Indian Institute of Technology, Kanpur

IME672A

# Project Report

*Authors:*
Vishal Meena
Gaurav Kumar
Harsh Singh
M Kalyan

*Instructor:*
Dr. Hamid

**Abstract**

Customer churn is a recurrent problem in the telecommunications industry, with 2-4% of customers changing brands every month. The following project aims to take data from a telecommunications company and use it to train classification models that will be able to predict future churn behavior of customers. The concepts of Data Preprocessing, Classification, and Ensemble Methods are applied for the same.

# Contents

# 1 Overview

## 1.1 The Problem

To avoid customer churn, the project aims to accurately predict whether a customer is going to churn soon or not provided some information about him. The following sections will provide more insight into how the same is achieved.

## 1.2 The Data

The customer data provided consists of 71047 records of 78 attributes in a csv file. Of the attributes, CALIBRAT separates the training data from the test data, and the other 77 attributes give details about the customers. The attributes belong to 3 major logical categories:

- Attributes related to usage (REVENUE, MOU, ROAM etc.)
- Personal Attributes (AGE, INCOME, CHILDREN etc.)
- Attributes related to previous company contact (MAILORD, RETCALLS etc.)

The attributes can also be differentiated based on the type of data they contain. Namely:

- 37 Numerical Attributes (INCOME, REVENUE etc.)
- 40 Binary Attributes (CREDITA, CREDITAA etc.)

## 1.3 Initial Impressions

The data seemed suggestive, but not usable straightaway, especially given the large number of records that had all the binary attributes used to store nominal/ordinal variables zero, meaning that their value was unknown.

# 2 Data Preprocessing

Since Data Preprocessing has already been covered in Assignment 1, it will only be covered briefly, with only areas of interest receiving a detailed treatment. Statistical analysis of attributes is skipped altogether. By condensing binary attributes itself, we are left with only 58 attributes to deal with.

## 2.1 Basic Transformations

The csv file is imported in the R console with the simple command:

```
CELL <- read.csv("Cell2Cell.csv")
```

With the file read, we can manipulate the attributes as needed. Something that makes future handling of the data extremely easy is converting it completely into a numeric format without any loss of information, achieved by:

```
CELL = as.data.frame(sapply(CELL,as.numeric))
```

Note that even string attributes like CSA can be converted without losing out any information.

## 2.2 Handling Missing Values

The following table illustrates the number of missing values in the training data set alone (40000 records). Notice how variables like OCC, PRIZM, and MARRY have a very high number of missing values.

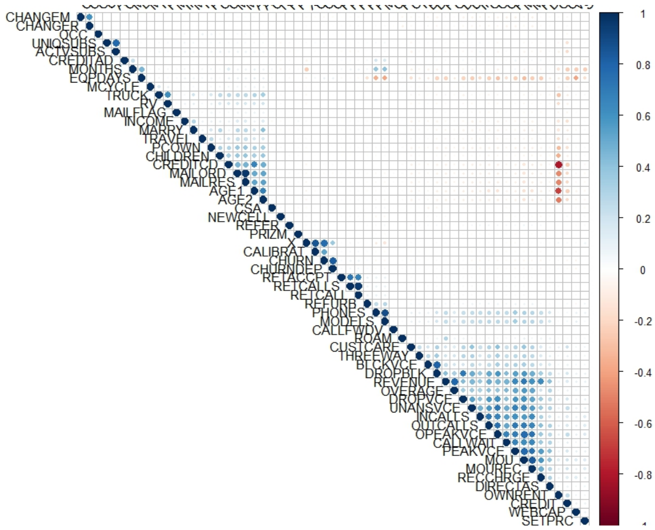
| | | | | | | | |
|---|---|---|---|---|---|---|---|
| RECCHRGE | DIRECTAS | OVERAGE | ROAM | CHANGEM | CHANGER | DROPVCE | BLCKVCE |
| 141 | 141 | 141 | 141 | 361 | 361 | 0 | 0 |
| THREEWAY | MOUREC | OUTCALLS | INCALLS | PEAKVCE | OPEAKVCE | DROPBLK | CALLFWDV |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| MONTHS | UNIQSUBS | ACTVSUBS | PHONES | MODELS | EQPDAYS | AGE1 | AGE2 |
| 0 | 0 | 0 | 0 | 0 | 0 | 706 | 706 |
| WEBCAP | TRUCK | RV | OWNRENT | MAILORD | MAILRES | MAILFLAG | TRAVEL |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RETCALLS | RETACCPT | NEWCELLY | NEWCELLN | REFER | INCOME | MCYCLE | CREDITAD |
| 0 | 0 | 0 | 0 | 0 | 10106 | 0 | 0 |
| CHURNDEP | OCC | CREDIT | PRIZM | MARRY | | | |
| 0 | 29395 | 0 | 19283 | 15534 | | | |

To avoid using variables with very high number of missing values, we decided to drop the variables altogether whose number of misses was above 10000. For the other variables, we replaced the missing values with the mean values.

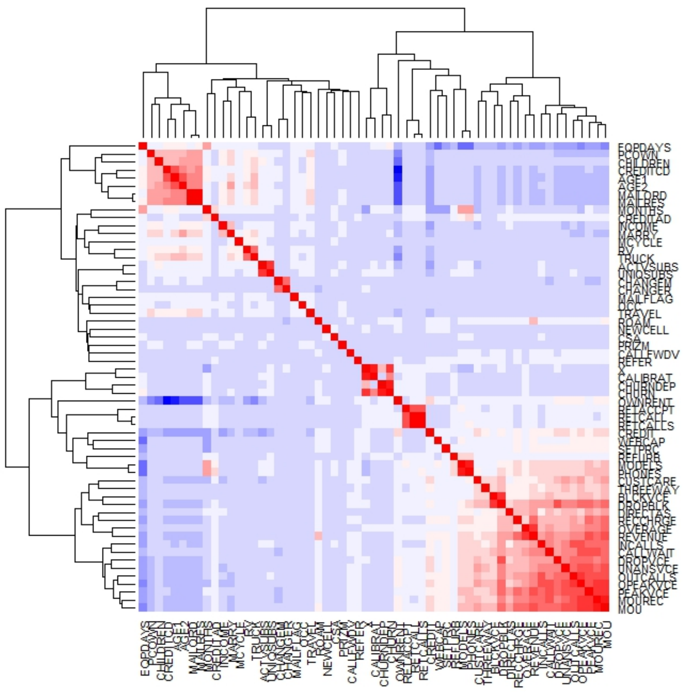
```
for(i in 1:ncol(CELL))
        CELL[is.na(CELL[,i]),i]<-mean(CELL[,i],na.rm =TRUE)
```

## 2.3 Visualizing Correlation among Attributes

Highly correlated attributes increase time taken to implement the data mining task without providing additional valuable information, and thus should be removed before applying algorithms on data. The following heat maps try to give an intuitive image of the positively and negatively correlated attributes.



```
library(Hmisc)
flattenCorrMatrix<-function(cormat, pmat)
{
        ut<-upper.tri(cormat)
        data.frame(row = rownames(cormat)[row(cormat)[ut]],column = rownames(corm
}
matrix<-rcorr(as.matrix(CELL[,1:59]))
flattenCorrMatrix(matrix$r, matrix$P)
library(corrplot)
simple = cor(CELL)
corrplot(simple, type = "upper", order="hclust", tl.col="black", tl.srt=45)
col<- colorRampPalette(c("blue", "white", "red"))(20)
heatmap(x = simple, col = col, symm = TRUE)
```



## 2.4 Dimensionality Reduction using Rough Sets

Having already gone over noise and outlier handling in Assignment 1, let us now employ an advanced step in data reduction, using rough set theory.

```
library(sets)
library(class)
library(RoughSets)
decision.table<-SF.asDecisionTable(CELL,decision.attr = 23)
#Since index of class label is 23 (CHURN attribute)
var<-D.discretize.equal.intervals.RST(decision.table, nOfIntervals = 30)
decision.table<-SF.applyDecTable(decision.table,var,control=list())
res.1<-FS.greedy.heuristic.superreduct.RST(decision.table,qualityF=X.nOfConflicts
res.2<-FS.quickreduct.RST(decision.table,control=list())
decision.table<- SF.applyDecTable(decision.table, res.1) #Or whatever res you wan
```

Rough sets are different from crisp (normal) sets in terms of the fact that an object can belong to it in a degree more than nothing and less than completely belonging to it. A reduct is a subset of the attributes that contains the entire information of the data. For example, say you had $A_i \subseteq A$. Now each record could be classified into a partition of equivalence classes where two records belonged to the same class if their values for all variables in $A_i$ were the same. This is the same as saying that $A_i$ does not provide enough information to differentiate the two. If the equivalence classes thus divided have all their records share the same class label, we have achieved a reduct $A_i$. Now, in reality, this is hardly the case. So it is possible that there are two partitions which both have approximately the same equivalence classes, which means the attributes while not identical are dispensable. Thus we can remove some attributes without reduction in predicting power. The reduct we calculated was {CHURN,REVENUE,MOU,RECCHRGE,CHANGEM,DROPVCE,UNANSVCE,OUTCALLS,OPEAKVCE,UNIQSU

## 2.5 Training Set and Test Set

The data is divided into training and test datasets of 40000 and 31047 records each according to values of the CALIBRAT attribute. All models will be trained on the Training dataset and validated on the Test set.

```
trainingCELL=subset(CELL,CELL$CALIBRAT==1)
> testCELL=subset(CELL,CELL$CALIBRAT==0)
```

Table 1: Training Dataset

| Actual/Predicted | 0 | 1 |
|---|---|---|
| 0 | 11488 (TN) | 8456 (FP) |
| 1 | 8512 (FN) | 11544 (TP) |

Table 2: Test Dataset

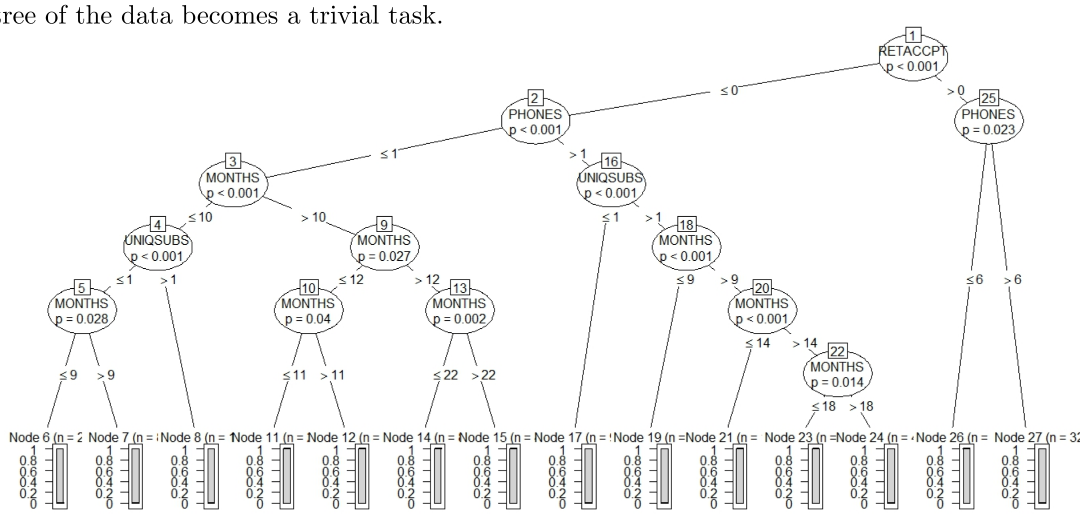
| Actual/Predicted | 0 | 1 |
|---|---|---|
| 0 | 17560 (TN) | 255 (FP) |
| 1 | 9810 (FN) | 354 (TP) |

# 3 Classifiers

Now that we have our data in readily usable format, this section will go through the different classifiers we built to predict churn values.

## 3.1 Decision Tree

Decision trees conducted from data use condition based nodes to divide the data into smaller subsets housed at leave nodes classified into the same class. Using the reduct calculated, making a decision tree of the data becomes a trivial task.




```
library(party)
tree<-ctree(CHURN˜MONTHS+UNIQSUBS+PHONES+RETACCPT,data=trainingCELL)
print(tree)
plot(tree)
Prediction<-predict(tree,trainingCELL, type="node")
table(Prediction,trainingCELL$CHURN)
Prediction<-predict(tree, testCELL, type="node")
table(Prediction, testCELL$CHURN)
```

The tree was tested on both the training and test data set. The results for both are displayed below: Similiarly, a Decision tree can also be made by the rpart package, generating a subtly different tree compared to party.

```
library(rpart)
tree<-part(CHURN˜REVENUE+MOU+RECCHRGE+CHANGEM+DROPVCE+UNANSVCE+OUTCALLS+OPEAKVCE+
printcp(tree)
plot(tree,uniform=TRUE,main="Classification␣Tree␣for␣CHURN")
text(tree,use.n=TRUE,all=TRUE,cex=0.8)
pred<-predict(tree,trainingCELL,type="class")
table(pred,trainingCELL$CHURN)
pred<-predict(tree,testCELL,type="class")
```

Table 3: Comparing Decision Tree Performance

| | Training (party) | Test (party) | Training (rpart) | Test (rpart) |
|---|---|---|---|---|
| Accuracy | 57.8% | 64.2% | 55.1% | 58% |
| Sensitivity | 57.6% | 3.5% | 55.4% | 2.4% |
| Specificity | 57.6% | 98.6% | 54.8% | 98.3% |

```
table(pred,testCELL$CHURN)
```

Table 4: Naive Bayes Performance

| | Training | Test |
|---|---|---|
| Accuracy | 53.98% | 57.80% |
| Sensitivity | 3.71% | 4.28% |
| Specificity | 96.43% | 97.98% |

## 3.2 Naive Bayes Classifier

A Naive Bayes classifier works on the famous Bayes theorem from probability. The probability of a particular record belonging to a certain class can be calculated using the probability of having the same combination of attribute values given a particular class. The naive in the name comes from the rather simple approximation that assumes that attributes take values independent of each other.

```
library(class)
library(e1071)
testCELL$CHURN<-as.factor(testCELL$CHURN)
trainingCELL$CHURN<- as.factor(trainingCELL$CHURN)
model<- naiveBayes(CHURN~REVENUE+MOU+RECCHRGE+CHANGEM+CHANGER+DROPVCE+BLCKVCE+UNA
pred1<-predict(model, testCELL)
table(pred1,testCELL$CHURN)
NBPrediction <- pred1
pred2<- predict(model, trainingCELL)
table(pred2, trainingCELL$CHURN)
```

## 3.3 Logistic Regression Model

Logistic regression model measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic function. The model of logistic regression is based on quite different assumptions from those of linear regression. In particular, the predicted values are probabilities and are therefore restricted to (0,1). A logistic regression model was trained to predict the CHURN values.

```
trainingCELL$CHURN<-as.factor(trainingCELL$CHURN)
model<-glm(CHURN~REVENUE+MOU+RECCHRGE+CHANGEM+DROPVCE+UNANSVCE+OUTCALLS+OPEAKVCE+
trainingCELL$CHURN=as.numeric(trainingCELL$CHURN)
pred<-ifelse(predict(model,type="response", trainingCELL)>0.5,1,0)
table(pred,trainingCELL$CHURN)
pred<-ifelse(predict(model,type="response",testCELL)>0.5,1,0)
table(pred,testCELL$CHURN)
```

Table 5: Performance of Logistic Regression

| | Training | Test |
|---|---|---|
| Accuracy | 56.3% | 54.9% |
| Sensitivity | 56% | 2.5% |
| Specificity | 56.5% | 98.5% |

Table 6: My caption

| | Training | Test |
|---|---|---|
| Accuracy | 63.4% | 55.0% |
| Sensitivity | 62.7% | 2.6% |
| Specificity | 64.2% | 98.6% |

```
Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  5.412e-01  6.492e-02   8.336  < 2e-16 ***
REVENUE      4.321e-03  3.774e-04  11.447  < 2e-16 ***
MOU         -3.168e-04  4.093e-05  -7.740 9.97e-15 ***
RECCHRGE    -6.855e-03  5.834e-04 -11.751  < 2e-16 ***
CHANGEM     -3.210e-04  4.031e-05  -7.963 1.68e-15 ***
DROPVCE      7.571e-03  1.558e-03   4.858 1.19e-06 ***
UNANSVCE     3.754e-04  4.006e-04   0.937   0.3487
OUTCALLS     1.848e-05  4.856e-04   0.038   0.9696
OPEAKVCE    -3.685e-04  2.265e-04  -1.627   0.1036
UNIQSUBS     1.870e-01  1.951e-02   9.582  < 2e-16 ***
ACTVSUBS    -2.179e-01  2.686e-02  -8.114 4.89e-16 ***
CSA          8.756e-05  4.768e-05   1.836   0.0663 .
MODELS      -1.068e-01  1.270e-02  -8.407  < 2e-16 ***
AGE1        -5.099e-03  6.399e-04  -7.968 1.61e-15 ***
REFURB       2.680e-01  3.019e-02   8.880  < 2e-16 ***
CREDITCD     2.341e-02  2.946e-02   0.795   0.4268
REFER       -4.266e-02  4.116e-02  -1.037   0.3000
CREDIT      -5.646e-02  6.983e-03  -8.086 6.16e-16 ***
MARRY        3.467e-02  2.715e-02   1.277   0.2017
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

### 3.4 SVM

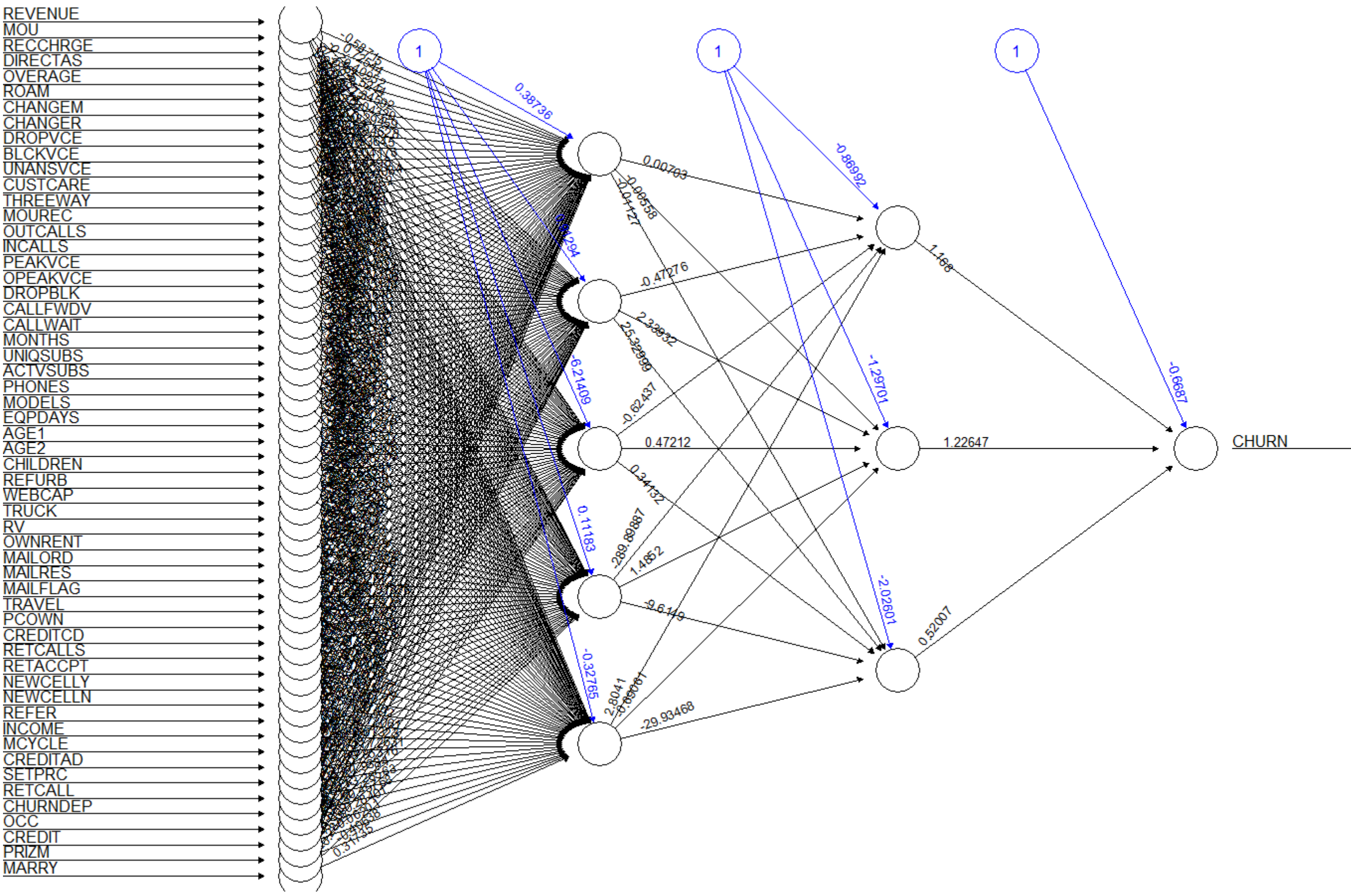
Support Vector Machines or SVMs is another concept to classify the given data. For implementing an SVM, the given data can be plotted in an n-dimensional space, with n being the number of independent attributes and hyperplanes drawn to divide the plotted data into two clusters.The points closed to the hyperplane are known as support vectors.

```
library(class)
library(e1071)
model<-svm(CHURN~REVENUE+MOU+RECCHRGE+CHANGEM+CHANGER+DROPVCE+BLCKVCE+UNANSVCE+MO
pred1<-predict(model,trainingCELL)
table(pred1,trainingCELL$CHURN)
pred1<-predict(model,testCELL)
table(pred1,testCELL$CHURN)
```

### 3.5 Neural Networks

Neural Networks are mathematical models that are supposed to be capable of modeling any function giving enough training examples. Using the nifty back-propagation algorithm, neural networks are inspired from human brains. They consist of nodes (or neurons) with input and output streams. The input and output nodes are separated by one or more "hidden" layers, where the real computation takes place and weights to the in and out channels are assigned. For fitting the data we took a neural network with 2 hidden layers, one of 5 neurons and other of 3.

```
maxs<-apply(CELL, 2, max)
mins<-apply(CELL, 2, min)
scaled<-as.data.frame(scale(CELL,center=mins,scale=maxs- mins))
library(neuralnet)
n <- names(trainscaled)
f <- as.formula(paste("CHURN␣~", paste(n[!n %in% "CHURN"], collapse = "␣+␣")))
nn <- neuralnet(f,data=trainscaled,hidden=c(5,3),linear.output=T)
plot(nn)
```




# 4 Ensemble Methods

An ensemble of all the aforementioned predictive models was created to boost the accuracy. We incorporated the prediction results from
a). Decision Tree based classification
b). Logistic Regression Model
c). Naive Bayes Classification
d). Support Vector Machines
e). Neural Networks
into an ensemble predictive model to give the results.

```
NBPrediction <- as.numeric(NBPrediciton)
```

Table 7: My caption

| | Test |
|---|---|
| Accuracy | 68.32% |
| Sensitivity | 2.6% |
| Specificity | 98.6% |

```
DTPrediction <- as.numeric(DTPrediction)
for (i in 1: nrow(DTPrediction)){
        ensemble[i,1] <- (DTPrediction[i,1] + NBPrediction[i,1] + SVMPrediction[i
}

for (i in 1:nrow(ensemble)){
        ensemble[i,1] <- ensemble[i,1]/5
}

for (i in 1:nrow(ensemble)){
        ifelse(ensemble[i,1] > 0.5, ensemble[i,1] <- 1, ensemble[i,1] <- 0)
}
```

# 5 Questions for Discussion

## 5.1 Describe your predictive churn model. What statistical technique did you use and why? How did you select variables to be included in the model? Is your model adequate? Justify.

As a part of this project, we have analyzed different predictive models, namely:
a). Decision Tree based classification
b). Logistic Regression Model
c). Naive Bayes Classification
d). Support Vector Machines
e). Neural Networks
As the raw data, as supplied in Cell2Cell.csv consisted of 78 attributes and a considerable number of them were redundant, data pre-processing had an important role. Several of the binary variables were combined to give a single piece of information, which reduced the 78 attributes to 58 attributes. Attributes reduction using Rough Sets theory was carried out to remove the redundant attributes:. We have used a heuristic way to find the attributes that were important for the predictive churn model.
By using this approach we were left with 21 attributes, namely: X, REVENUE, MOU, RECCHRGE, CHANGEM, DROPVCE, UNANSVCE, OUTCALLS, OPEAKVCE, UNIQSUBS, ACTVSUBS, CSA, MODELS, EQPDAYS, AGE1, REFURB, CREDITCD, REFER, CHURNDEP, CREDIT, MARRY
We believe that Ensemble method is one of the most efficient way to construct a predictive model. The added advantage is due to the use of multiple algorithms to generate better predictive performance than could be obtained from any of the constituent learning algorithms. The predictions from all the models

## 5.2 Demonstrate the predictive performance of the model. Is the performance adequate?

Ensemble methods have been among the most influential method in Data Mining. There ability to combine multiple models into one usually more accurate than the best of its components by using Ensemble methods, to develop a predictive model we found that the accuracy of our model is 68 percent on test dataset. This amount of accuracy on such a database is significantly enough and is higher from that of decision tree, naive bayes or the SVM.the comparison of ensemble methods with other are shown below:
Ensemble Methods- 68 percent

Decision tree- 58.85 percent
SVM- 54.95 percent
Naïve Bayes-57.80 percent
Logistic Regression Model- 54.85 percent

### 5.3 What are the key factors that predict customer churn? Do these factors make sense? Why or why not?

By using Logistic regression the relationship between the categorical dependent variable and one or more independent variables can be established by estimating probabilities using a logistic function. Using Training Data to construct LRM gives following Result:

```
Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)  5.412e-01  6.492e-02   8.336  < 2e-16 ***
REVENUE      4.321e-03  3.774e-04  11.447  < 2e-16 ***
MOU         -3.168e-04  4.093e-05  -7.740 9.97e-15 ***
RECCHRGE    -6.855e-03  5.834e-04 -11.751  < 2e-16 ***
CHANGEM     -3.210e-04  4.031e-05  -7.963 1.68e-15 ***
DROPVCE      7.571e-03  1.558e-03   4.858 1.19e-06 ***
UNANSVCE     3.754e-04  4.006e-04   0.937   0.3487
OUTCALLS     1.848e-05  4.856e-04   0.038   0.9696
OPEAKVCE    -3.685e-04  2.265e-04  -1.627   0.1036
UNIQSUBS     1.870e-01  1.951e-02   9.582  < 2e-16 ***
ACTVSUBS    -2.179e-01  2.686e-02  -8.114 4.89e-16 ***
CSA          8.756e-05  4.768e-05   1.836   0.0663 .
MODELS      -1.068e-01  1.270e-02  -8.407  < 2e-16 ***
AGE1        -5.099e-03  6.399e-04  -7.968 1.61e-15 ***
REFURB       2.680e-01  3.019e-02   8.880  < 2e-16 ***
CREDITCD     2.341e-02  2.946e-02   0.795   0.4268
REFER       -4.266e-02  4.116e-02  -1.037   0.3000
CREDIT      -5.646e-02  6.983e-03  -8.086 6.16e-16 ***
MARRY        3.467e-02  2.715e-02   1.277   0.2017
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The following attributes are found to be more important in deciding customer churn(High absolute z value and low p value) :
Revenue,Mean monthly minutes of use,percentage change in minutes of use,Mean no of dropped voice calls,No of usinq and active subs,Models issued,Age,Refurbished headset,Credit Rating
Credit rating , percentage change in minutes of use ,MOU and Mean no of dropped voice calls can be intuitively understood to be linked to churn as those who cannot afford to pay an increase in bill tend to churn to maintain balance .
Revenue has a deeper meaning of dependence as "Increased revenue implies more charging on customers "
The other attributes though contribute more to churning of customers as evident from regression cannot be understood intuitively but as a whole.

## 5.4 What offers should be made to which customers to encourage them to remain with Cell2Cell? Assume that your objective is to generate net positive cash flow, i.e., generate additional customer revenues after subtracting out the cost of the incentive.

The goal is to retain the customer. This should the topmost priority of any company.Here we suggest some way to reduce the customer churn:

1-One way is is addressing to the customers problem efficiently and effectively. Wireless-telecom industry involves a list of problem such as slow/down network, billing errors etc. To achieve a good relationship with customers one need to keep track of what most customer have asked about. The second thing that a company can use is to develop a system where a customer can know the way the company is using to resolve his/her query, the tentative date by which the problem will get fixed and how to provide the contact details of right person for further inquiry.
2- Make use of cross-selling: An important telecom outbound message is the one that gets customers to sign up for an additional product. Surveys have shown that the retention rate of a customer in a telecom industry is a function of the number of products that a customer is buying from your company. For instance,by selling a landline connection to a broadband customer, you make a profit from the broadband, but you also reduce the likelihood of churn by this particular customer.

3-The other way is responding wisely to the customers e-mail.The e-mail sent to the customer must be relevant to what the customer might be interested in and just not for the sell purpose.

## 5.5 Assuming these actions were implemented, how would you determine whether they had worked?

Loyalty Programs and Technology

Finally, loyalty programs have come a long way since the introduction of the simple punch card you handed out to your customers. Today you can manage your program through social media and smartphone apps, or a customer management system that allows you to track buyer behaviors and connect with customers through personalized emails or SMS marketing. The data you collect from your customer incentive program can also help you to cross promote or up sell additional products or services and create highly targeted and relevant marketing campaigns to further improve your business.