Customer Churn

Kevin Phan, kevin.phan@ryerson.ca

Summary:

This paper is an in-depth analysis of the Customer Churn data set achieved by utilizing machine learning algorithms as well as macros. The Customer Churn data set is a list of records for a large number of customers of a telecommunication company with the class attribute of interest being Churn. (possessing values of True or False). We want to address the following:

- What causes a customer to churn?
- What are the characteristics of a customer that will most likely churn?
- How can the company prevent churning? We performed three classification algorithms: Naïve Bayes, Decision Tree and Random Forest. Each algorithm was performed under a 66% training-testing data split as well as a 10-fold Cross Validation where the Random Forest algorithm appears to be most effective in predictive capabilities. The F-Measure was 0.858, 0.944 and 0.948 respectively under k=10-fold environment as it is proven in the latter part of this paper to be the most efficient data partitioning method. Our post-predictive analysis indicates most prevalently that no of customer service calls averaging greater than 3.5 have a strong correlation to a TRUE value of Churn. This remains true irrelevant of the variable regressed on No of Customer Service calls. Visualizations on WEKA indicate that the pattern remains consistent. This was also present in our association rules generated via the a-priori algorithm in WEKA. Few suggestions would be to perform customer service data collection to further analyze and produce customer retention strategies as well as early resolution tactics for immediate customer satisfaction.

Data Preparation

Data consists of 21 attributes with 3,333 instances. The table 1 shows a modified 5-point summary of all attributes with the data type. This information was obtained from sas.

```
proc import out=WORK.CHURN

datafile = "C:\Users\kevin.phan\Desktop\My SAS Files\9.4\churn.csv"

dbms=csv replace; getnames=yes, Datarow = 2;

run; proc print univariate data = WORK>CHURN; run;
```

Attribute	Min	Max	Mean	Standard Deviation	Data Type
State	N/A	N/A	N/A	N/A	Nominal
Account Length	1	243 (818 inst)	101.06	39.8	Numeric
Area Code	N/A	N/A	N/A	N/A	Nominal
Phone Number	N/A	N/A	N/A	N/A	Nominal
Inter Plan	N/A	N/A	N/A	N/A	Nominal
Voicemail Plan	N/A	N/A	N/A	N/A	Nominal
Number of Voicemail Messages	0	51 (846 inst)	8.099	13.78	Numeric
Total Day Min	0	350.8 (366 Inst)	179.77	54.46	Numeric
Total Day Calls	0	350.8(366 inst)	100.43	20.06	Numeric
Total Day Charge	0	30.91 (2733 inst)	30.56	9.25	Numeric
Total Evening Min	43.9	367.7(2733 inst)	200.98	50.71	Numeric
Total Evening Calls	0	170(3220 inst)	100.11	19.92	Numeric

Total Evening Charge	3.73 (822)	30.91 (2733 inst)	17.08	4.31	Numeric
Total Night Min	50.1(2757)	395(2664 inst)	200.87	50.57	Numeric
Total Night Calls	33 (2660)	175 (494 inst)	100.10	19.56	Numeric
Total Night Charge	1.04(1261 inst)	17.77(2664 inst)	9.03	2.27	Numeric
Total Int Min	0	20 (116 inst)	10.23	2.79	Numeric
Total Int Calls	0	20(3292 inst)	4.47	2.46	Numeric
Total Int Charge	0	5.40(116)	2.76	0.75	Numeric
No of Calls Customer Service	0	9 (2381 inst)	1.56	1.31	Numeric

The data set does not contain any NULL or NA values. Each attribute has many outliers. We have decided to keep the outliers that were 3 x IQR since most of the outliers were clusters. The spread of the outliers was very small and so the differences were near negligible when trying to determine what a point of reference between whether the outliers were kept or not. Suggestions were made toward utilizing the Mahalanobis distance and removing any outliers below 5% but as said, the distances seem too similar from one outlier to the other from the distribution and so 3 x IQR was decided upon in order to include most outliers. Figure 2 shows a box plot that was modelled in R studio. As aforementioned, it is seams in the box plots that there are many points that lie outside the fences but models have been adjusted for those. This figure merely shows that traditional outliers (> 1.5 IQR) exists within the data set.

```
Churn <- read.csv("//Users//kevinphan//Desktop//dataset//churn.csv", header = TRUE, na.strings =
c("","NA"), stringsAsFactors = TRUE)

attach(Churn)

grid(20,20, col = "lightgray", lty = "dotted",lwd = par("lwd"), equilogs = TRUE)

boxplot(Account.Length,VMail.Message,Day.Mins,Day.Calls,Day.Charge,

Eve.Mins,Eve.Calls,Eve.Charge, Night.Mins, Night.Calls,

Night.Charge,Intl.Mins,Intl.Calls,Intl.Charge, CustServ.Calls,

names = c("AL","VM","DM","DC","DCH","EM","EC","ECH","NM","NC","NCH","IM","IC","ICH","CUSTS")

,col = c("blue","coral","turquoise","black","magenta","lightgray","yellow","skyblue","orange",

         "green","gray","red","brown","forestgreen"))
```

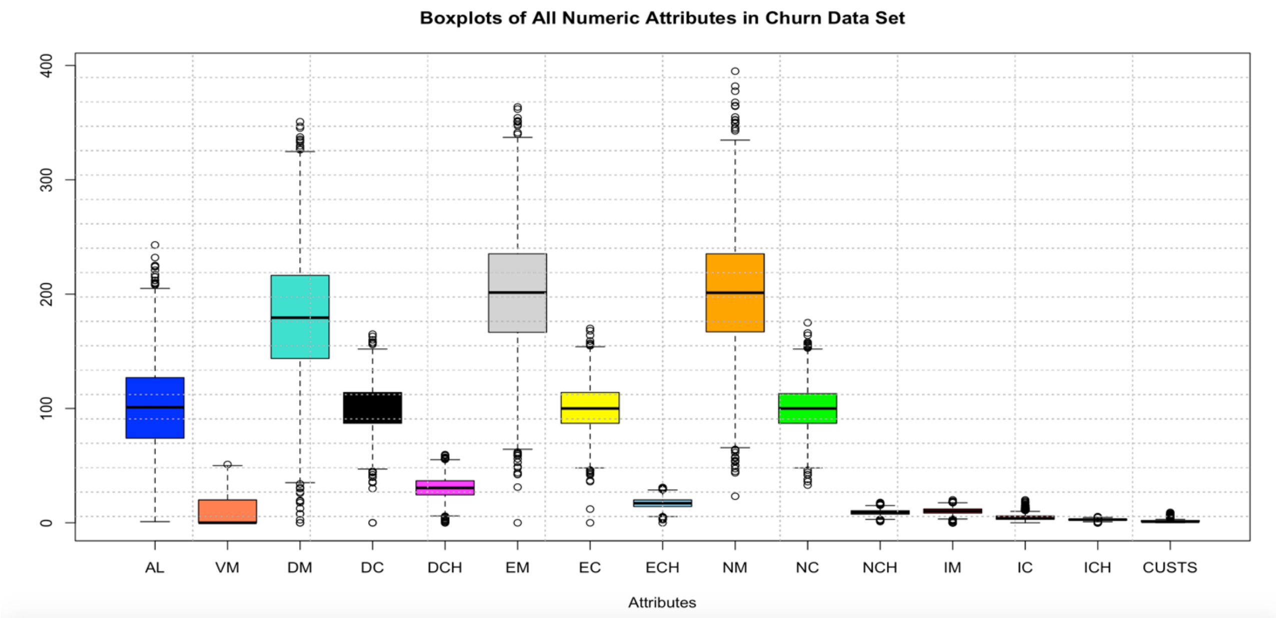
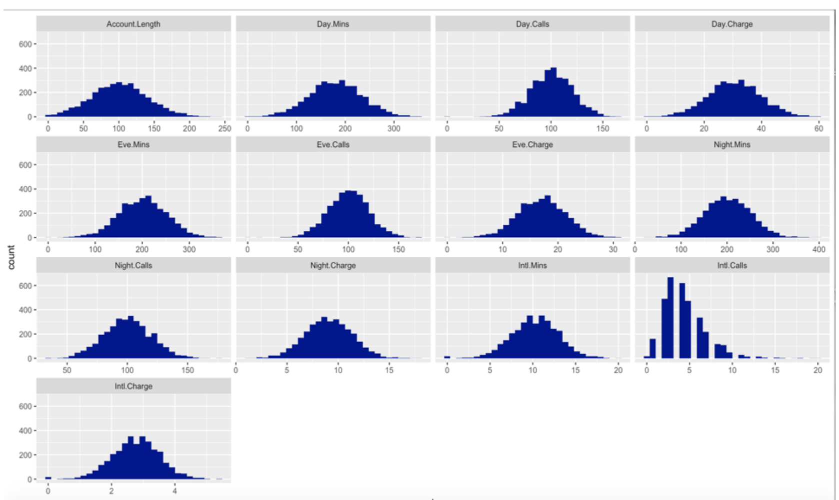
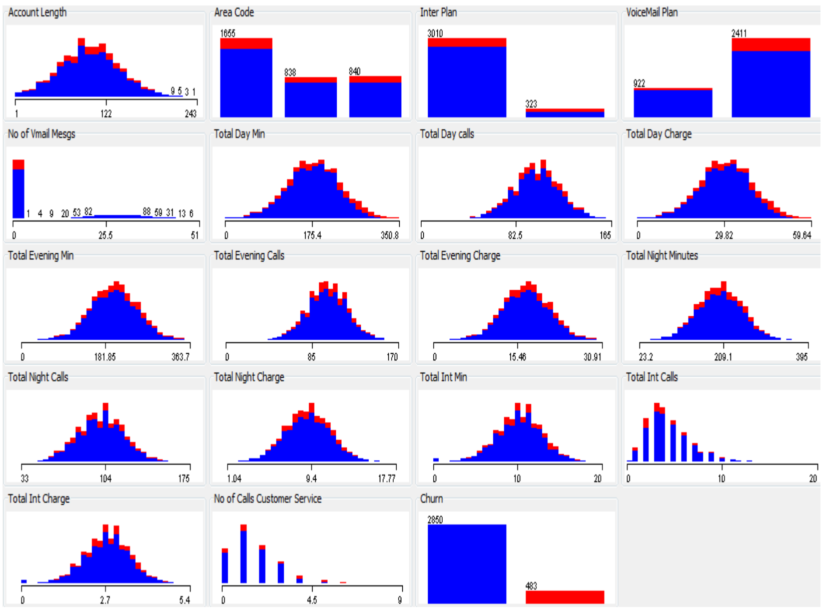


Figure 2

Figure 3 shows the histograms of all attributes simulated in R studio. We do see that most attributes are normal. Normally distributed attributes such as Account Length, Day minutes, Day Charge, Eve Calls, Eve Charge and Intl Min have low scores of kurtosis which is correlated with flat tops ear the mean. We see the rest of the normally distributed attributes to have peak data implications. Needless to say that the number of Cust. Serv Calls is heavily skewed to the right.

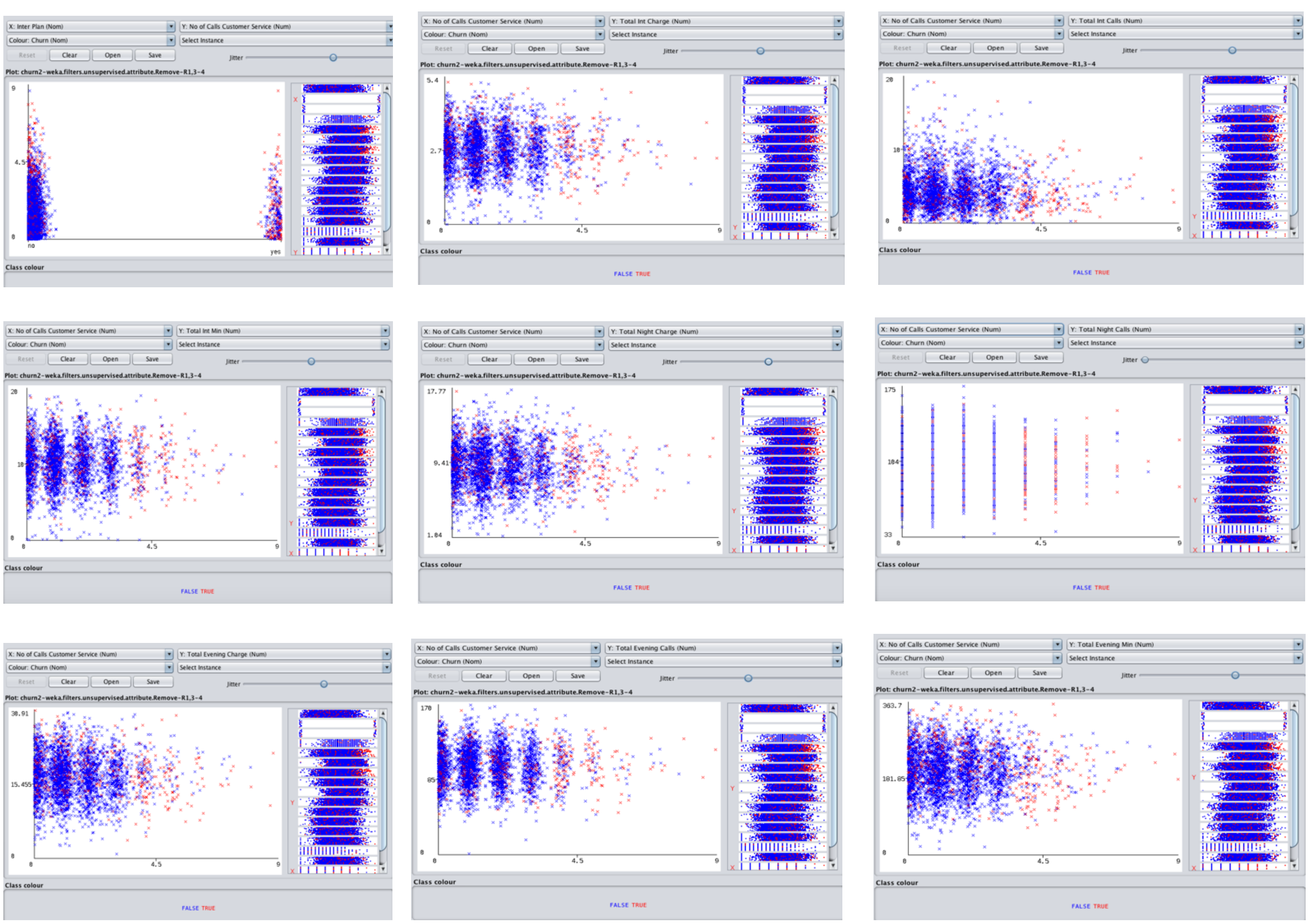


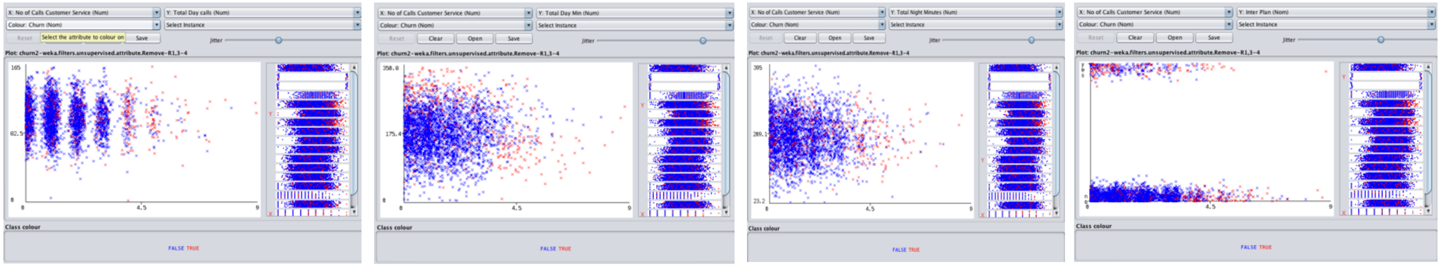



```
install.packages("ggplot2")

library("ggplot2")

d <- melt(Churn[,-c(2:4)])

ggplot(d,aes(x = value)) +

  facet_wrap(~variable,scales = "free_x") +

  geom_histogram(fill = "darkblue")
```

According to histograms in WEKA, we can see by looking at the voicemail plan, it is easy to inference that people with voicemail plans are more prone to churning. We also see in total Cust Serv. Calls around 4-5 are more linked to churning. We also see that total day min and total day charge are highly correlated (not surprising) where higher values in both attributes are linked to churning.

We now see that there are some correlations between attributes but we can further see more correlations by viewing the visualization tab in WEKA. In total, we can state 31 correlations that we can see that would result in Churning = TRUE. We can see that the most prevalent correlation is the no customer service call to any other variable that is being regressed on it. As said in the last figure, we can see that on average, number of customer service calls over 4.5 will be more correlated to churning without causation. The variables follow a simple Cartesian rule set where the variable defined as x is placed horizontally and y place vertically. Red indicates Churn = FALSE and blue is Churn = TRUE. Figure 3 shows the correlations of customer service calls and other variables. Each has the same pattern where instances are red > 4.5.





We also see a few interesting correlations between international plan and voicemail plan as well as total day minutes. People with international plan with high total day mins will be more proned to churning. We also see that having international plan and voicemail plan are more linked to churning. This can only be assumed that the bigger charge as a on the plans have a role to play in that. (figure 4)

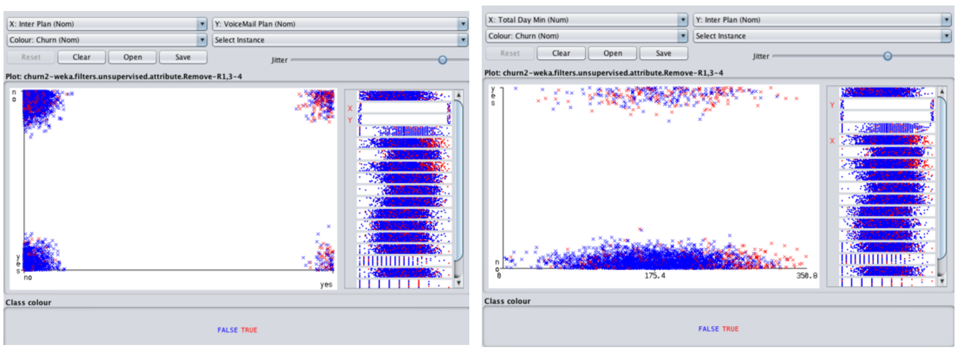


Figure 5

Other Correlations:

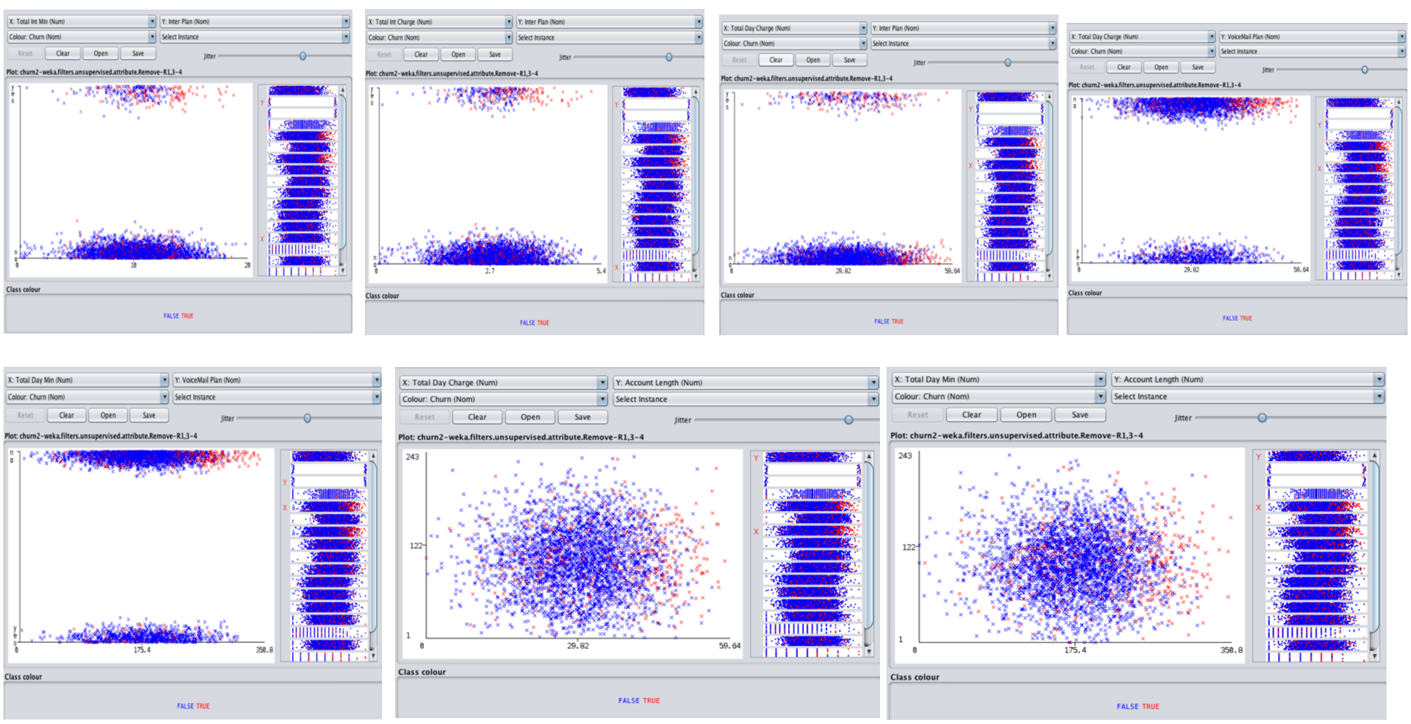


Figure 5.1

From looking at the data set, there are a few attributes that we could remove. The first three are:

- Area, Phone Number & State. (Removed manually due to the assumption that they have no predictive capabilities)

- All charges. We see that the minutes are heavily correlated with their respective charges and that it would only be redundant
- Thus we would be left with 14 variables in which we will use In the classifier algorithms that follow.

table(Churn$Churn.)

```
False.  True.
  2850    483
```

We see that we have an imbalanced class distribution where there are many instances where Churn = False. Significantly more than true which can cause biasness. We will address matters later on in the paper.

Predictive Modeling/Classification

We will first be demonstrating the results of the **Naïve Bayes Classifier**. As we said in the data preparation portion of the analysis, there are certain attributes that could be removed. But the removing of attributes does not always result in better results. Here we decided to pay close attention to 4 measurements. Precision, Recall, F measure and Root mean squared error. We found the value in both precision and Recall and so the F-Measure would allow us to determine the power of a single attribute in predicting our class attribute by giving us a weighted average of both precision and recall. Each set and attribute used a training-test split of 66% and a 10-fold Cross Validation method. We first apply the Naïve Bayes Classifier on the full data set consisting of 21 variables as a bench mark on how the removal of certain attributes can affect the efficiency of the algorithm. We then applied the algorithm on 18 attributes. (Removal of Area code, State and Phone Number.) We did the same with 14 variables by removing the 4 additional charge attributes. This was done with the belief that since these charges were heavily correlated to their corresponding minutes, their elimination would eliminate redundancy within the model. We then run the algorithm on an attribute set of 8. This set (listed below) was decided by WEKA after removing Are code, State and Phone number using weka.filters.supervised.attribute.AttributeSelection. The list was generated through mechanisms within WEKA and cannot be justified by us. We can only assume that these were chosen based on its ability to be able to predict the Class attribute. We then finally arrive at a set of 5 using the same filter without removing the 3 attributes like we did in the former. The reason as to why we have different sets is because we are unsure of which set of attributes in combination are the best predictors of the class attribute and so we attempt for each. All attributes have been discretized using weka.filters.supervised.attribute.Discretize to produce nominal variables out of numeric in order to use naïve bayes.

According to a paper done by Karger and colleagues of the Massachusetts institute of Technology in Cambridge, they state "… when the training data is skewed, the weights will be lower for the class with less training data. Hence, classification will be erroneously biased toward one class over the other". *(Karger, Rennie, Shih & Teevan, 2003)* They suggest using the compliment Naïve bayes which uses more training data per class. We took this information and we decided to replicate this idea by using the resampling filter on WEKA in order to simulate a uniform distribution of all classes and obtain more equality between classes. We try the entire process on each set once again but with each attribute and set resample to replicate a balanced data set. (weka.filters.supervised.Instance.Resample)

8 ATTRIBUTE SET

1. Inter Plan
2. Voicemail Plan
3. Total Day Min
4. Total Evening Min
5. Total Int Min
6. Total Int Calls
7. No of Calls Customer Service
8. Churn

5 ATTRIBUTE SET

1. Phone Number
2. Inter Plan
3. Total Day Min
4. No of Calls Customer Service
5. Churn

Attributes	% Split 66 Correctly Classified (Resample, discretized)	CFV10 (Resample, discretized) Correctly Classified	Precision (Weighted Avg) split/Cross Validation	Recall split/Cross Validation	F-Measure split/Cross Validation	Root Mean Squared	% Split 66 Correctly Classified (Discretized, unbalanced)	CFV10 Correctly Classified (Discretized, unbalanced) CC	Precision split/Cross Validation	Recall split/Cross Validation	F-Measure split/Cross Validation	Root Mean Squared
Full 21	90.37%	91.08%	0.900/0.908	0.904/0.911	0.901/0.909	0.2633/ 0.2592	86.14%	87.36%	0.857/0.869	0.861/0.874	0.859/0.871	0.314/0.3047
18	85.78%	86.64%	0.843/ 0.851	0.858/0.866	0.848/0.856	0.2962/ 0.3002	86.31%	87.78%	0.851/0.865	0.863/0.878	0.856/0.869	0.3063/0.2976
8	85.43%	86.91%	0.824/0.846	0.854/0.869	0.825/0.843	0.2912/ 0.2891	**88.08%**	**88.08%**	**0.865/0.867**	**0.881/0.881**	**0.864/0.858**	**0.2909/0.2875**
14	85.96%	86.79%	0.834/0.844	0.860/0.868	0.824/0.840	0.2984/ 0.2941	87.81	88.32%	0.861/0.869	0.878/0.883	0.861/0.863	0.2898/0.2864
5	**90.11%**	**92.13%**	**0.894/0.917**	**0.901/0.921**	**0.859/0.916**	**0.2616/ 0.2476**	87.90%	87.69%	0.879/0.871	0.879/0.877	0.879/0.874	0.3038/0.300
State	84.81	85.50%	N/A	N/A	N/A	N/A	84.81	85.50%	N/A	N/A	N/A	N/A
Account Length	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Area Code	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Phone Number	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Inter Plan	84.37%	84.87%	N/A	N/A	N/A	N/A	84.37%	84.87%	N/A	N/A	N/A	N/A
Voicemail Plan	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
No Voicemail Msgs	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Total Day Min	85.34%	86.04%	N/A	N/A	N/A	N/A	85.34%	86.04%	N/A	N/A	N/A	N/A
Total Day Calls	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Total Day Charge	85.34%	86.04%	N/A	N/A	N/A	N/A	85.34%	86.04%	N/A	N/A	N/A	N/A
Total Evening min	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Total Eve Calls	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Total Evening Charge	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Total Night Min	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Total Night Calls	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Total Night Charge	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Total Int Min	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Total Int Calls	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
Total Int Charge	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A
No of Calls Customer Service	85.43%	86.13%	N/A	N/A	N/A	N/A	85.43%	86.13%	N/A	N/A	N/A	N/A
Churn	84.81%	85.50%	N/A	N/A	N/A	N/A	84.81%	85.50%	N/A	N/A	N/A	N/A

Figure 6

According to figure 6, we see that overall, k = 10-fold cross Validation works better than the training-test split of 66%. We see that under the Cross-Validation Method, the recall, Precision, F-Measure are higher and the Root Mean Squared Error is lower. The highest attribute is No of Customer Service Calls with 86.13 % under 10-fold Cross Validation. We see that the Attribute Set of 5 performed the best when the attributes were resampled (92.13% Correctly Classified under Cross Validation) and the Attribute Set of 8 performed the best (88.08%) when the Attributes were left unbalanced also under k = 10-Fold-Cross Validation. Out of these two which are highlighted in our results, the set of 5 outperformed the set of 8 having more correctly classified instances in both data partitioning methods (% [92.13>88.08]) and better results on all 4 measurements. (% 0.917>0.867, 0.921 > 0.881, 0.916 > 0.858 and 0.2476 < 0.28]) Intuitively, we can assume that this set of 5 is the best predictor of the Churn attribute and should be used for the following algorithms. We have to realize that the Naïve

Bayes Classifier is a GENERATIVE model. This means that rebalancing the data set may not be a good idea since the Naive Bayes classifier operates on the principal of conditional probabilities of the original generative process. We are trying to infer a class based on the observed values. Therefore, it is better to use the attribute set of 8 rather than 5 as that can be seen more as the true results of the Naïve Bayes Classifier. Figure 6.1 shows the WEKA statstics.

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        2936               88.0888 %
Incorrectly Classified Instances       397               11.9112 %
Kappa statistic                          0.3561
Mean absolute error                      0.1746
Root mean squared error                  0.2875
Relative absolute error                 70.3996 %
Root relative squared error             81.6706 %
Total Number of Instances             3333

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area
                 0.982    0.716    0.890      0.982    0.934      0.405    0.876     0.965
                 0.284    0.018    0.729      0.284    0.408      0.405    0.876     0.615
Weighted Avg.    0.881    0.615    0.867      0.881    0.858      0.405    0.876     0.915

=== Confusion Matrix ===

    a    b   <-- classified as
 2799   51 |    a = FALSE
  346  137 |    b = TRUE
```

Figure 6.1

An Attempt was done in R but the packages and incompatibilities with the current version of R made it difficult to simulate the classifier.

```
Churn <- read.csv("//Users//kevinphan//Desktop//dataset//churn.csv", header = TRUE, na.strings =
c("","NA"), stringsAsFactors = TRUE)

Churn.Revised <- subset(Churn, select = - c(State, Area.Code,Phone,Day.Charge, Eve.Charge,
Night.Charge, Intl.Charge, Account.Length, VMail.Message, Day.Calls, Eve.Calls, Night.Mins,
Night.Calls))

install.packages("caret") install.packages("e1071")

library("caret") library("e1071")

table(Churn.Revised$Churn.)

training.index <- createDataPartition(Churn.Revised$Churn., p = 0.7, list = FALSE)

training.set <- Churn.Revised[training.index,]

test.set <- Churn.Revised[-training.index,]

nrow(training.set) nrow(test.set)

e1071 <- naiveBayes(Churn. ~ ., data = training.set)

prediction <- predict(e1071, test.set)

print(confusionMatrix(prediction,test.set$Churn., positive = "True.", dnn = c("Prediction.",
"True")))
```

The next two machine learning algorithms makes use of discriminative models. We now perform the decision Tree model on the data set. The results are as follows.

Attributes	Cross-validation – 10 Folds Correctly Classified Instances (%)	Percentage split – 66% Correctly Classified Instances (%)

Full Set - 21	93.6994	93.6452
Manual set - 18	94.5395	94.2630
Manual set – 17 (rem 4 charges)	93.6694	93.5569
SA Set - 8	94.6295	94.6161
SA Set - 5	88.5389	88.3495
State	85.5086	85.5252
Account Length	85.5086	85.5252
Area Code	85.5086	85.5252
Phone Number	85.5086	85.5252
Inter Plan	85.5086	85.5252
Voicemail Plan	85.5086	85.5252
Number of Voicemail Messages	85.5086	85.5252
Total Day Min	86.5887	85.9665
Total Day Calls	85.5086	85.5252
Total Day Charge	86.5887	85.9665
Total Evening Min	85.5086	85.5252
Total Evening Calls	85.5086	85.5252
Total Evening Charge	85.5086	85.5252
Total Night Min	85.5086	85.5252
Total Night Calls	85.5086	85.5252
Total Night Charge	85.5086	85.5252
Total Int Min	85.5086	85.5252
Total Int Calls	85.5086	85.5252
Total Int Charge	85.5086	85.5252
No of Calls Customer Service	86.1986	85.1721
Total Day Min & No of Calls Customer Service**	88.5989	88.4378

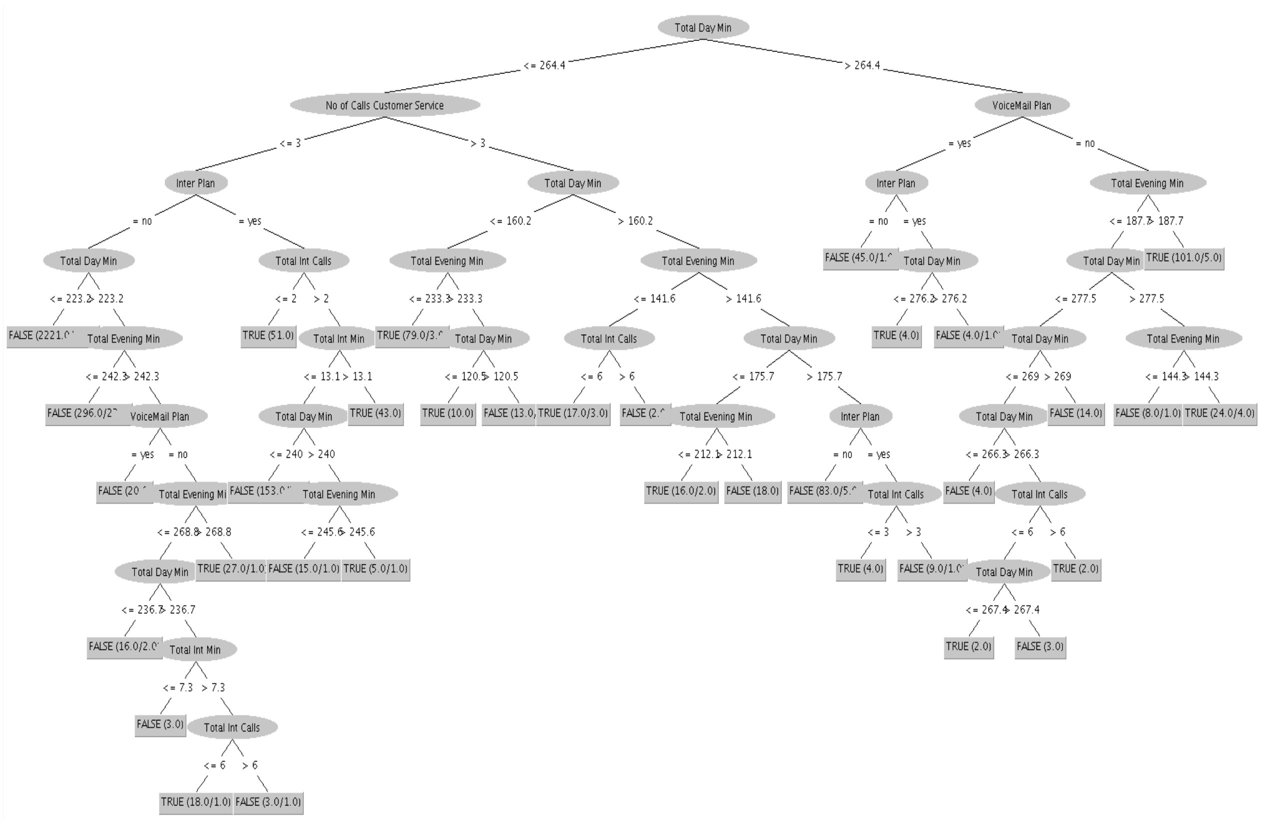
Figure 7

Only 3 attributes (Total Day Min, Total Day Charge, No of Calls Customer Service) produced >85.8086 and >82.5252. Total Day Min and Charge are strongly correlated. The last row is a set of 3 attributes as a result – Total Day Min, No of Calls Customer Service and Churn. To give a more relaxed and spacious analysis as supposed to the last algorithm, we analyze the top 3 sets that have been highlighted red in figure 7.

	Attributes	Correctly Classified Instances (%)	Precision (weighted avg)	Recall (weighted avg)	F-Measure (weighted avg)	Root mean squared error
CV10F	Full Set - 21	93.6994	0.934	0.937	0.934	0.2313
	Manual set - 18	94.5395	0.943	0.945	0.943	0.2254
	SA Set - 8	94.6295	0.944	0.946	0.944	0.2221
66%	Full Set - 21	93.6452	0.934	0.936	0.933	0.2362
	Manual set - 18	94.2630	0.941	0.943	0.940	0.2341
	SA Set - 8	94.6161	0.945	0.946	0.943	0.2205

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        3154               94.6295 %
Incorrectly Classified Instances       179                5.3705 %
Kappa statistic                          0.7653
Mean absolute error                      0.0818
Root mean squared error                  0.2221
Relative absolute error                 32.9864 %
Root relative squared error             63.0856 %
Total Number of Instances             3333

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.984    0.277    0.954      0.984    0.969      0.771    0.882     0.958     FALSE
                 0.723    0.016    0.886      0.723    0.796      0.771    0.882     0.767     TRUE
Weighted Avg.    0.946    0.240    0.944      0.946    0.944      0.771    0.882     0.930

=== Confusion Matrix ===

    a    b   <-- classified as
 2805   45 |    a = FALSE
  134  349 |    b = TRUE
```

figure 8



The Set of 8 with Cross-validation, 10-fold works out to have the highest values for Correctly Classified Instances, Precision, Recall and F-measure and the second lowest root mean squared error value. This testifies to our prior inference that the Cross Fold Validation is a more effective technique in or case than the testing-training set split. Since we had gotten the same outcome in the Naive Bayes Algorithm (The set of 8 having the overall best predictive abilities) we see on comparison that the Set of 8 in the Decision Tree model outperforms the set of 8 in the Naïve Bayes Model by a wide gap. Shown above is the picture for our decision tree as well as the WEKA output statistics that was recorded in the table.

Correctly classified instances: 94.62% > 88.08

Precision: 0.944%> 0.867

Recall: 0.946% > 0.881

F-Measure: 0.944% > 0.858

Root Mean Squared Error: 0.221% < 0.2875

(Decision tree Values being the first in Red)

We can make an inference that for this particular data set, a discriminative model may be a better predictor to infer about the subset of attributes of interest. According to a paper written by Phd. Tesfaye Gudeta from Institute of Technology Tallaght Dublin, Ireland he had tested his data set against the J48 Algorithm and had found that this model had given the best results (94.59%) just second to the Jrip Algorithm while the Naive Bayes Model had given the lowest on average of 85%. (Gudeta, 2013) this can further backup our assumption of the dominance of discriminative models in this case.

We next perform he Random Forest Algorithm. We have established that the set of 8 Attributes under a 10-Fold Cross Validation for the data partition is the best predictor for or class attribute and so we will perform the Random forest Classifier under these parameters. Show is the results in figure 9.

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances        3166               94.9895 %
Incorrectly Classified Instances       167                5.0105 %
Kappa statistic                          0.7815
Mean absolute error                      0.0953
Root mean squared error                  0.2078
Relative absolute error                 38.4322 %
Root relative squared error             59.0253 %
Total Number of Instances             3333

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.986    0.263    0.957      0.986    0.971      0.787    0.910     0.970     FALSE
                 0.737    0.014    0.899      0.737    0.810      0.787    0.910     0.848     TRUE
Weighted Avg.    0.950    0.227    0.948      0.950    0.948      0.787    0.910     0.953

=== Confusion Matrix ===

    a    b   <-- classified as
 2810   40 |    a = FALSE
  127  356 |    b = TRUE
```

Figure 9

In this result, we see that the correctly classified instances are 94.98% which is greater than that of the decision tree. We have decided to leave the data unbalanced. We can see by the confusion matrix that we did correctly classify a lot of FALSE values and we did correctly classify a lot of TRUEs. Since the data is skewed the TRUE values seem a lot smaller but we were precise in precision and recall. We can now state that the Random Forest Classifier is the best model to use for this data set. This also makes sense intuitively. We have a large data set with over 3000 instances. A normal decision Tree tend to have high variances when utilizing different training and test sets on the same data due to overfitting. They are more usefuly for exploratory data analysis rather than a good predictor. A random Forest creates many Trees in which a specific rule set is applied over and over again which makes it a better predictor for our Churn Variable. We can see the Decision tree as a more volatile reactor to changes in say No of Customer Service Calls.

We see from the results that all three algorithms perform their best when the data set is filtered into the 8 Attribute Set whose attributes are listed earlier in the paper under a k=10 Fold Cross Validation. Out of the top three (one from each algorithm) we see that the Random Forest on 8 Attributes under the Cross-Validation Data Partitioning method had the highest precision, recall, Correctly classified instances (94.98%) and lowest Root Mean Squared Error at 0.2078.

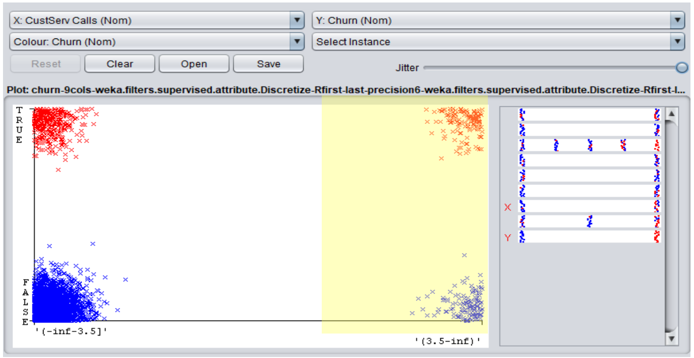
Post-prediction Analysis

Cluster Analysis – Kmeans

In this exercise, we looked at clusters of various attributes such as customer service calls, total charges in relation to churn/retention. We calculated total charges by adding all the charges together.

Plot 1

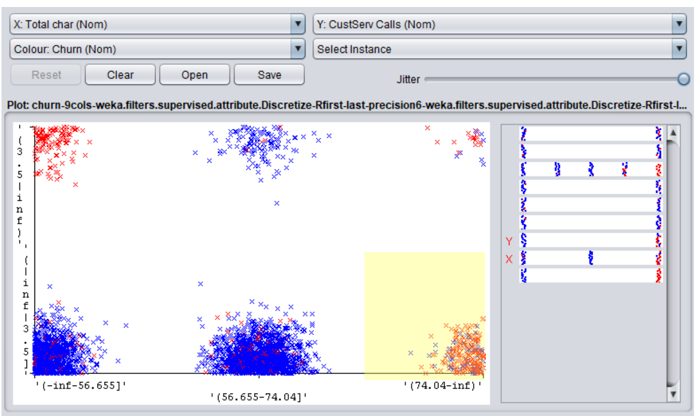
Clusters indicate out of customers making service calls over 3.5 times, almost half of them are churners



Plot 2

Clusters indicate that customer service center is utilized the least by clients in churning customers in high spending group.

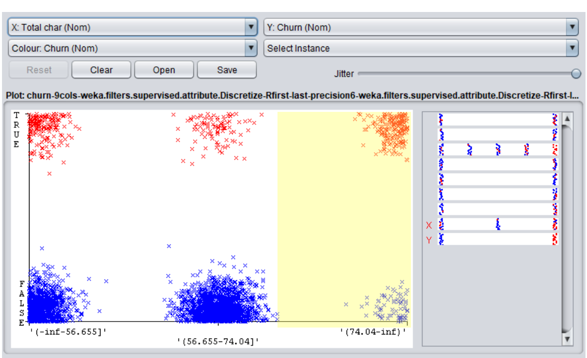
- ➔ Customers in premium spending group leave without expressing their dissatisfaction frequently. Hence, should provide early resolution to improve retention rate.



Plot 3

Clusters indicate considerable number of high value customers (total spending over $74.04) are churning

→ Based on issue found in plot 2, the company should revisit process for complaints handling for high value customers. Also, we suggest customer service data collection in this phase to further analyze retention challenges with high value customers.



Association rule - Apriori

Due to the imbalanced data, there is an accuracy paradox and association rules do not produce meaningful outputs. As shown in the table below, outputs from each churn = TRUE and churn = FALSE result are similar.

Churn = TRUE	Churn = FALSE
2. Intl Mins='(0-13.15) 374 ==> Churn=TRUE 374 conf:(1)	4. Intl Mins='(0-13.15) 2494 ==> Churn=FALSE 2494 conf:(1)
3. Eve Mins='(0-249.15) 358 ==> Churn=TRUE 358 conf:(1)	5. Eve Mins='(0-249.15) 2391 ==> Churn=FALSE 2391 conf:(1)
4. Intl Plan=no 346 ==> Churn=TRUE 346 conf:(1)	8. Intl Plan=no Intl Mins='(0-13.15) 2308 ==> Churn=FALSE 2308 conf:(1)
5. CustServ Calls='(0-3.5) 345 ==> Churn=TRUE 345 conf:(1)	1. CustServ Calls='(0-3.5) 2721 ==> Churn=FALSE 2721 conf:(1)
6. Intl Calls='(2.50)' 344 ==> Churn=TRUE 344 conf:(1)	7. Intl Calls='(2.50)' 2322 ==> Churn=FALSE 2322 conf:(1)
7. VMail Plan=no Intl Mins='(0-13.15) 322 ==> Churn=TRUE 322 conf:(1)	6. Intl Mins='(0-13.15) CustServ Calls='(0-3.5) 2384 ==> Churn=FALSE 2384 conf:(1)
8. Intl Plan=no VMail Plan=no 302 ==> Churn=TRUE 302 conf:(1)	9. Eve Mins='(0-249.15) CustServ Calls='(0-3.5) 2289 ==> Churn=FALSE 2289 conf:(1)
9. VMail Plan=no Eve Mins='(0-249.15) 295 ==> Churn=TRUE 295 conf:(1)	10. Intl Plan=no Eve Mins='(0-249.15) 2235 ==> Churn=FALSE 2235 conf:(1)
10. VMail Plan=no CustServ Calls='(0-3.5) 295 ==> Churn=TRUE 295 conf:(1)	2. Intl Plan=no 2664 ==> Churn=FALSE 2664 conf:(1)
1. VMail Plan=no 403 ==> Churn=TRUE 403 conf:(1)	3. Intl Plan=no CustServ Calls='(0-3.5) 2544 ==> Churn=FALSE 2544 conf:(1)

To work with an imbalanced dataset, we have under sampled retention customers to make the number of retention the same as churns. As a result, we learned that customers without voice mail plan, high day time usage, low customer service calls, high charges are associated with churn.

1. VMail Plan=no Day Mins='(234-inf)' CustServ Calls='(0-3)' Total char='(72-inf)' 178 ==> Churn=TRUE 169 conf:(0.95)
2. VMail Plan=no Day Mins='(234-inf)' Intl Calls='(0-7)' Total char='(72-inf)' 155 ==> Churn=TRUE 147 conf:(0.95)
3. Intl Plan=no VMail Plan=no Day Mins='(234-inf)' Total char='(72-inf)' 165 ==> Churn=TRUE 156 conf:(0.95)
4. VMail Plan=no Day Mins='(234-inf)' Total char='(72-inf)' 194 ==> Churn=TRUE 183 conf:(0.94)
5. VMail Plan=no Intl Calls='(0-7)' Total char='(72-inf)' 183 ==> Churn=TRUE 167 conf:(0.91)
6. Intl Plan=no VMail Plan=no Intl Calls='(0-7)' Total char='(72-inf)' 159 ==> Churn=TRUE 145 conf:(0.91)
7. VMail Plan=no Intl Calls='(0-7)' CustServ Calls='(0-3)' Total char='(72-inf)' 168 ==> Churn=TRUE 153 conf:(0.91)
8. Day Mins='(234-inf)' CustServ Calls='(0-3)' Total char='(72-inf)' 193 ==> Churn=TRUE 175 conf:(0.91)
9. VMail Plan=no CustServ Calls='(0-3)' Total char='(72-inf)' 210 ==> Churn=TRUE 190 conf:(0.9)
10. VMail Plan=no Total char='(72-inf)' 228 ==> Churn=TRUE 206 conf:(0.9)

Conclusions and Recommendations

In conclusion, as our pre-data processes suggests, a Churn value of TRUE is related to the amount of customer service calls being made, the type/number of plans they have and charges associated. Customers that have a high customer service call rater as said before over 4.5 is related to churning. We can see this relationship with any variable. This intuitively makes sense as there is a sense of dissatisfaction manifesting elsewhere that would cause the customers to repeatedly call for assistance. The results of the data also suggest that customers with international plans and arguably with higher minutes in the day are running the risk of churning as there is a high number of TRUE values under those parameters. Customers who also have an international plan and high international minutes. It is also evident form the data that customers with no voicemail plan (see figure 5.1) are linked to churning as well. It is also evident that customers with high total day and International (more so than night and evening are more related to the risk of churning.

The best predictive model is the Random Forest Model.

A few recommendations that can be made to the company are things that can address the characteristic above. The company should develop and call recording system and keep as case studies in order to better solve customer problems the first time they call for service. Problems that customers call in for could be taken into account and resolved if it is a problem on the companies end to prevent other customers from calling in regarding the same problem. It is evident that customers want to make international calls but the surplus charges prevent them from doing a such. The company should invest in long term deals with long distances offices to reduce international charges. The company should also hold promotions for voicemail plans as an incentive for the loyalty of those who stay.