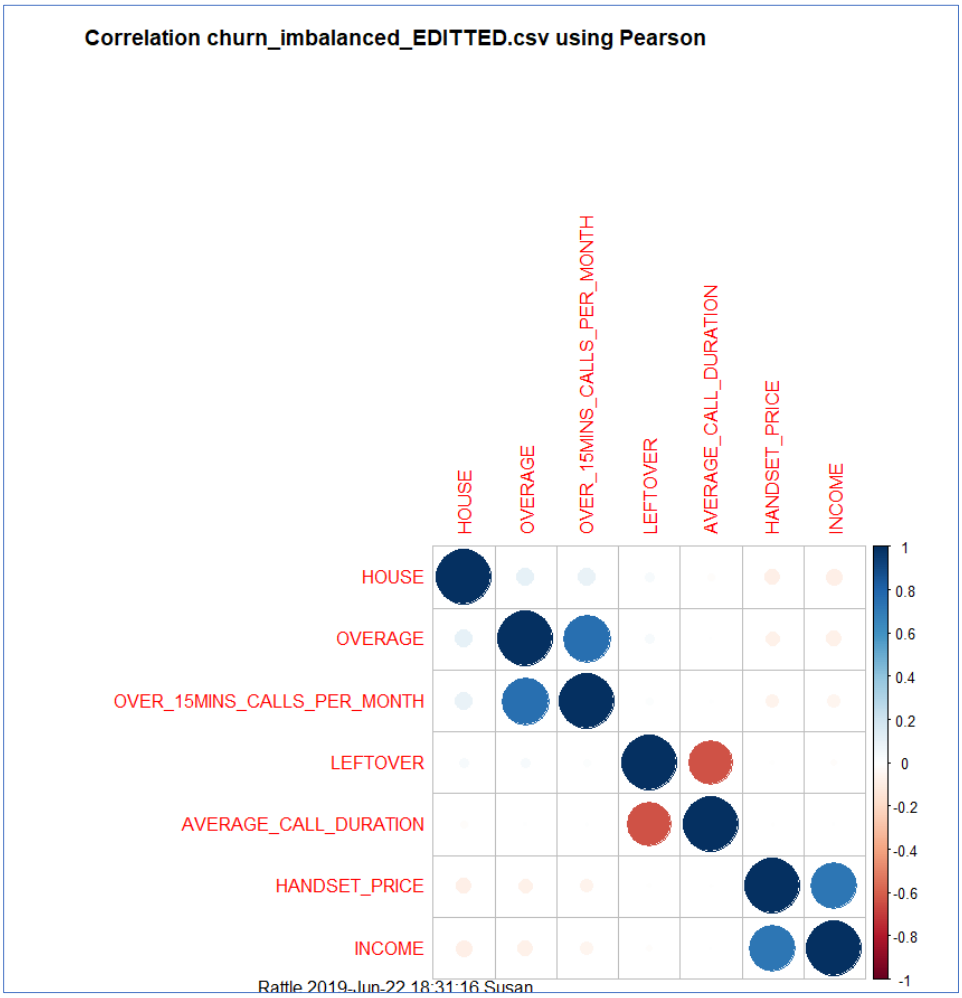

Distribution of HANDSET_PRICE (sample)
by result
Density
0.002
0.001
0.000
250
500
750
HANDSET_PRICE
result
LEAVE
STAY
Rattle 2019-Jun-22 18:20:57 Susan


Distribution of OVER_15MINS_CALLS_PER_MONTH (sample)
by result
Density
0.20
0.15
0.10
0.05
0.00
0
10
20
30
OVER_15MINS_CALLS_PER_MONTH
result
LEAVE
STAY
Rattle 2019-Jun-22 18:22:54 Susan


Distribution of AVERAGE_CALL_DURATION (sample)
by result
Density
0.10
0.05
0.00
4
8
12
AVERAGE_CALL_DURATION
result
LEAVE
STAY
Rattle 2019-Jun-22 18:26:13 Susan

Appendix 2

Correlation Matrix



Appendix 3

Logistic regression initial model results

```
Terms added sequentially (first to last)


                                       Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                                                    5306     3331.6
COLLEGE                                 1    0.026      5305     3331.6  0.872609
REPORTED_SATISFACTION                   4    1.434      5301     3330.1  0.838336
REPORTED_USAGE_LEVEL                    4    1.479      5297     3328.6  0.830390
CONSIDERING_CHANGE_OF_PLAN              4    2.589      5293     3326.1  0.628783
IMD_RLG_INCOME                          1    9.206      5292     3316.9  0.002412 **
IMD_RLG_HOUSE                           1   37.721      5291     3279.1 8.163e-10 ***
IMD_RLG_HANDSET_PRICE                   1    0.002      5290     3279.1  0.960689
IMD_RLG_AVERAGE_CALL_DURATION           1    3.603      5289     3275.5  0.057691 .
IMD_IMD_RLG_OVERAGE                     1  103.944      5288     3171.6 < 2.2e-16 ***
IMD_IMD_RLG_LEFTOVER                    1    2.505      5287     3169.1  0.113496
IMD_IMD_RLG_OVER_15MINS_CALLS_PER_MONTH 1   19.697      5286     3149.4 9.075e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Appendix 4

Logistic regression optimized model results

```
Terms added sequentially (first to last)


                                       Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                                                    5306     3331.6
IMD_RLG_INCOME                          1    9.259      5305     3322.3  0.002343 **
IMD_RLG_HOUSE                           1   38.097      5304     3284.2 6.731e-10 ***
IMD_IMD_RLG_OVERAGE                     1  104.816      5303     3179.4 < 2.2e-16 ***
IMD_IMD_RLG_OVER_15MINS_CALLS_PER_MONTH 1   20.385      5302     3159.0 6.331e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Appendix 5

Logistic regression model validation confusion matrix.

```
Error matrix for the Linear model on churn_imbalanced_EDITTED_OUTLIERSREMOVED.csv [validate] (counts):

      Predicted
Actual   No Yes Error
   No  1030   0     0
   Yes  107   0   100

Error matrix for the Linear model on churn_imbalanced_EDITTED_OUTLIERSREMOVED.csv [validate] (proportions):

      Predicted
Actual   No Yes Error
   No  90.6   0     0
   Yes  9.4   0   100

Overall error: 9.4%, Averaged class error: 50%
```

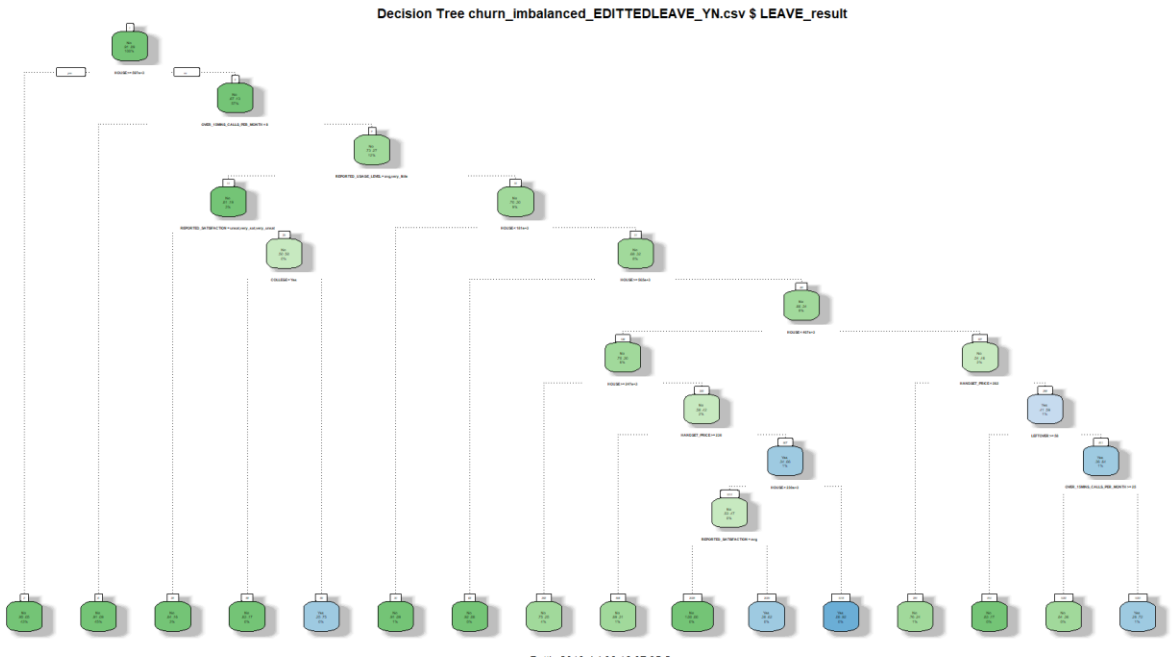
Appendix 6

Decision tree optimal model validation confusion matrix

```
Error matrix for the Decision Tree model on churn_imbalanced_EDITTEDLEAVE_YN.csv [validate] (counts):

      Predicted
Actual   No Yes Error
   No  1018  12   1.2
   Yes  104   3  97.2

Error matrix for the Decision Tree model on churn_imbalanced_EDITTEDLEAVE_YN.csv [validate] (proportions):

      Predicted
Actual   No Yes Error
   No  89.5 1.1   1.2
   Yes  9.1 0.3  97.2

Overall error: 10.2%, Averaged class error: 49.2%
```

Appendix 7

Optimized decision tree plot



Appendix 8

Polynomial 1

For this model pre-processing included outliers being replaced with median and numeric features rescaled 0 to 1. I did run multiple models different pre-processing (without outliers removed, no rescaling, natural logging, highly correlated features removed etc.) but not matter what was attempt it failed to find a validation error that changed irrespective of complexity parameter. Within R console, I found this message 'maximum number of iterations reached 0.005979982 0.005690865' from C=0.1 and anything above. I select the best pre-processing with lowest validation error and the best AUC amongst all attempts. For this the pre-processing was as I first described - outliers being replaced with median and numeric features rescaled 0 to 1.

I tuned the model using the complexity parameter with the results shown below. Degrees was set to 1.

Complexity	C=0.1	**C=1**	C=10	C=20
Validation Error	9.4%	**9.4%**	9.4%	9.4%
Validation AUC	0.5635	**0.6001**	0.4613	0.4142

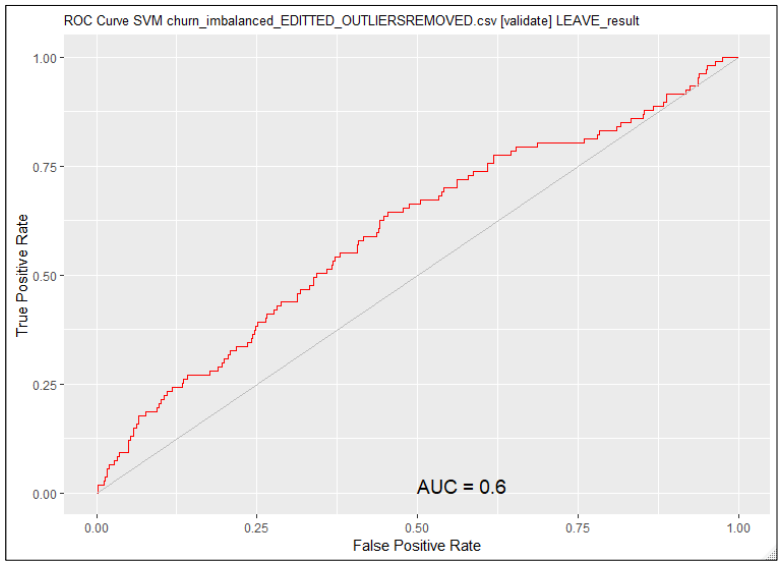
Complexity	C=0.2	C=0.3	C=0.4	C=0.5	C=0.6	C=0.7	C=0.8	C=0.9
Validation Error	9.4%	9.4%	9.4%	9.4%	9.4%	9.4%	9.4%	9.4%
Validation AUC	0.5161	0.3934	0.5888	0.5933	0.5540	0.5732	0.4065	0.4529

Complexity	C=1.1	C=1.2	C=1.3	C=1.4	C=2	C=3	C=4	C=5
Validation Error	9.4%	9.4%	9.4%	9.4%	9.4%	9.4%	9.4%	9.4%
Validation AUC	0.5714	0.5135	0.5817	0.3862	0.4350	0.4683	0.4990	0.4463

As shown, I could not get the validation error to drop at any point, but the AUC did change. I therefore went with the optimal model for Polynomial 1 with complexity at 1. The validation confusion matrix is shown below. There were no Yes predictions for this model.

```
Error matrix for the SVM model on churn_imbalanced_EDITTED_OUTLIERSREMOVED.csv [validate] (counts):

      Predicted
Actual   No Yes Error
   No  1030   0     0
   Yes  107   0   100

Error matrix for the SVM model on churn_imbalanced_EDITTED_OUTLIERSREMOVED.csv [validate] (proportions):

      Predicted
Actual   No Yes Error
   No  90.6   0     0
   Yes  9.4   0   100

Overall error: 9.4%, Averaged class error: 50%
```

The validation ROC curve is also shown below which is displaying results only slighty better than randon guessing. It is a poor model overall. The AUC is only 0.6001.



Polynomial 2

The pre-processing for this model included outliers replaced with median and numeric features rescaled 0 to 1. Again, I did run multiple models but found either the validation error did not change irrespective of complexity due to 'maximum number of iterations reached 0.0001362127 0.0001349079' in the R con sole or the model froze while running and results did not display for C=1.1 and above. I used mainly the AUC to make a determination on the best optmized model.

The results of the various complexity parameter tuning I was able to do are shown below. Degrees was s et to 2.

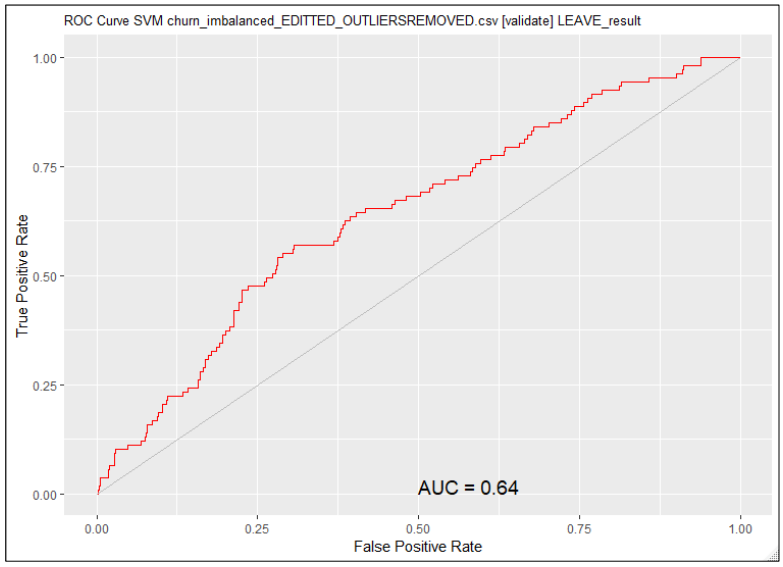
Complexity	**C=0.1**	**C=1**	**C=10**
Validation Error	9.3%	9.3%	Would not run - froze
Validation AUC	0.6312	0.6181	Would not run - froze

Complexity	**C=0.2**	**C=0.3**	**C=0.4**	**C=0.5**	**C=0.6**	**C=0.7**	C=0.8	**C=0.9**
Validation Error	9.3%	9.3%	9.3%	**9.3%**	9.3%	9.3%	9.3%	9.3%
Validation AUC	0.6200	0.5285	0.5801	**0.6433**	0.5979	0.5868	6041	0.6152

The optimal model was therefore chosen as the lowest complexity at C=0.5 which had the highest AUC. The validation error is 9.3% with the validation confusion matrix shown below. There were no Yes predictions in this model.

```
Error matrix for the SVM model on churn_imbalanced_EDITTED_OUTLIERSREMOVED.csv [validate] (counts):

      Predicted
Actual   No Yes Error
   No  1031   0     0
   Yes  106   0   100

Error matrix for the SVM model on churn_imbalanced_EDITTED_OUTLIERSREMOVED.csv [validate] (proportions):

      Predicted
Actual   No Yes Error
   No  90.7   0     0
   Yes  9.3   0   100

Overall error: 9.3%, Averaged class error: 50%
```

The ROC curve is presented below and again similar to polynomial 1, is only just slightly better than ramndom guessing. It is not a good model. The AUC is 0.6433.



These were the worst models of the SVM model type for this dataset when using validation AUC as the deciding factor.

Appendix 9

RBF Model Validation confusion matrix

```
Error matrix for the SVM model on churn_imbalanced_EDITTEDLEAVE_YN.csv [validate] (counts):

      Predicted
Actual   No Yes Error
   No  1030   0   0.0
   Yes  105   2  98.1

Error matrix for the SVM model on churn_imbalanced_EDITTEDLEAVE_YN.csv [validate] (proportions):

      Predicted
Actual   No Yes Error
   No  90.6 0.0   0.0
   Yes  9.2 0.2  98.1

Overall error: 9.2%, Averaged class error: 49.05%
```

Appendix 10

ANN validation confusion matrix

```
Error matrix for the Neural Net model on churn_imbalanced_EDITTEDLEAVE_YN.csv [validate] (counts):

      Predicted
Actual   No Yes Error
   No  1029   1   0.1
   Yes  103   4  96.3

Error matrix for the Neural Net model on churn_imbalanced_EDITTEDLEAVE_YN.csv [validate] (proportions):

      Predicted
Actual   No Yes Error
   No  90.5 0.1   0.1
   Yes  9.1 0.4  96.3

Overall error: 9.1%, Averaged class error: 48.2%
```

Appendix 11

Forest validation confusion matrix

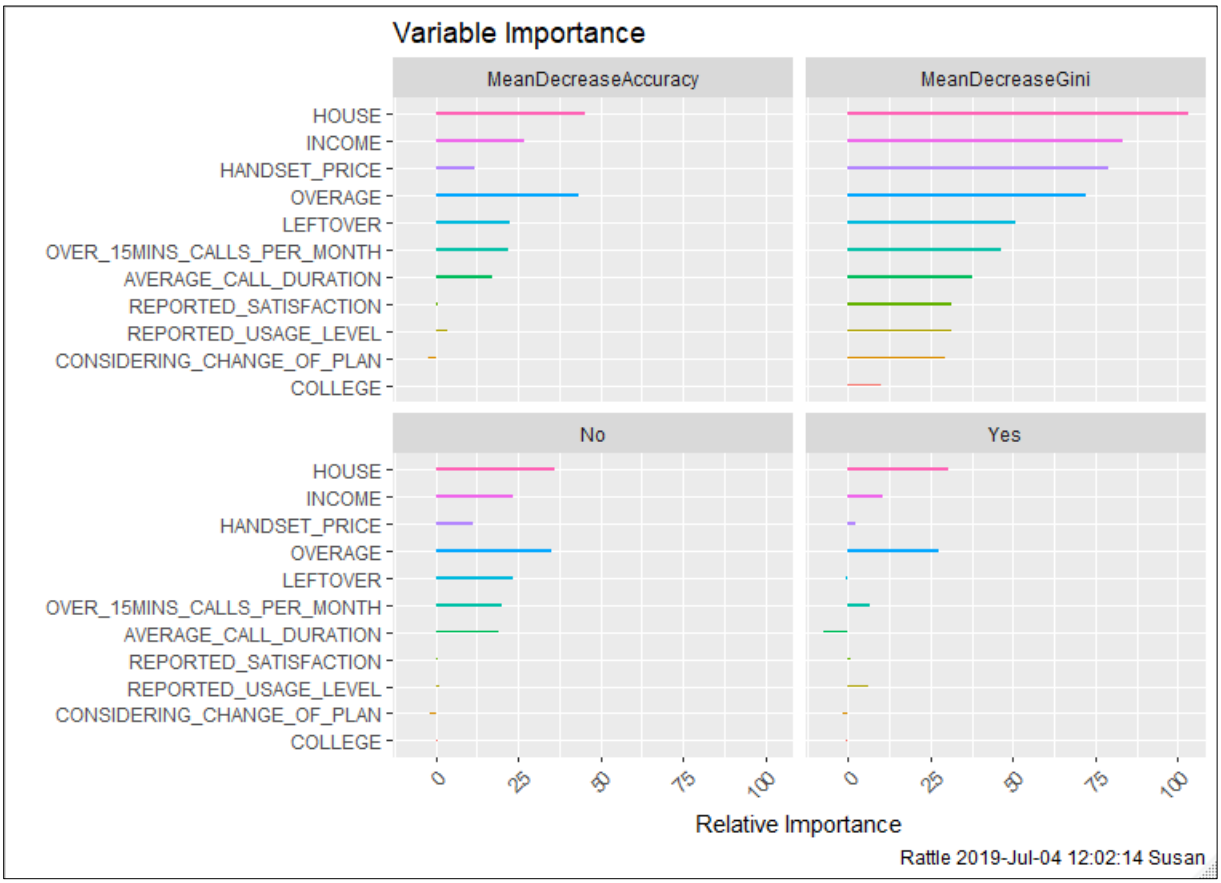
```
Error matrix for the Random Forest model on churn_imbalanced_EDITTEDLEAVE_YN.csv [validate] (counts):

      Predicted
Actual   No Yes Error
   No  1027   3   0.3
   Yes  106   1  99.1

Error matrix for the Random Forest model on churn_imbalanced_EDITTEDLEAVE_YN.csv [validate] (proportions):

      Predicted
Actual   No Yes Error
   No  90.3 0.3   0.3
   Yes  9.3 0.1  99.1

Overall error: 9.6%, Averaged class error: 49.7%
```

Appendix 12

Random forest text view confusion matrix is shown below. The out-of-bag estimate of the error is 9.44% and suggests when resulting model is applied to new observations the answers will be of this error amount. It is fairly close to the validation error of 9.6% shown above. This is very similar to the validation error.

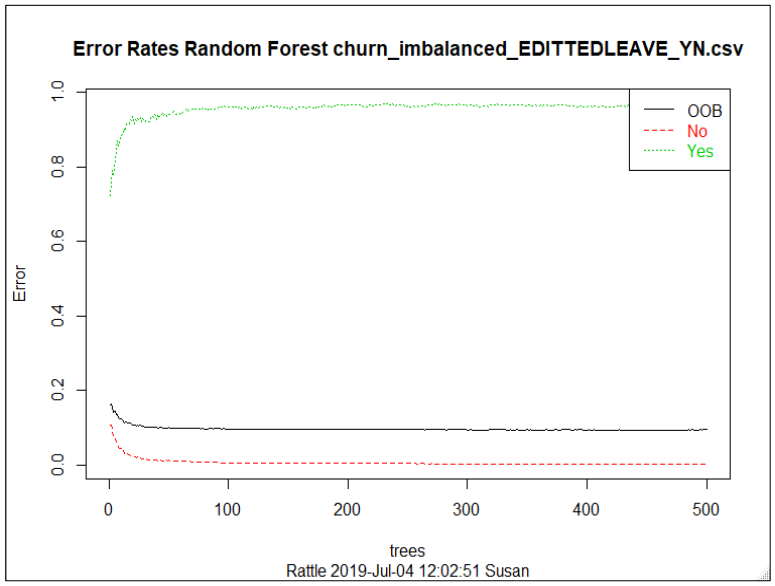
```
Call:
 randomForest(formula = LEAVE_result ~ .,
              data = crs$dataset[crs$train, c(crs$input, crs$target)],
              ntree = 500, mtry = 3, importance = TRUE, replace = FALSE, na.action = randomForest::na.roughfix)

               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 3

        OOB estimate of  error rate: 9.44%
Confusion matrix:
      No Yes class.error
No  4790  13 0.002706642
Yes  488  16 0.968253968
```

The below visual plot displays the accuracy and the Gini importance measures. House is the most important.



The below plot reports the accuracy of the forest of trees against the number of trees that have been included in the forest. The key point to take away is that after some point there is very little changes

after adding more trees. Going beyond 30 trees adds very little value when considering the out-of-bag error rate.



Appendix 13

Adaptive Boosting model validation confusion matrix

```
Error matrix for the Extreme Boost model on churn_imbalanced_EDITTED_OUTLIERSREMOVED.csv [validate] (counts):

      Predicted
Actual   No Yes Error
   No  1029   1   0.1
   Yes  107   0 100.0

Error matrix for the Extreme Boost model on churn_imbalanced_EDITTED_OUTLIERSREMOVED.csv [validate] (proportions):

      Predicted
Actual   No Yes Error
   No  90.5 0.1   0.1
   Yes  9.4 0.0 100.0

Overall error: 9.5%, Averaged class error: 50.05%
```

Appendix 14

Adaptive boosting model training error plot is displayed below and shows the decreasing error rate as more trees are added to the model. This is fairly typical where the error rate drops early on. The 1's just represent curve.

The variable importance plot below is a relative measure that calculates for each tree the improvement in accuracy that the variable chosen to split the dataset offers the model. The top 3 most important features are HANDSET_PRICE, INCOME, and OVERAGE.