PROJECT REPORT

**Abstract** - Telecommunication market is expanding day by day. Companies are facing a severe loss of revenue due to increasing competition hence the loss of customers. Conventional statistical methods (e.g. logistics regression, decision tree, and etc.) are very successful in predicting customer churn. The objective of this project is to predict customer churn by using data from a leading telecommunications company (Vodafone) provided to us. We were expected to develop a model to predict the churn score based on usage pattern. The findings from this project are helpful for the company to optimize their customer retention and/or treatment resources in their churn reduction efforts.

**Introduction** - In the telecommunication industry, customers are able to choose among multiple service providers and actively exercise their rights of switching from one service provider to another. In this fiercely competitive market, customers demand tailored products and better services at lesser prices, while service providers constantly focus on acquisitions as their business goals. Given the fact that the telecommunications industry experiences an average of 30-35 percent annual churn rate and it costs 5-10 times more to recruit a new customer than to retain an existing one, customer retention has now become even more important than customer acquisition. With retention strategies in place, many companies start to include churn reduction as one of their business goals. Churn management is very important for reducing churns as acquiring a new customer is more expensive than retaining the existing ones. Churn rate is the measurement for the number of customers moving out and in during a specific period of time. If the reason for churning is known, the providers can then improve their services to fulfill the needs of the customers. Analyzing the past history of the potential customers systematically can reduce churns. Telecom companies maintain large amount of information for each of their customers that keeps on changing rapidly due to competitive environment. This information includes the details about billing, calls and network data. The huge availability of information arises the scope of using Data Mining techniques in the telecom database. The information available can be analyzed in different perspectives to provide various ways to the operators to predict and reduce churning. Only the relevant details are used in analysis, which contribute to the study from the information given.

**Dataset** - The dataset provided to us has Account Related data, Disconnection data, Request data and the data of usage pattern of customers for ten discontinuous months. In those months, each month was classified into 3 data files: data, pay and usage.

Data of a particular month contains columns: ‘id’, ‘VAR2’, ‘VAR3’, ‘vol2g’, ‘vol3g’. User id gives information regarding the usage pattern of customers in terms of 2g and 3g.

Pay of a particular month contains columns: ‘id’, ‘cust\_value\_seg’, ‘cust\_group’, ‘section’, ‘acct\_actv1’, ‘mob\_actv1’, ‘var15a’ (denotes Total\_Bill), ‘var16a’ (denotes RentalCharge), ‘var17a’ (denotes NonRentalCharge), ‘var18a’ (denotes Adjacent), ‘var19a’ (denotes Usage).

Pay gives information of users and their payment pattern and classifies them on the basis of their activation date, customer segment, customer group, Total\_Bill, RentalCharge, NonRentalCharge, Adjacent, and Usage accordingly.

Usage of a particular month contains columns: ‘id’, ‘var2a (denotes Local Outgoing Calls on same network per minutes of usage)’, ‘var3a (denotes STD Outgoing Calls on same network minutes of usage)’, ‘var4a (denotes Local Outgoing Calls to another network minutes of usage)’, ‘var5a (denotes STD Outgoing Calls to another network minutes of usage)’, ‘var6a (denotes STD Outgoing Calls minutes of usage)’, ‘var9a (denotes ISD Outgoing Calls minutes of usage)’, ‘var10a (denotes Total Outgoing Calls minutes of usage)’, ‘var11a (denotes Local Incoming Calls on the same network minutes of usage)’, ‘var12a (denotes STD Incoming Calls on the same network minutes of usage)’, ‘var13a (denotes Local Incoming Calls to another network minutes of usage) ’, ‘var14a (denotes STD Incoming Calls to another network minutes of usage) ’, ‘var15a (denotes STD Incoming Calls minutes of usage)’, ‘var16a (denotes ISD Incoming Calls minutes of usage) ’, ‘var17a (denotes Roaming on Outgoing Calls minutes of usage)’, ‘var18a (denotes Roaming on Incoming Calls minutes of usage) ’, ‘var19a (denotes Total Incoming Calls minutes of usage)’

Usage gives information about calling pattern of customers. E.g.: Local Calls, STD Calls, ISD Calls, Roaming Calls, Incoming Calls, Outgoing Calls, Calls on same network, Calls to another network, etc.

Account Related file contains columns: ‘id’, ‘cust\_value\_seg’, and ‘cust\_group’.

Account Related gives information about the user id’s of customers, their segmentation and their groups.

(We didn’t take Account Related information because the columns were repeating in pay data file of each month).

Disconnection file contains only disconnected date column. It gives information about the date on which the customer churned.

Request contains two data files: Request1 and Request2. It contains columns: ‘id’, ‘VAR2 (it contains factor variables which give information about Request, Query, Complaint, Feedback)’, ‘date2 (gives information about the dates on which a specific customer gave a feedback, made a request, registered a complaint, etc.)’

Request gives information about various customers who made Requests, Registered Complaints, Gave Feedbacks, and Asked Queries on particular dates.

We only worked on the data of five months, which were continuous, i.e. July - November.

**Methodology** - Tools Used:

1. Python, R, RHadoop

* Pre Processing of Data
* Processing of Data

1. MS Word

* Report Creation
* ROC Curve Plotting

1. MS Excel

* Database

We performed Data Cleaning (Pre Processing) in Python and rest of the analysis using R and RHadoop on the server.

Data Cleaning Process in Python:

1. Libraries Imported:
2. Pandas – Pandas Library provides high – performance, easy to use data structures and data analysis tools for the Python Programming language. It is used for structured data operations and manipulations. It is extensively used for data munging and preparation.
3. NumPy – NumPy stands for numerical Python. The most powerful feature of NumPy is n-dimensional array. This library also contains basic linear algebra functions, Fourier transforms, advanced random number capabilities and tools for integration with other low level languages like C and C++.
4. Defined Functions:
5. First we defined a function (mis\_na) to find missing values.
6. Then we defined another function (get\_median\_filtered) for transforming outliers.
7. Fetching Data and Cleaning:
8. First we fetched the Respective Month’s data files individually (data, pay and usage). Then we converted the respective individual data files into a data frame. Then we checked for NA values using the function defined earlier (mis\_na). There were no NA’s to be found for the data file (data) for all the five months. There were some NA’s in the data file (pay) for two months, which we replaced with median, and the rest three months didn’t have any NA’s. There were many NA’s found in the data file (usage) of all the months. We replaced them with median.
9. Then we moved on and detected outliers using median filtering. Using the function defined earlier (get\_median\_filtered), we assigned a variable (signal) and set the threshold in the range of 3. Using NumPy, we took the absolute difference between the original variable and the median of original variable. Then we run a loop till it transforms the outliers.
10. Merging of data:
11. After transforming the outliers, we merged all the three data files by ‘id’. The additional NA’s, which got created after merging, were replaced by 0.
12. Then we fixed the duplicate values (transformed).
13. Finally we created a new csv of the merged data of all the three cleaned individual data files (data, pay and usage).
14. Single View:
15. We followed the above procedure (steps 1-4) for all the five months and got five final csv’s.
16. Then we merged the final five csv’s on ‘id’ into one (new\_final) for a single view.

Processing using R and RHadoop on the server:

1. Libraries Imported:
2. dplyr - dplyr is a package which provides a set of tools for efficiently manipulating datasets in R. dplyr is a powerful R-package to transform and summarize tabular data with rows and columns.
3. data.table – It is used for fast aggregation of large data (e.g. 100GB in RAM), fast ordered joins, fast add/modify/delete of columns by group using no copies at all, list columns and a fast file reader (fread).
4. rmr2 – R can be connected with Hadoop through the rmr2 package. The core of this package is MapReduce() function that allows writing some custom MapReduce algorithms. It allows R developer to perform statistical analysis in R via Hadoop MapReduce functionality on a Hadoop cluster.
5. rhdfs – It provides basic connectivity to the Hadoop Distributed File System. R programmers can browse, read, write, and modify files stored in HDFS from within R.
6. lubridate – Makes it easier to work with dates and times.
7. pROC – It displays and analyzes ROC curves in R.
8. caret – It provides Miscelleneous functions for training and plotting classification and regression models. he package contains tools for data splitting, pre-processing, feature selection, model tuning using resampling and variable importance estimation.
9. Reading single view file and changing the date format:
10. We created a new variable sd1 and fetched the single view csv file (new\_final) using RHadoop.
11. Then we changed the date format of all the months and replaced the NA’s in the date column with the starting date of the month.
12. Then we changed the column names of September and October to avoid repetation of column names.
13. Deriving Variables:
14. We created a new variable (by\_id) and used the group by function (The group\_by function takes an existing tbl and converts it into a grouped tbl where operations are performed "by group". It groups a tbl by one or more variables) by taking the variable sd1 and id .
15. Then we created another variable (newdata) and assigned by\_id to it and used the mutate function to create new variables. Mutate adds new variables and preserves existing; transmute drops existing variables. We derived new variables using the sum of common column names of all the months. We derived 20 new variables.
16. We converted newdata into a data frame.
17. Separating the required variables:
18. We then separated the required variables from sd1 and newdata by creating an additional variable newf1.
19. Reading Disconnection data and changing the date format:
20. We created a new variable sd2 and fetched the Disconnection csv file (Discon1) using Rhadoop.
21. Then we changed the date format of the date column.
22. And changed the index name from ‘ID’ to ‘id’ to avoid problems while merging.
23. Merging the files:
24. We merged newfi and sd2 by ‘id’ and assigned it to a new variable (finalmerge).
25. After merging, we got additional NA’s in the date column which we replaced with the last date of the November month.
26. After replacing the NA’s, we created a new target churn variable (churn\_status) through which we can predict the churn. For that, we applied the condition to the date column that if the date is less than the last date of November then replace it by “1”, else “0”.
27. We replaced rest of the NA’s in the dataset by 0.
28. Writing and Reading the merged file:
29. We wrote a csv (finalmerge) of the merged file and Read it.
30. We then applied the logistic regression on the dataser but we got the error: datatype was not acceptable.
31. To resolve the error we changed the data type of date from ‘Character’ to ‘Date’ and Nulled the ‘Index’ and ‘id’ columns.
32. Logistic Regression:
33. Then after making the appropriate changes we applied logistic regression.
34. We subset the data into two classes: class1 and class0.
35. We split the data into two chunks: training and testing set. The training set will be used to fit our model which we will be testing over the testing set. We applied the glm function on target vriable (churn\_status) using train data.
36. By setting the parameter type='response', R will output probabilities in the form of P(y=1|X). Our decision boundary will be 0.5. If P(y=1|X) > 0.5 then y = 1 otherwise y=0.
37. Then we generated Confusion Matrix and ROC Curve Plotting.

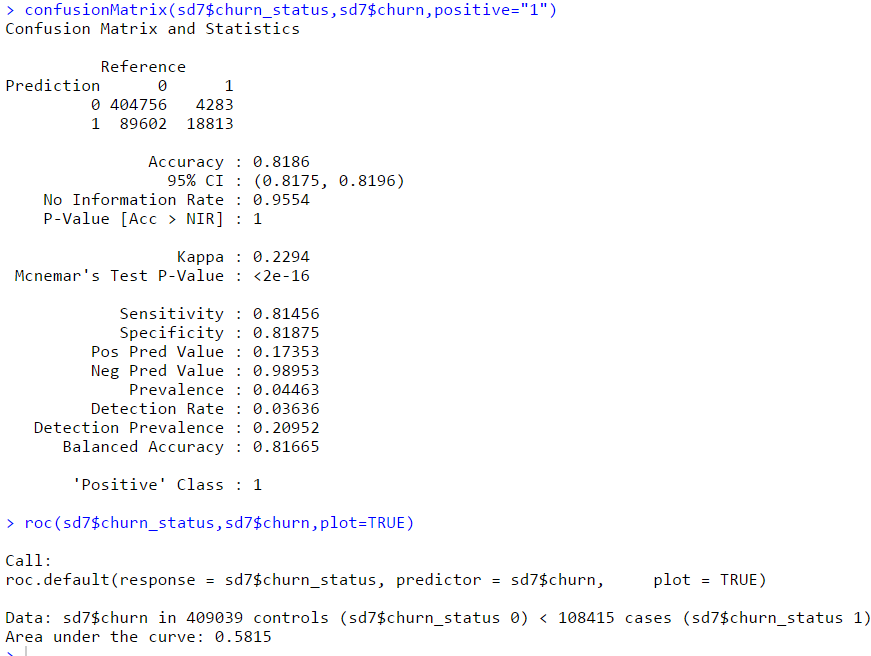
**Predictors:**

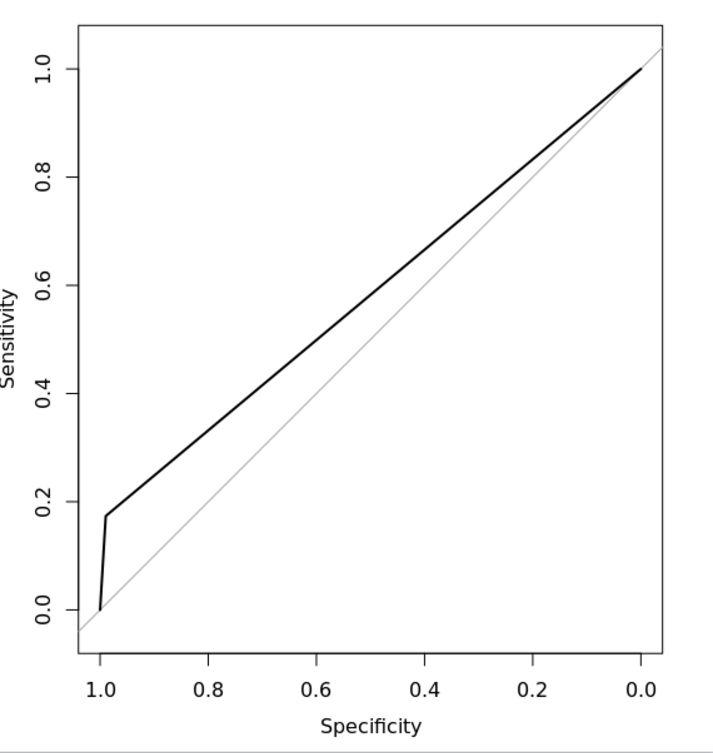
* VAR2
* VAR3
* vol2g
* vol3g
* mob\_actv1
* Total\_Bill
* RentalCharge
* NonRentalCharge
* Adj
* Usage
* LOC\_OG\_XYZ2XYZ\_MOU
* STD\_OG\_XYZ2XYZ\_MOU
* LOC\_OG\_XYZ2M\_MOU
* STD\_OG\_XYZ2M\_MOU
* STD\_OG\_MOU
* ISD\_OG\_MOU
* TOTAL\_OG\_MOU
* LOC\_IC\_XYZ2XYZ\_MOU
* STD\_IC\_XYZ2XYZ\_MOU
* LOC\_IC\_XYZ2M\_MOU
* STD\_IC\_XYZ2M\_MOU
* STD\_IC\_MOU
* ISD\_IC\_MOU
* ROAM\_OG\_MOU
* ROAM\_IC\_MOU
* TOTAL\_IC\_MOU
* Date

**Target Variable:**

Churn: if the customer has churned (1=yes; 0 = no)

**Result** -





**Conclusion** - Customer churn prediction plays a central role in churn management in mobile telephony

industry. In order to reduce the various costs associated with customer churn, it is imperative that mobile

service providers deploy churn predictive models that can reliably identify customers who are about to leave.

After the possible churners are identified, intervention strategies should be put in place with the aim of

retaining as many customers as possible.

Retention approach is based on combining customer churn prediction and [marketing action optimization](http://www.optimove.com/learning-center/marketing-action-optimization/).

Thus goes beyond “actionable [customer analytics](http://www.optimove.com/learning-center/deep-customer-analytics/)” to automatically determine exactly what marketing action

should be run for each at-risk customer to achieve the maximum degree of retention possible.

The analysis focused on churn prediction is based on logistic regression. The findings of this study indicate

that, in case of logistic regression model, the user should update the model to be able to produce predictions

with high accuracy.

Team:

* Sunaina Mallipedi

- Nikhil Dhapodkar