Project Report

Predict Customer Churn for Telecom Industry

Pruthvi Patel (NetId: prp4)

IS 590 MD SP2020: Methods for Data Science

Problem Statement

The biggest challenge for the telecom companies these days is the churn rate for the customers. Many consumers abandon their telecom companies because they receive better deals from other networks, or for several other reasons. A perfect example of this is the introduction of Reliance Jio in India, a telecommunications company that offered services at a very low price and succeeded in attracting consumers who used other providers that culminated in the merger of several service providers and knocked few out of business. Due to this reason, other providers started to reduce their service rates to stay in the competition. As the increase in customer churn rate directly affects companies' revenues, especially in the telecommunications industry, companies are seeking to develop ways of predicting potential customers who can churn. Therefore, it is really important to identify factors that increase customer churn and take the appropriate steps and reduce customer churn

Benefits of having a statistical learning solution

“Statistical learning is the ability for humans to extract statistical regularities from the world around them to learn about the environment”, - Wikipedia.

After referring to an article (https://towardsdatascience.com/machine-learning-powered-churn-analysis-for-modern-day-business-leaders-ad2177e1cb0d), we learned 2 important things:

1. An increase in 2% of customer retention is equivalent to a 10% reduction in costs.
2. As per the White House Office of Consumer Affairs, it is 6-7 times more expensive to acquire a new customer than to retain an old one.

For companies who still use conventional methods to determine whether or not a customer stays with the service, it would be useful for them to learn more about consumers faster than other rivals by learning from consumer actions and transforming that information into practice that would also be faster than their rivals. Not only would they be able to retain existing clients, but they would also increase the company's profit margin.

The key reason for using statistical learning is to understand the environment and behavior of the user and to implement the required solutions by predicting their behavior.

Research Question

From this project, we are trying to identify the factors that will help the company be aware of whether a specific customer is leaving/staying with them. By building a predictive model, we seek to predict whether a customer will remain with them or leave their company by considering their past data history.

Primary Research Question:

“Which are the most important factors that can affect customer’s decisions to switch to some other telecom network”?

Secondary Research Question:

“How can the retention team of the telecom industry find and target the high-risk customers with lucrative offers, who can possibly leave their network?”

As per our presentation, we had three research questions out of which we decided to remove the third research question which is “How to identify what type of promotions should telecom companies offer in order to retain customers?” from our report because we are anyways going to answer this question as a part of the primary and secondary research questions mentioned above.

The approach towards the project

We had divided our approach for this project into 7 parts:

Part 1: Connecting the data

We have hosted all our data files on GitHub so that the files can be easily accessed by our teammates and anyone can run our code on their side without having those files saved on their machines. As we have 3 different files in our dataset, we created 3 different data frames and combine them into one data frame to be used in our project.

Part 2: Exploratory Data Analysis

Exploratory Data Analysis also provides you with the context needed to create a suitable and efficient model. It gives you the visual representation of how the relationship between two variables. In this step, we have aggregated all the variables with our predictor variable i.e. “Churn” and have performed Exploratory Data Analysis by taking predictor variable on “X-axis” and other variables on “Y-axis”

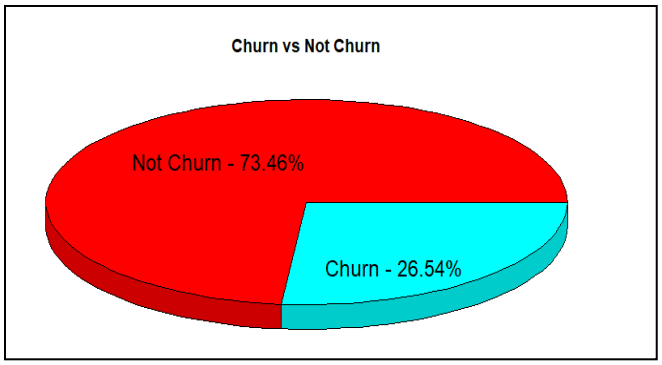

Figure 1: Not Churn vs Churn

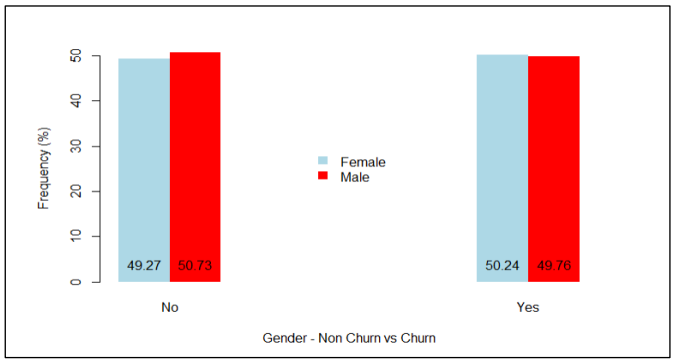

Figure 2: Gender - Non Churn vs Churn

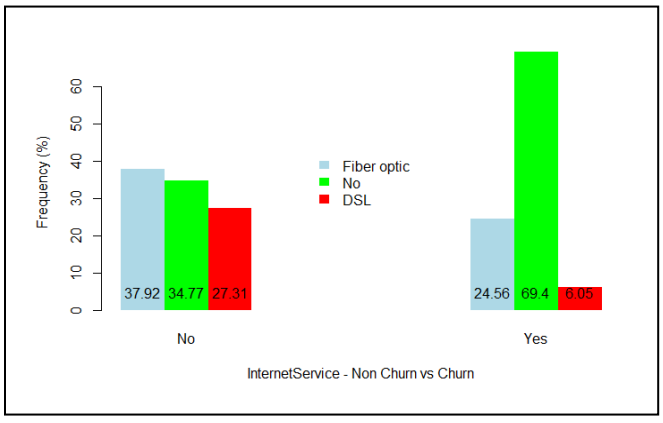

Figure 3: InternetService - Non Churn vs Churn

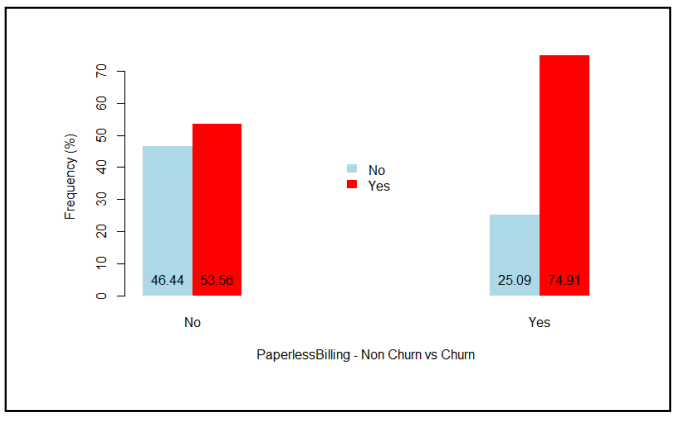

Figure 4: PaperlessBilling - Non Churn vs Churn

From figure 1, the Ratio of churn to Not Churn is seen which is approximately 27:73.

From figure 2, we can say that female member has slightly more churn rate, but it's negligible.

From figure 3, we can say that the major reason why a customer left the service was that they were not using the internet service provided.

From figure 4, we can say that, among the customer who left the provider, PaperlessBilling was the major reason as out of all customers who left, 75% of those were using Paperless billing.

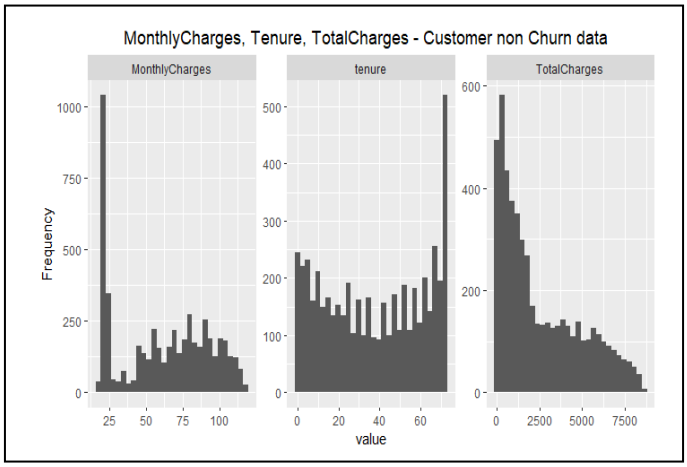

Figure 5: Distribution of customer non churn data for MonthlyCharges, Tenure and Total Charges

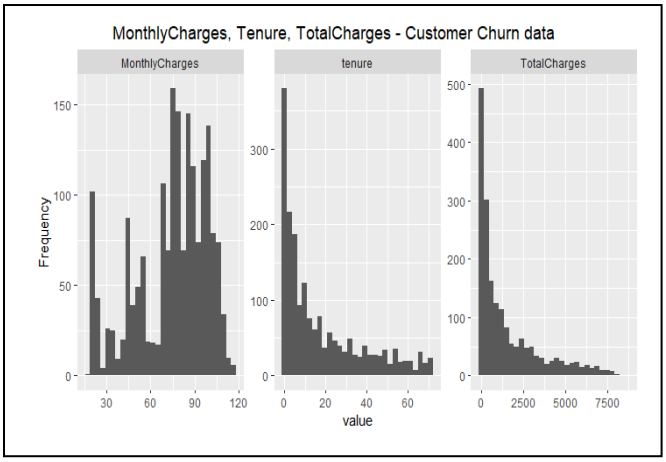

Figure 6: Distribution of customer churn data for MonthlyCharges, Tenure and Total Charges

From figure 5 and figure 6, one of the insights we can obtain is that people with higher monthly payments have left the services, while the proportion of customers who left the service was high when their tenure was smaller, but when it rose, fewer people left the service.

Part 3: Data Cleaning

After understanding what each column means and what is their importance, we moved towards cleaning our dataset. Once, we had the whole data, we performed data cleaning where we had deleted the "NA" or missing value if the column was categorical and had replaced the "NA" or missing value with the median, if the column was numerical.

Part 4: Model Building and Evaluation

We have used 5 algorithms to build the model. While building the model, we had taken all the variables in the training and testing set. We had split the training and testing set into 70:30 ratio i.e. 70% for the training set and 30% for the testing set.

Initially, we started with a simple model. Since logistic regression was easy and fast, we had built our first model using Logistic Regression. The other advantage of using Logistic Regression was that it also supported multi-classification.

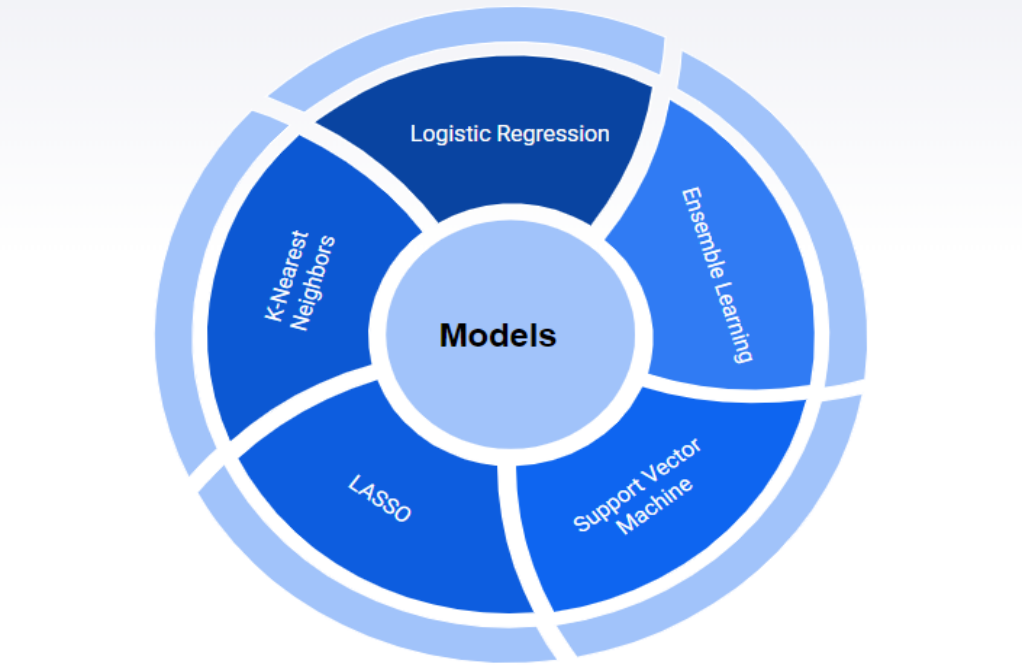

Figure 7: Different models implemented

Once we had built the initial model and by checking the p-value, we started understanding the association of variables with the predictor variables. There are very fewer chances that all variables would have an association with the predictor variable. For this stage, we made use of LASSO to build our model. The main reason for using LASSO to build the predictive model was that it eliminates irrelevant variables that are not associated with the target variable. This in turn also provided us good accuracy because, by removing the co-efficient, it reduced the variance without much increase in the bias.

Once, we had removed irrelevant variables, it was important for us to tune it. We had used KNN because it only required a few hyperparameters to tune. The other reason for using KNN was its advantage of handling non-linear models.

While building a model, it is always necessary to handle outliers efficiently. Since SVM with linear kernel derives maximum margin solution, it had handled the outliers in a better way compared to other models.

While building the model, there are always chances that the training set learns more patterns and becomes so flexible that it does not perform well on the testing set. This is basically, overfitting the model. We tried to overcome this problem by using the Random Forest approach. Since Random Forest allows us to select the number of trees and number of features at each node, it gave us a more generalized approach.

Lastly, we know that once we have built our model, there is always room for improvement. The process of iterative learning always proves to give the best result. We had using gradient boosting technique because by building one tree at a time, it corrects the errors made by the previously trained tree. By doing this, the model becomes more expressive.

(Reference: https://www.quora.com/What-are-the-advantages-disadvantages-of-using-Gradient-Boosting-over-Random-Forests)

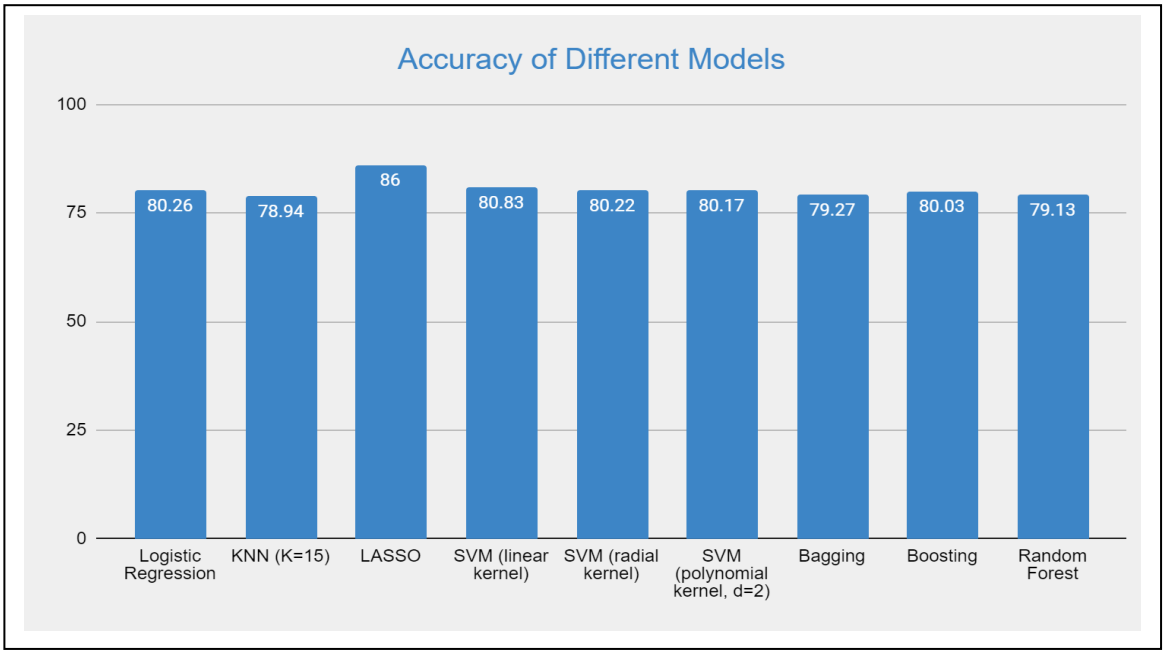

Figure 8: Evaluation of different Models by comparing their accuracy

Observation: Top 5 performing models are Logistic Regression, LASSO, SVM with linear kernel, Bagging, and Random Forest when all the variables are used.

Part 5: Model tuning by only selecting important variables

By comparing p-values from Logistic Regression, taking results from LASSO, by using the "varImp" function for Random Forest and the "summary" function for boosting, we took out 7 top variables which were important. In this stage, we have only selected the top 7 variables along with the predictor variables and had built our model. The top 7 variables are "tenure","Contract", "MonthlyCharges", "PaperlessBilling", "PaymentMethod", "TotalCharges" and "InternetService".

```
Coefficients:
                                        Estimate Std. Error z value Pr(>|z|)
(Intercept)                            1.139e+00  9.850e-01   1.156 0.247548
gender                                -3.293e-02  7.749e-02  -0.425 0.670842
SeniorCitizen                          1.545e-01  1.014e-01   1.524 0.127537
Partner                               -1.146e-02  9.238e-02  -0.124 0.901273
Dependents                            -1.283e-01  1.067e-01  -1.202 0.229252
tenure                                -6.496e-02  7.570e-03  -8.581  < 2e-16 ***
PhoneService                          -3.771e-02  7.847e-01  -0.048 0.961668
ContractOne.year                      -7.111e-01  1.283e-01  -5.542 3.00e-08 ***
ContractTwo.year                      -1.417e+00  2.136e-01  -6.633 3.28e-11 ***
PaperlessBilling                       3.369e-01  8.858e-02   3.803 0.000143 ***
PaymentMethodCredit.card.automatic    -9.038e-03  1.361e-01  -0.066 0.947043
PaymentMethodElectronic.check          3.506e-01  1.129e-01   3.106 0.001896 **
PaymentMethodMailed.check             -3.133e-02  1.377e-01  -0.228 0.819997
MonthlyCharges                        -3.465e-02  3.841e-02  -0.902 0.366995
TotalCharges                           3.832e-04  8.602e-05   4.454 8.42e-06 ***
MultipleLines                          4.545e-01  2.126e-01   2.138 0.032518 *
InternetServiceFiber.optic             1.509e+00  9.642e-01   1.565 0.117473
InternetServiceNo                     -1.733e+00  9.774e-01  -1.773 0.076228 .
OnlineSecurity                        -2.044e-01  2.148e-01  -0.952 0.341248
OnlineBackup                          -3.004e-02  2.117e-01  -0.142 0.887185
DeviceProtection                       2.017e-01  2.126e-01   0.949 0.342656
TechSupport                           -2.845e-01  2.181e-01  -1.304 0.192095
StreamingTV                            5.307e-01  3.939e-01   1.347 0.177884
StreamingMovies                        5.645e-01  3.938e-01   1.433 0.151735
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 9: p-values from Logistic Regression model

```
25 x 1 sparse Matrix of class "dgCMatrix"
                                              1
(Intercept)                        3.661761e-01
(Intercept)                        .
gender                             .
SeniorCitizen                      2.529878e-02
Partner                            .
Dependents                        -1.075298e-02
tenure                            -3.319195e-03
PhoneService                      -7.780434e-03
ContractOne.year                  -8.941174e-02
ContractTwo.year                  -5.450470e-02
PaperlessBilling                   3.867400e-02
PaymentMethodCredit.card.automatic .
PaymentMethodElectronic.check      8.392859e-02
PaymentMethodMailed.check          .
MonthlyCharges                     .
TotalCharges                      -1.790232e-05
MultipleLines                      2.287997e-02
InternetServiceFiber.optic         1.461571e-01
InternetServiceNo                 -1.323504e-01
OnlineSecurity                    -4.293596e-02
OnlineBackup                      -7.381070e-03
DeviceProtection                   .
TechSupport                       -5.078046e-02
StreamingTV                        2.191053e-02
StreamingMovies                    3.075505e-02
```

Figure 10: Non-zero coefficient estimates from LASSO model

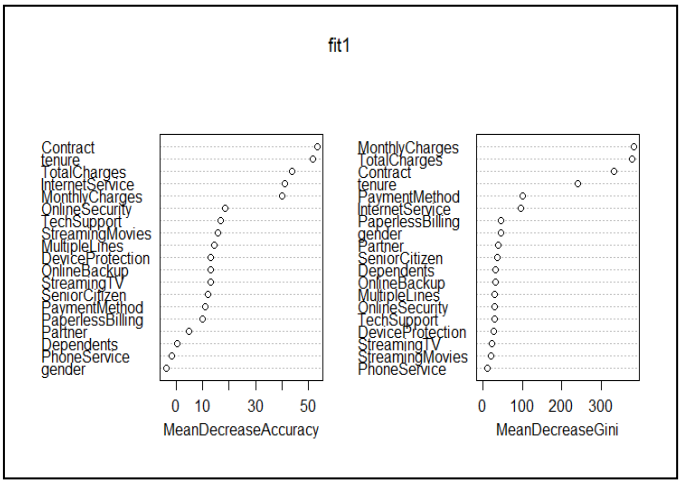

Figure 11: Plot of important variables from Random Forest

	var	rel.inf
	<chr>	<dbl>
TotalCharges	TotalCharges	24.1089048
Contract	Contract	20.1731051
MonthlyCharges	MonthlyCharges	17.8380104
tenure	tenure	12.6216733
InternetService	InternetService	6.3735776
PaymentMethod	PaymentMethod	5.6762763
PaperlessBilling	PaperlessBilling	1.9358967
SeniorCitizen	SeniorCitizen	1.5968673
gender	gender	1.5414920
OnlineSecurity	OnlineSecurity	1.4689254

1-10 of 19 rows

Figure 12: Important variables from Boosting model

Below is a bar chart graph comparing the accuracy of all the model which we had implemented by using all the variables and by using the 7 important variables which we had identified.

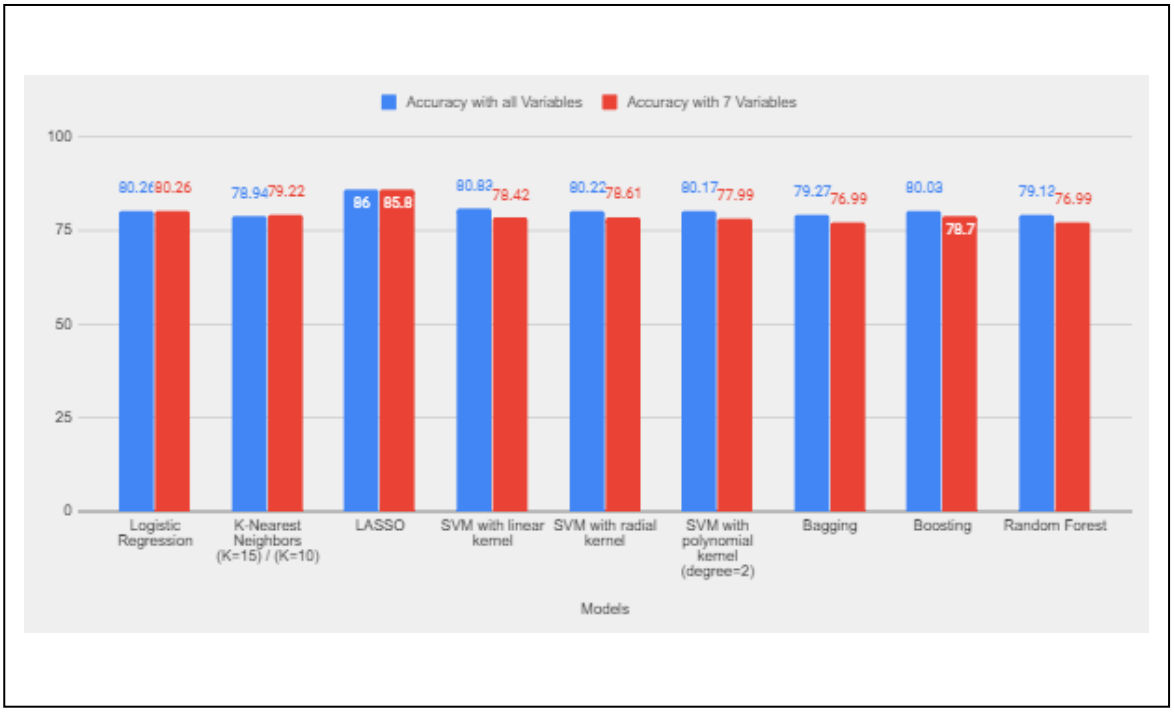

Figure 13: Model evaluation with all variables vs 7 variables

Observation: As we can see, after taking only 7 variables, there is not much change among the top-performing models i.e. Logistic Regression, LASSO, SVM with linear kernel, Boosting, and Random Forest.

Part 6: Applying Sampling technique

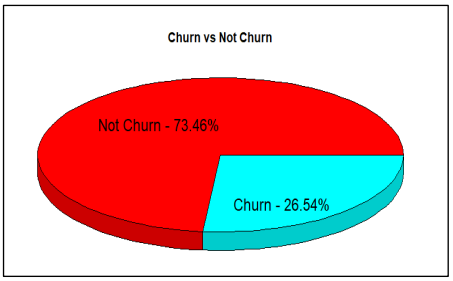

Figure 14: Customer - Not Churn vs Churn proportion of the dataset

As we can see from figure 14, the ratio of non-churn to churn is approximately, 73:27. This means that the data involves imbalanced classification. To overcome this problem, we are oversampling the minority class i.e. we are using Synthetic Minority Oversampling Technique (SMOTE). With the help of sampling, we created a new data frame that had an equal number of minority and majority class.

```r
```{r}
library(DMwR)

## Smote : Synthetic Minority Oversampling Technique To Handle Class Imbalancy In Binary Classification
newdata <- SMOTE(Churn ~., cleanData, perc.over = 100)

as.data.frame(table(newdata$Churn))
```
```

Var1 <fctr>	Freq <int>
0	3738
1	3738

2 rows

Advantage of using SMOTE:

"SMOTE first selects a minority class instance at random and finds its k nearest minority class neighbors. The synthetic instance is then created by choosing one of the k nearest neighbors b at random and connecting a and b to form a line segment in the feature space. The synthetic instances are generated as a convex combination of the two chosen instances a and b."

(Reference: https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/)

Evaluating the Model Performance (after using SMOTE) using Area under Curve:

When the classification is imbalanced, accuracy is always not a good measure. In our project, we had evaluated the model for random forest and boosting using Area under Curve.

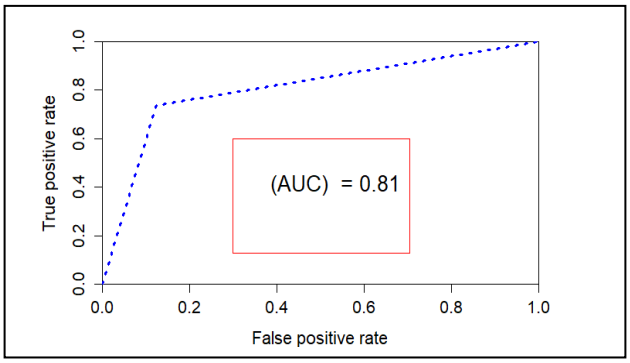

Figure 15: Area under Curve for Random Forest

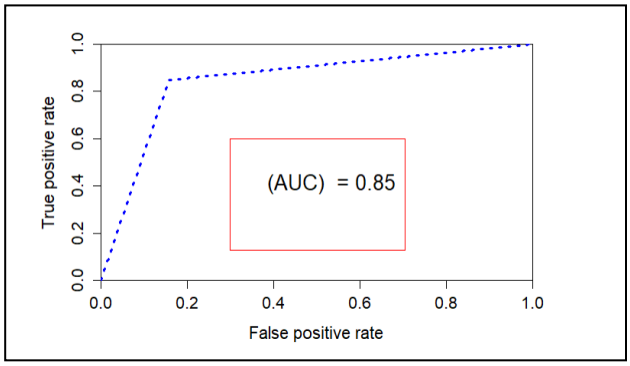

Figure 16: Area under Curve for Boosting

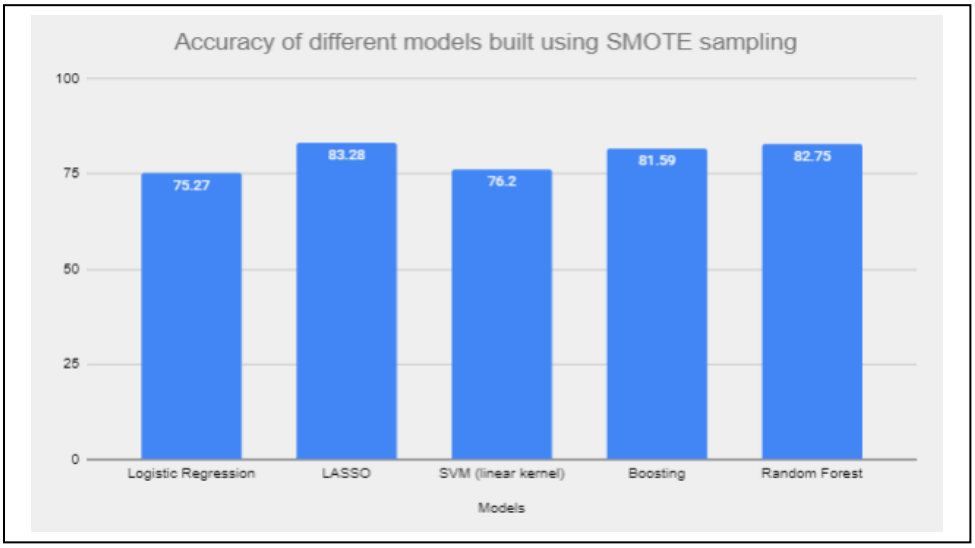

Figure 17: Accuracy of different models implemented using SMOTE Sampling

As seen in figure 17, we had implemented the top 5 performing models from part 4 where we used all the variables. By implementing SMOTE sampling, we can compare figure 8 and figure 17 to see that the accuracy of all the models dropped by approximately 3% to 5% except for Boosting and Random Forest models as we can see a spike of about 2% to 3% in their accuracy.

Part 7: Applying one-hot encoding on the selected important categorical variables

Many machine learning algorithms cannot directly work with categorical data. Initially, for those algorithms who could not directly work with categorical data, we had converted all the categorical variables into a numeric form which is also called Integer Encoding. Though as integer encoding is not recommended when the categorical variables have no ordinal relationship because it allows the model to assume a natural order between categories and can lead to poor performance or unexpected results.

So, in such a case, one-hot encoding can be applied for the integer representation. One hot encoding represents the categorical variables as binary vectors. This is where the encoded integer variable is deleted, and a new binary variable for each unique integer value is added.

(Reference - https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/)

As we had 3 categorical variables in the important seven variables selected, which were “contract”, “PaymentMethod” and “InternetService”, we transformed all these variables by using one-hot encoding into dummy variables and again built different models to check their performances. Below is a chart comparing the accuracy of various high-performance models by using all variables, using the seven important variables, and using the seven important variables along with dummy variables.

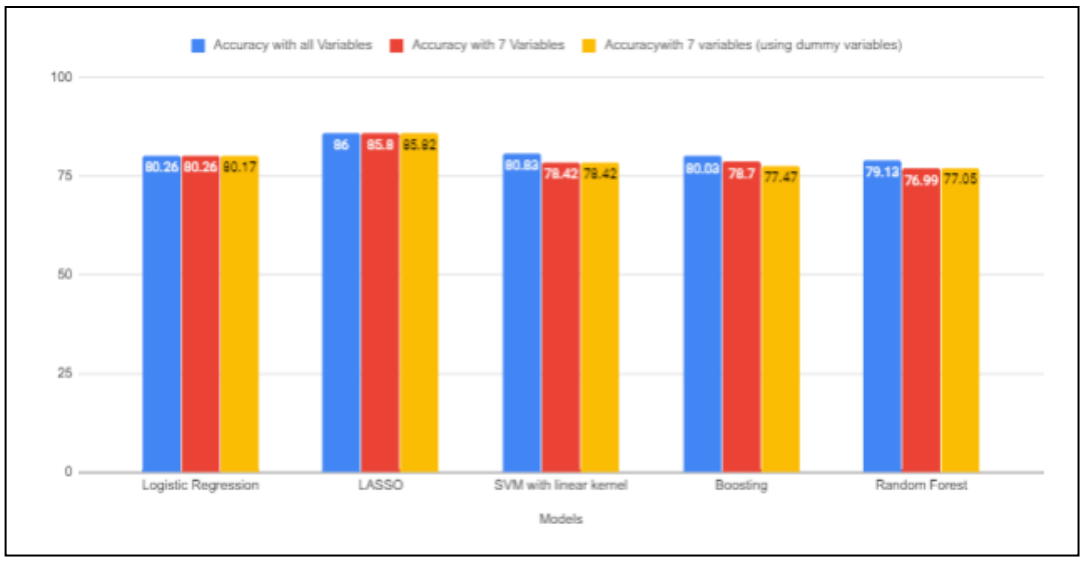

Figure 18: Comparing accuracy of different models using different variables

As seen in figure 18, after applying one-hot encoding the accuracy of some models has increased by a very small margin. We can see that accuracy of models like LASSO and Random Forest has increased a bit as compared to when only seven important variables were used without implementing one-hot encoding. Whereas the accuracy of models like Logistic Regression and Boosting has decreased a bit. The accuracy of model SVM with linear kernel has remained the same.

Out of all the models which we had implemented after trying all the variables, important seven variables and important seven variables with dummy variables, we can see from figure 18 that LASSO was the highest performing model which has an accuracy of around 86% followed by Logistic Regression Model.

Recommendations for Research Question

To answer the primary research question which is “Which are the most important factors that can affect customer’s decisions to switch to some other telecom network?”, we have identified seven important variables that can affect a customer’s decision in leaving a network are “tenure”,”Contract”, ”MonthlyCharges”, “PaperlessBilling”, “PaymentMethod”, “TotalCharges” and “InternetService”.

To answer the secondary research question which is “How can the retention team of the telecom industry find and target the high-risk customers with lucrative offers, who can possibly leave their network?”, we have considered 3 variables “totalcharges”, “paperlessbilling” and “contract” (which were also considered among the top seven important variables) to form our recommendation. We had taken a median of total charges which was 1397 and to round it up, we took the total charges variable value as 1400 for comparison.

The below figures displays the number of customers who do not churn and who churn with total charges greater than 1400 having different contract types and has not opted for paperless billing.

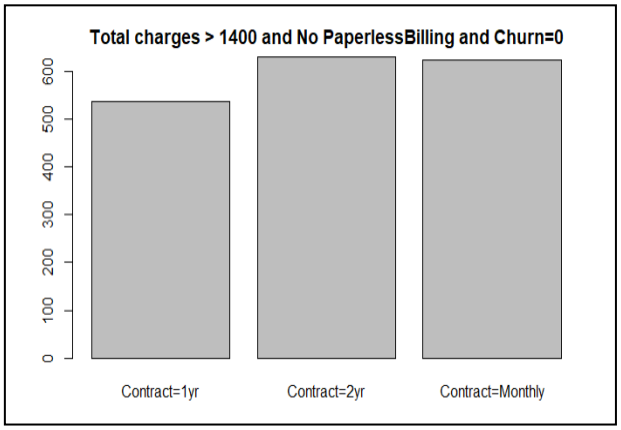

Figure 19: Customers with different contracts who don't churn when total charges are above 1400 and doesn't use PaperlessBilling

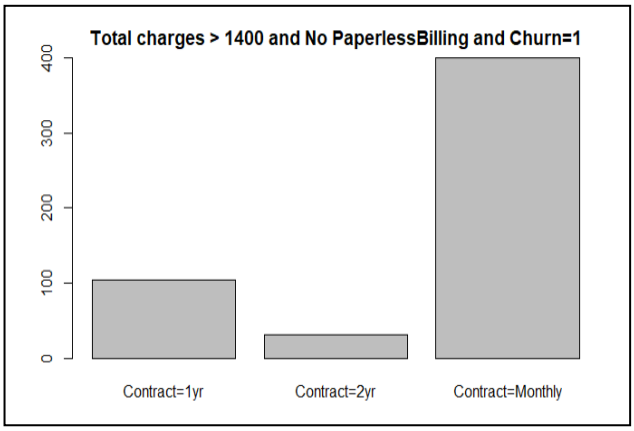

Figure 20: Customers with different contracts who churns when total charges are above 1400 and doesn't use PaperlessBilling

For making a recommendation, we had calculated the ratio of figure 19 to figure 20 across individual entities as shown below in the table.

Contract type	Customers not churning with No Paperless Billing (A)	Customers Churning with No Paperless Billing (B)	The ratio of B/A+B (Probability of customer churning)
1yr	538	104	16%
2yr	631	31	5%
Month-to-Month	625	400	39%

- From the above table, we can say that the customers who are not using Paperless Billing in blend with monthly services are more likely to churn i.e. Customers in a month-to-month contract, with no Paperless Billing and are within 12 months of tenure, are more likely to churn.
- We would recommend to the retention team that they should target customers whose tenure is less than 12 months, who do not use paperless billing, and are on a month-to-month contract because they are more likely to churn.
- Retention team should provide offers to such customers to convert their contract type from monthly to yearly to make them use their services for over 12 months and suggest customers to use the Paperless Billing method which can decrease the probability of them churning as seen in the table above.

The company can also take an initiative to make the billing online, which in turn would help to retain the customer and at the same time help reduce paper waste and contribute towards the environmental well being.

Data Source

We have used the telecom churn dataset from Kaggle. This dataset includes historical data of a telecom company that includes 21 different parameters and over 7000 rows. Each row represents a customer, each column contains customer's attributes described on the column Metadata.

Data source link - https://www.kaggle.com/vpfahad/telecom-churn-data-sets

Are there other previously published solutions to this problem? If so, how does our solution differ or compare?

There are solutions available on Kaggle, but we have not referred to any to keep ourselves away from any biased ideas.

Things we tried but did not work out

We tried to utilize the approach of the forward and backward selection of variables to understand which variable has the top-priority but we were facing an error, due to which we could not complete it. Also, the course only covered its usage for linear regression models whereas our case is the classification (logistic regression) which uses a different package/method than the one we covered/used in the course.

Expectations going into the project that was proved correct or incorrect

We had one plan, where we felt "total annual charges" and "monthly charges" would play a very important role directly. But, when we built the model using just "total annual charges" and "monthly charges," we discovered that the model's accuracy was less compared to the models where we had incorporated multiple variables.

We realized from this that only the amount payable by the customer is not relevant. If there are additional services used with any contract in conjunction along with these two variables will provide more accurate insights.

For example, an individual leaving the company whose monthly charges were $200 would not offer in-depth insights. On the other hand, say, A person who had a 2-year contract and whose monthly payments were $100 left the company, we can examine that while his monthly fee was lower, he still left the company. This will give us the idea that we would evaluate other services that are used by the customer to get better insights.

How did our team worked together and how was the work distributed?

We had worked together on different parts of the project and coordinated our meetings through an online zoom meeting. There were doubts about how to provide attribution in EDA, where we came up with a solution as a team by discussing various approaches. For working on models, we had split our work and once it was completed, we tagged up again to discuss the recommendation and worked on the report together.

Things we would like to do for this project in the future if we had more time

We would like to examine every single variable and how it relates to customer decision making. We were only able to evaluate a few variables for this project in-depth to recommend the retention teams given in the report above in recommendation for the research question section. Given more time, we would want to combine more variables, create a new function and evaluate whether it plays a significant role and suggest more ways to retention team to retain the customers based on the variables identified.

There can be chances, for example, that customer is using two phones. One has internet access, another has not. There's a good probability the other phone will be used only for business calls. During this time, consumers will prefer deals without internet access and with low monthly fees. Now in this situation, we have to find an explanation as to why did the customer leave the service by aggregating different variables.

We would also like to integrate data from multiple sources to solve such a problem.