**“Customer churn prediction model development with the help of Machine learning”**

**Introduction-** The phenomena where the customer leaves the organization is referred to as customer churn in financial terms. Identifying which customers are likely to leave the organization, in advance can help companies take measures to reduce customer churn.

In this case, we explain how machine learning algorithms can be used to predict churn for organization. The article shows that with help of sufficient data containing customer attributes, machine learning models can be developed that are able to predict which customers are most likely to leave the organization in future, with high accuracy.

Machine Learning is the art of Predictive Analytics where a system is trained on a set of data to learn patterns from it and then tested to make predictions on a new set of data. The more accurate the predictions are, the better the model performs.

In this project we will be building a model that Predicts customer churn with Machine Learning. We will do this by implementing a predictive model with the help of python. Prediction of Customer Churn means how many customers will stay or leave the usage of products or services of given company.

For any business in which everything depends on the behavior of customers, retaining them is the number one priority for the organization. Customer churn is the process in which the customers stop using the products or services of a business.

Customer Churn or Customer Attrition is a better business strategy than acquiring the services of a new customer. Retaining the present customers is cost-effective, and a bit of effort could regain the trust that the customers might have lost on the services.

On the other hand, to get the service of the new customer, a business needs to spend a lot of time, and money on to the sales, and marketing department, more lucrative offers, and most importantly earning their trust. It would take more recourses to earn the trust of a new customer than to retain the existing one.

This will be a great help for organizations to take necessary and corrective steps to retain them for robust health of businesses. As time, energy and sources are required more in acquiring new customers then retaining the current customers, as it is said that “Permanent customer is the heart of any business”.

**Problem Definition**- on analyzing the given data set, it is evident that it is a binary classification model as churn of customers are predicted as a result of either YES or NO, so we will use the classification model to find out the result. We can understand the whole process as-

The following method will take us through the way we have written to predict the customer churn.

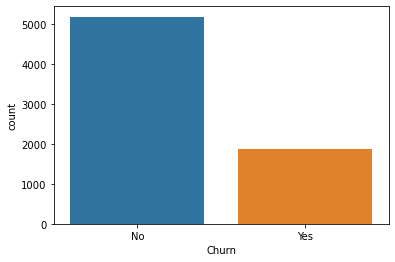
* First, we have imported all the necessary libraries we would need to proceed further in our code
* Just to get an idea of how our data looks likes, we have read the CSV file and printed out the first five rows of our data in the form of a data frame
* Once, the data is read, some pre-processing needed to be done to check for null, outliers, and so on
* Once the pre-processing is done, the next step is to get the relevant features to use in our model for the prediction. For that, we have done some data analysis to find out the relevancy of each predictor variables.
* After the data has been plotted, I have dropped customer id, gender, phone service and contract column it is observed that Gender doesn’t have much influence on churn, Based on our observation, we have taken the features which have more influence on churn prediction
* The data is scaled, and split it into train and test set
* We have fitted the all classification models to our new scaled data

**Data Analysis**- The objective is to find out the customer churn based on using classification models. The given dataset contains 7043 rows and 21 columns. It has object and integer data types and we can see there are no missing values in dataset. I have divided the dataset in to categorical and numerical features classifications to have more clarity. We have 18 columns for categorical values and 3 columns for numerical values.

I need to do data cleaning and feature corrections-

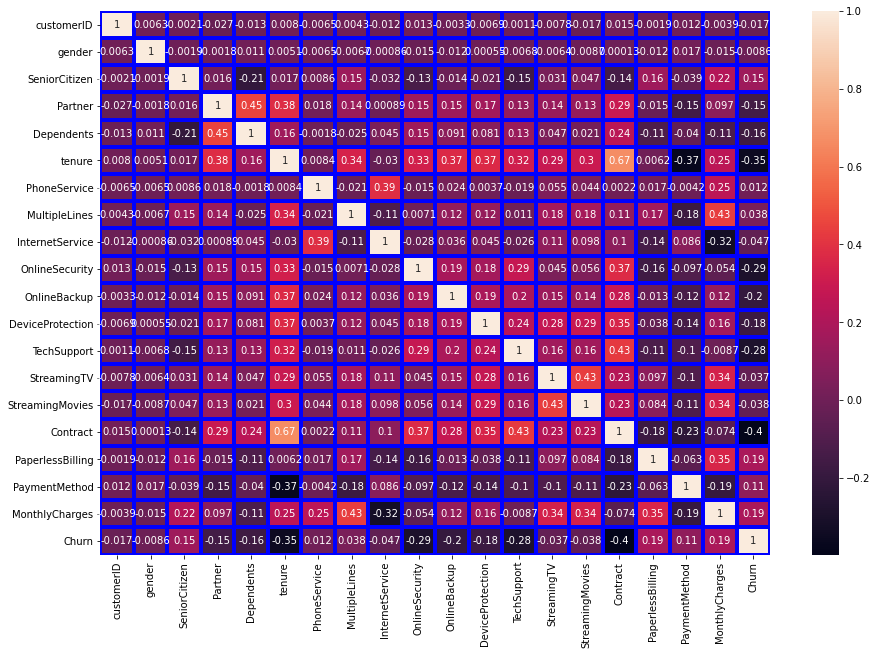
I imported Regular expression (with command of import re) to clean and formatting ‘Contract’, ’Payment method’, ’Internet service’, ’Multiple lines and ‘Customer ID’.

Target dataset is imbalanced hence either oversampling or under sampling has to be done to correct the same.



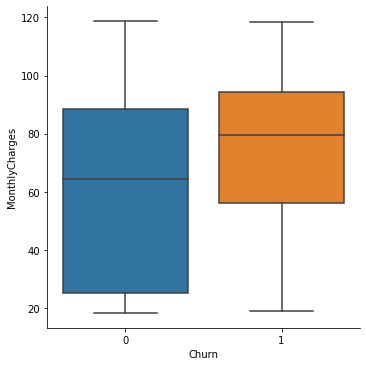
I need to convert categorical features to numerical ones for building machine learning models so using Label Encoder from sklearn to convert categorical features into numerical data.

To establish correlation among attributes, I have applied ‘correlation’ matrix and plotted graph. Target data(Churn) is highly correlated with Monthly charges, Paperless billing, Senior Citizen and payment methods than other features.

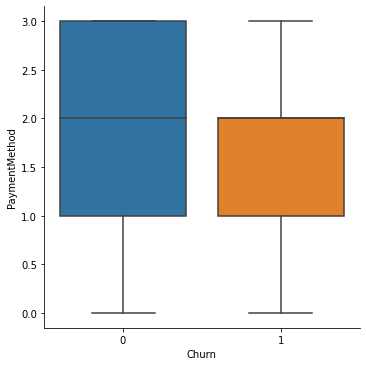


After analyzing we can drop customer id, gender, phone service and contract as these won't affect results

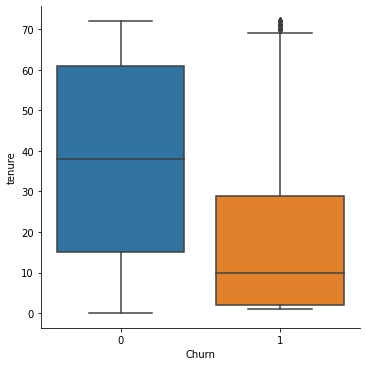
After plotting factor plot between monthly charges and churn it is clear that monthly charges are more so the churn is high.



After plotting factor plot between payment method and churn it is clear that payment method is of with more options then the churn is low.



After plotting factor plot between tenure and churn it is clear that the more the tenure of customer with services is less the churn.



I used boxplot (Subplot) method to have a complete glance and found that there are not much outliers presents in the dataset and with distribution plots got distribution of data of all attributes at one glance then plotting individual graphs, applied this for boxplot and distribution plot to capture all images at one shot.

**EDA Concluding Remarks-**

After importing dataset to jupyter environment, I did data cleaning, formatting, and feature engineering for the dataset. The data set comprises of 7043 rows and 21 columns, and is a mix of object and integer type. There are 18 categorical and 3 numerical columns. I have dropped customer id, gender, phone service and contract column as these are of no relevance and won’t affect much on prediction.

I also found that the target data set is imbalanced, and oversampling is required. There are not many outliers present in dataset, so outlier removal is not required and skewness also in permissible range. Analysis of the factor plot between churn and monthly charges, payment method and tenure shows churn is high with more monthly charges, and less with more payment options and more tenure respectively.

As per correlation analysis churn is highly associated with monthly charges, paperless billing, senior citizen and payment method chronologically.

**Preprocessing Pipeline-**

1 Applied skew test to dataset to find out the skewness.

2 No NaN and Infinity values present in dataset so correction is not required.

3 Data Standardization- from sklearn.preprocessing imported standard scaler for standardization of dataset.

4 Oversampling- since the target data set is very imbalanced so I had to do either over sampling or under sampling, but I opted for over sampling so from imblearn.oversampling imported SMOTE to normalize the target data set.

**Building Machine Learning Models-**

Since the problem statement is binary classification in nature, so imported Logistic regression model from sklrean.linear,Random Forest classifier from sklearn.ensemble,Decision Tree classifier from sklearn.tree and support vector form sklearn.svm as following-

from sklearn.linear model import Logistic Regression

from sklearn.tree import DecisionTreeClassifier

from sklearn.ensemble import RandomForestClassifier

from sklearn.svm import SVC

Then imported accuracy score from sklearn metrics to get accuracy score of the model, which is as follows-

from sklearn.metrics import accuracy score

imported train, test, split from sklearn metrics to classify the dataset into train data & test data so that we can upload the engineered data set for prediction, which is as follows-

from sklearn.metrics import confusion matrix,classification\_report

from sklearn.model\_selection import train\_test\_split

Now when we classified data in to training and testing data, and run logistic regression models, Decision tree classification, Random forest classification and support vector machine then we get accuracy score of each model for given dataset, as in this case, we got following accuracy scores-

Accuracy score of Decision Tree classifier- 77%

Accuracy score of Random Forest classifier-85%

Accuracy score of Support vector-79%

From above it is clear that random forest model is giving 85% prediction but at this stage it is not right to conclude because higher efficiency of model can be because of over fitting.

So, to check this we will import cross validation score, as follows-

from sklearn.model\_selection import cross\_val\_score

and we will get Cross validation score for each model, and we get following result-

CV score for Decision Tree classifier-77%

CV score for Random Forest classifier-84%

CV score for Support vector-79%

And then I calculated the difference of each model’s accuracy score and cross validation score and got two models for final consideration which are Random Forest classifier and Support vector.

Now to improve further efficiency for these models need to do hyperparameter tuning.

So, imported GridsearchCV from sklearn as follows-

from sklearn.model\_selection import GridSearchCV

now to do the hyper parameter tuning for support vector-

selected parameters for the model and run to get best parameters, and then applied those shortlisted parameters to get result for the model, which is as follows-

* parameters={'C':[0.1,1,10,100,500],

'kernel':['poly','rbf','sigmoid']

}

* for the model the best parameters are-

**{'C': 100, 'kernel': 'rbf'}**

and with above Support Vector model is at 82% of accuracy.

Similarly did hyper parameter tuning for Random Forest model-

selected parameters for the model and run to get best parameters, and then applied those shortlisted parameters to get result for the model, which is as follows-

* parameters ={'n\_estimators':[100],

'max\_features':['auto','sqrt'],

'max\_depth':[4,5,6,7,8],

'criterion':['gini','entropy']}

* for the model the best parameters are-

**{'criterion': 'gini',**

**'max\_depth': 8,**

**'max\_features': 'auto',**

**'n\_estimators': 100}**

And on above shortlisted parameters Random Forest model is working at 81% efficacy.

Since the Support Vector model is performing at higher accuracy then random forest and decision tree so we go with support vector model for prediction.

Saving The Model-

I have imported the ‘JobLib’ to save the model.

**Conclusion-**

The target data is binary classified, so after getting the dataset, and performed data cleaning, formatting, feature correction and exploratory data analysis. The dataset checked for skewness, standardized the dataset with standard scaler and with the help of SMOTE did over sampling of target column, and now dataset is ready for machine learning model building.

Since problem statement is of the nature of binary classification, I applied the classification model, checked the accuracy scores and to avoid overfitting/underfitting, I also analyzed the cross-validation scores for each model. And then did hyperparameter tuning for selected models.

And finally came to the conclusion that in this case “Support Vector Classifier” is the more accurate model.

And with the help of machine learning models it is easy for the companies to calculate the customer’s moves based on the given characteristics and to have prediction about their longevity with the company and they can -

1 Reduce churning of customers

2 Save lot of time, money and energy

3 Achieve long term association and self-sustainable business