ABSTRACT

This project is a part of Bussiness Analytics class Course aimed toward understanding customer churn behavior in telecommunication sector.

In the upcoming section we will analyze different attributes of customer that helps us to estimate likelihood that customer will churn and select the best subset of that. We will try different approach for predicting customer churn and select the best one.

We have used Python language throughout this project and the libraries that we have used includes:

1. Pandas (for EDA)
2. Matplotlib, Seaborn, Plotly (for better visualization)
3. Numpy (for mathematical modeling)
4. Scikit-learn (for different models)
5. Imblearn (for Oversampling)

At the end of this project we will be able to achieve standard accuracy over predicting customer churn on test data.

TABLE OF CONTENT

Chapter 1

Introduction

1.1 What is Customer Churn:

Customer churn is a critical metrics for a growing business to evaluate. It is defined as the percentage of customers who stopped using a company's product or service within specific time frame. While it's a sad measure, it's a metric that can give a company the harsh truth about its customer acquisition. It's more likely to fail if a company doesn't measure the inevitable failures. Each company strives for 100% of customers retention but that is impossible.

This measurement is important in those businesses which are subscriber-based, in which most of the revenues are generated via subscription fees.

1.2 Types of Customer Churn:

There are two types of churn:

1.**Voluntary churn**: It occurs when customers left a particular company in order to shift to another company.

2.**Involuntary churn**: It occurs when a customer stop doing business with company due to death, or relocation to a distant location.

1.3 Role of Customer Churn in Bussiness:

It is necessary for a company to have its growth rate greater than its churn rate in order to expand its client. The churn and growth rates are totally opposite factors as one measures the loss while other measures the rate of acquisition of customers.

A high churn rate could poorly affect profits and hamper the growth of the company whose revenues are heavily dependent on subscription.

e.g., Netflix, Telecommunication Industry, Amazon Prime, Antivirus.

Because of multiple options available to a customer, a company can measure it’s standing among other competitors. For Example- If one out of every 10 internet subscribers terminate their subscriptions from that particular internet provider within one year, the yearly churn rate for that internet provider would be 10%. So It is necessary for a company to understand the segments of customers who were expected to leave the company so that the company could engage them with specials offers and try their best to avoid loosing such customer because price of getting a new customer is usually higher than accomodating the old one.

Another field where the churn rate helps in determining the scenario is employment. Turnover of an Employee can also be measured with the help of churn rate, as it is a method for analyzing the company's hiring and acquisition patterns.

1.4 Problem Statement:

Building a machine learning model that identifies the segment of customers with the intention to leave Telecom company .

1.5 Objective of the Project:

The main objective of this project is to predict the segment of customer who are likely to leave the company. And by predictive analysis of this measure, company can evaluates themself and improve their service inorder to gain better customer satisfaction or simply could engage them with some special offers instead of loosing them inorder to avoid any bussiness losses.

Chapter 2

REVIEW OF LITERATURE:

* Customer churn classification was first introduced by Mozer in 2000 who used neural network architecture for this purpose.

* Then Burez and Van den Poel made a prediction model for European Tv in 2006. They used Random forest model for their work.

* Kim and Yoon published a realted model in 2004 using binomial logistic regression.

* Recently in 2012 Lee developed a KNN based classification model for this project and achieve better performance than the traditional architecture.

There are many well known algorithm for classifying churn prediction which includes decision-tree, logistic-regression, neural-network and genetic-algorithm.

Chapter 3

Methodology:

Language used: For entire project work, we have used python language.

Steps involve:

1.Data Collection: We have used Telecom Customer Churn dataset for our project. It consists of 21 features of 7043 unique customers.

2. Data Analysis: For data analysis and data visualisation we have used pandas, matplotlib, seaborn, plotly library. We did univariate, bivariate and multi variate analysis of features in this section.

3. Data Preprocessing and Data Cleaning:- In data preprocessing step, missing values, outliers, and inconsistencies are removed. Categorical features are converted into numeric ones. Then data is splitted into train set (70%) and test set(30%).

4. Best Model Selection:- We train different models starting with a simple model to complex one(Logistic-regression, Decision-Tree, Random-forest,Light-gbm) on the train dataset set then pick the best model which performs well on our test dataset.

5. Further Improvement:- We select the best model and tune its hyperparameter for further improvement of model's accuracy. We played with the learning rate,maximum leaves, maximum depth etc(which are not helpful in our case). We did Feature selection which leds to increase in accuracy.

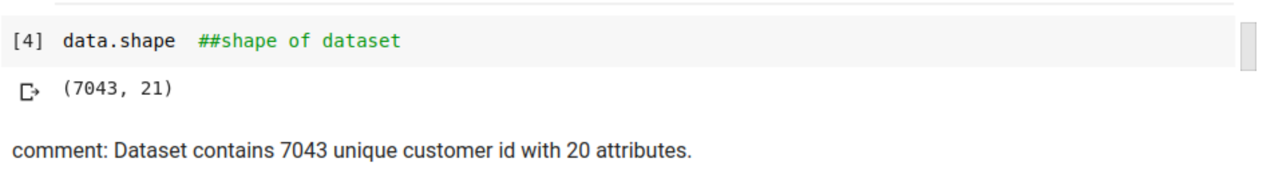
6. Conclusion:- After performing lots of experiment we summarize whole things in this section.

Chapter 4

Dataset Description:

3.1 Shape and Datatypes of each feature column:

The dataset we have used in this project is taken from kaggle whose links is given in appendix section. It is the data from a telecommunication sector which contains 7043 customers with unique id and 21 attributes corresponding to each customers.



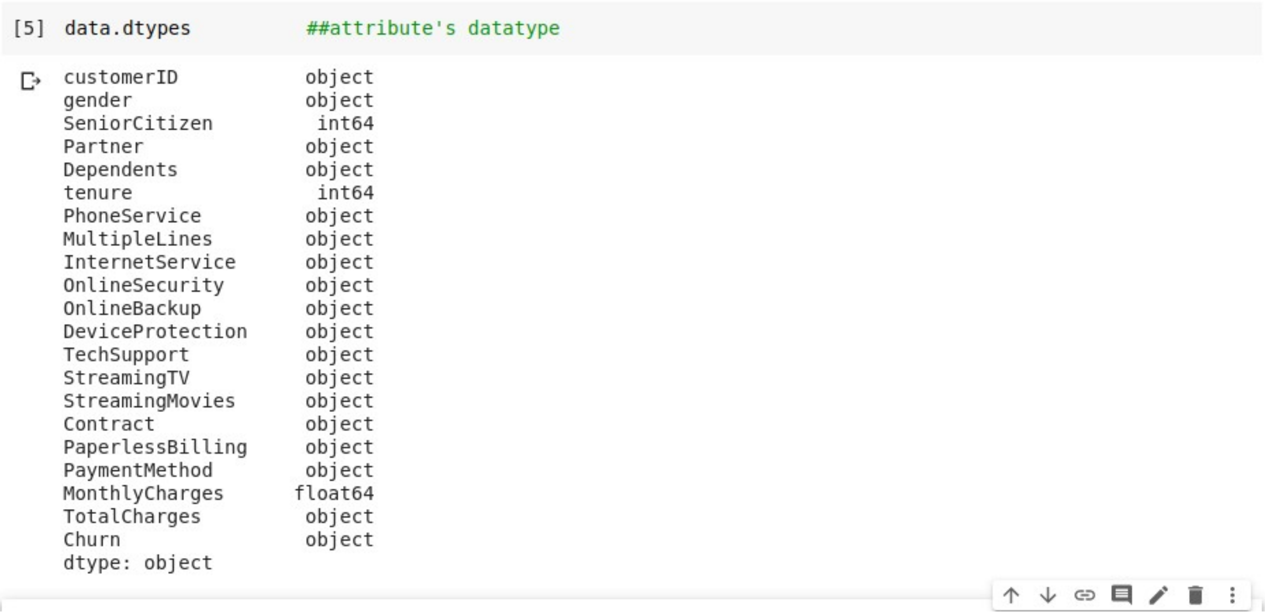
```
[4]  data.shape   ##shape of dataset

     (7043, 21)
```

comment: Dataset contains 7043 unique customer id with 20 attributes.

Source of image: Author.

Datatypes of each columns is shown below:



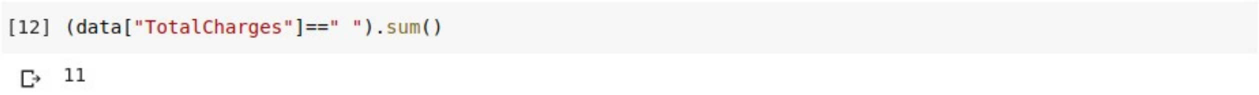
```
[5]  data.dtypes          ##attribute's datatype

     customerID            object
     gender                object
     SeniorCitizen          int64
     Partner               object
     Dependents            object
     tenure                 int64
     PhoneService          object
     MultipleLines         object
     InternetService       object
     OnlineSecurity        object
     OnlineBackup          object
     DeviceProtection      object
     TechSupport           object
     StreamingTV           object
     StreamingMovies       object
     Contract              object
     PaperlessBilling      object
     PaymentMethod         object
     MonthlyCharges       float64
     TotalCharges          object
     Churn                 object
     dtype: object
```

Source of image: Author

Note :- It is interesting to see that the datatype of Total Charges column is of object type but in actual it should be of float data type.

The reason behind this is, it contains some blank value which is not a typical nan type but simply a string containing space.



```
[12] (data["TotalCharges"]==" ").sum()
```

```
11
```

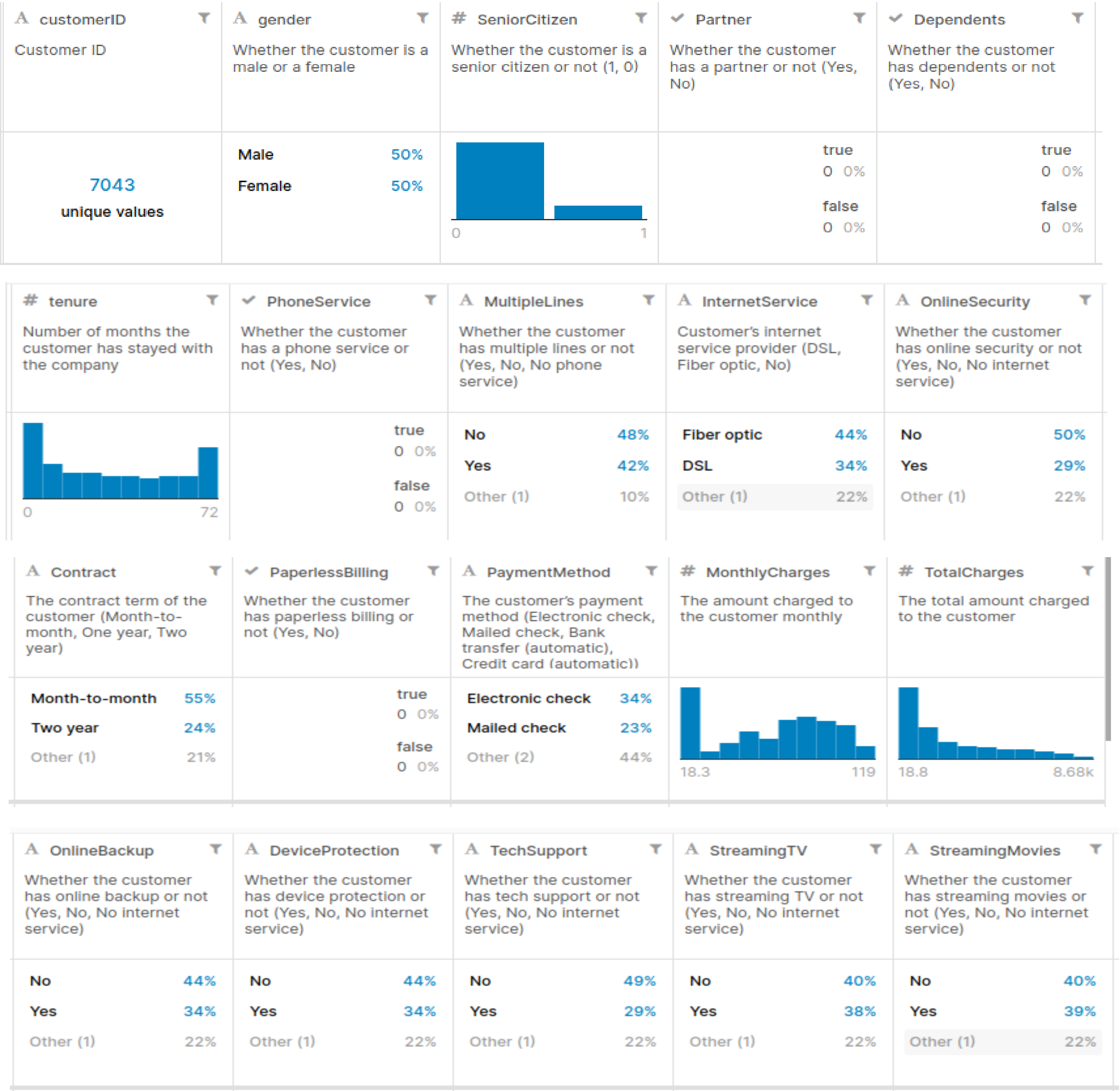
Assumptions:

1. It may be the case that customers have not paid any total charges or charges may be waived out by the company (kind of offer).

2. These gap may be left out accidently.

for now we will assume first case and fill those blank gap with 0.

3.2 Defination of Each Features Columns:



customerID

Customer ID

7043
unique values

gender

Whether the customer is a male or a female

Male	50%
Female	50%

SeniorCitizen

Whether the customer is a senior citizen or not (1, 0)

Partner

Whether the customer has a partner or not (Yes, No)

true
0 0%
false
0 0%

Dependents

Whether the customer has dependents or not (Yes, No)

true
0 0%
false
0 0%

tenure

Number of months the customer has stayed with the company

PhoneService

Whether the customer has a phone service or not (Yes, No)

true
0 0%
false
0 0%

MultipleLines

Whether the customer has multiple lines or not (Yes, No, No phone service)

No	48%
Yes	42%
Other (1)	10%

InternetService

Customer's internet service provider (DSL, Fiber optic, No)

Fiber optic	44%
DSL	34%
Other (1)	22%

OnlineSecurity

Whether the customer has online security or not (Yes, No, No internet service)

No	50%
Yes	29%
Other (1)	22%

Contract

The contract term of the customer (Month-to-month, One year, Two year)

Month-to-month	55%
Two year	24%
Other (1)	21%

PaperlessBilling

Whether the customer has paperless billing or not (Yes, No)

true
0 0%
false
0 0%

PaymentMethod

The customer's payment method (Electronic check, Mailed check, Bank transfer (automatic), Credit card (automatic))

Electronic check	34%
Mailed check	23%
Other (2)	44%

MonthlyCharges

The amount charged to the customer monthly

TotalCharges

The total amount charged to the customer

OnlineBackup

Whether the customer has online backup or not (Yes, No, No internet service)

No	44%
Yes	34%
Other (1)	22%

DeviceProtection

Whether the customer has device protection or not (Yes, No, No internet service)

No	44%
Yes	34%
Other (1)	22%

TechSupport

Whether the customer has tech support or not (Yes, No, No internet service)

No	49%
Yes	29%
Other (1)	22%

StreamingTV

Whether the customer has streaming TV or not (Yes, No, No internet service)

No	40%
Yes	38%
Other (1)	22%

StreamingMovies

Whether the customer has streaming movies or not (Yes, No, No internet service)

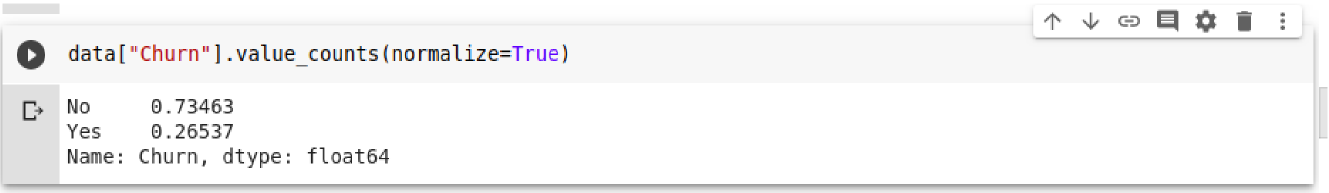
No	40%
Yes	39%
Other (1)	22%

source of above image: https://www.kaggle.com/blastchar/telco-customer-churn

3.3 Distribution of Target Feature:

The dataset we have, is imbalance in nature as the percentage of customer that churn is approximately one third of customer that do not churn. Even if our model learn "No" in our case the accuracy of model will be approximately 73 percent which sounds standard result but in actual it is very dangerous. The problem arises here is that our model will bias towards predicting "No" as the amount of data containig "No" churn is high (It will overfit in predicting No). To avoid this we can do upsampling or downsampling. Since the amount of data we have is less so will favour first approach, that is upsampling.

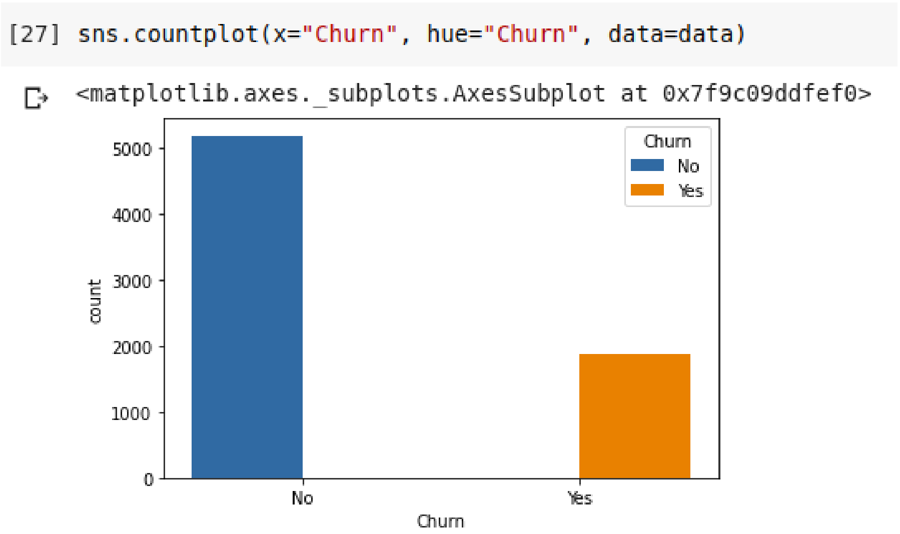
Source of image: Author



```
data["Churn"].value_counts(normalize=True)
```

```
No      0.73463
Yes     0.26537
Name: Churn, dtype: float64
```

Bar plot showing Distribution of Target feature:



```
[27] sns.countplot(x="Churn", hue="Churn", data=data)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f9c09ddfef0>
```

Source of image: Author

Here Bar having Blue colour represent count plot of customer who do not churn with the company while on the other hand orange colour bar represent the count plot of customer who are expected to churn with the company.

Chapter 5

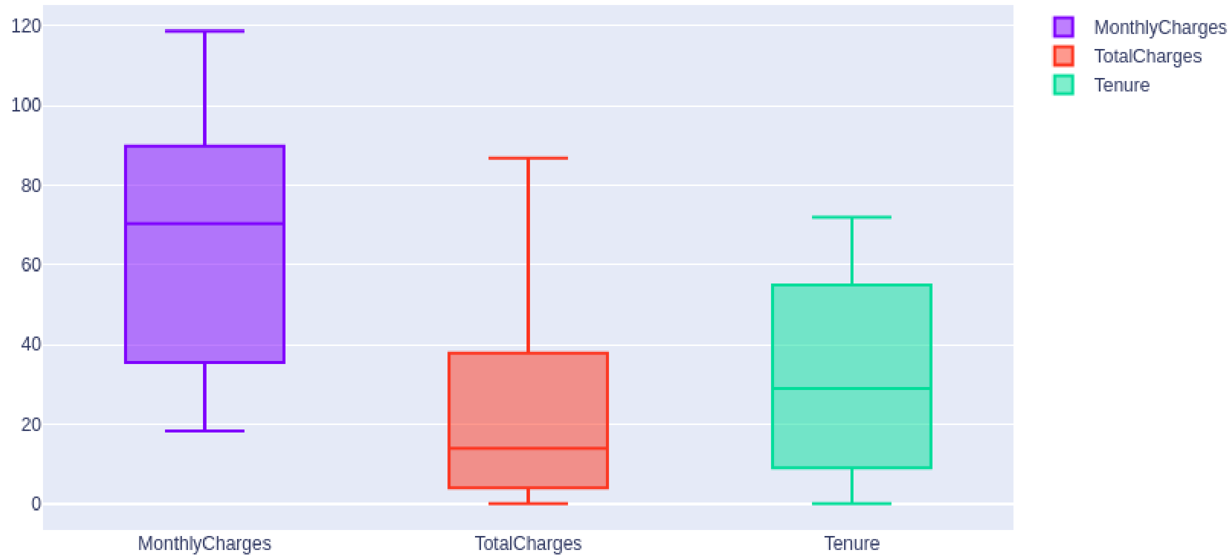
Data Analysis:

4.1 UNIVARIATE ANALYSIS:

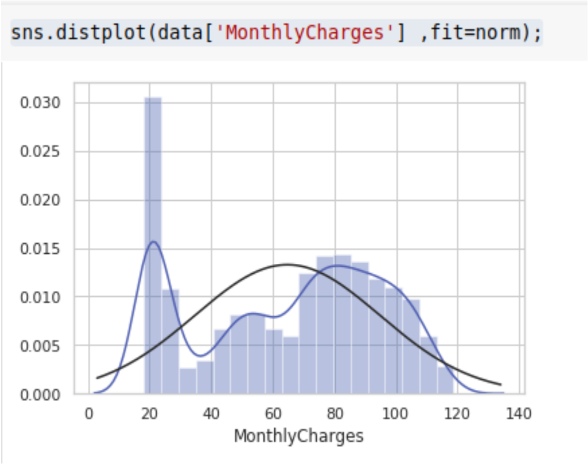
After performing data cleaning we then move to exporing each feature. We first start with univariate analysis. In this section we check out whether outliers are present in our dataset or not, Skewness and krutosis of each numerical features. For checking presence of outlier we plot, boxplot of all the numerical features present in our dataset.

The box-plot is shown below:

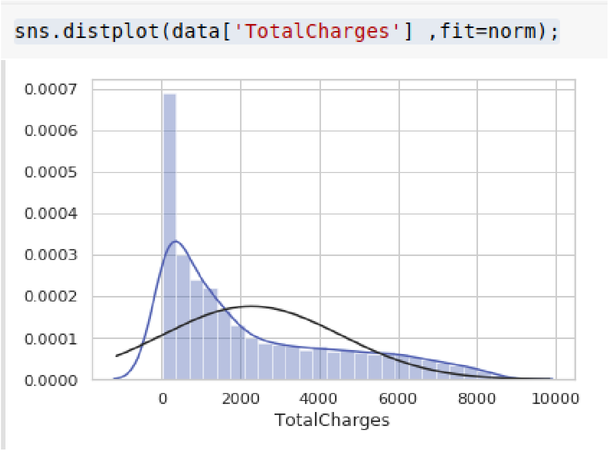
Source: Author



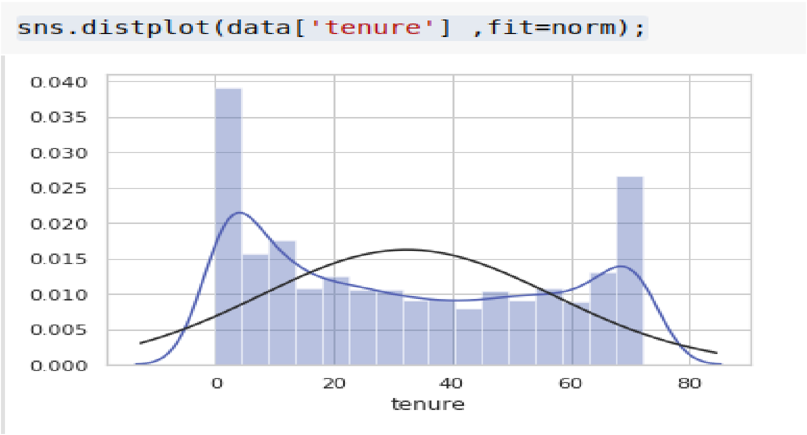
From the above figure we can conclude that outliers are absent in our dataset. We again plot the distribution-plot of each numerical features to find out the type of krutosis and skewness present in it. Results are shown below:



```
sns.distplot(data['MonthlyCharges'] ,fit=norm);
```



```
sns.distplot(data['TotalCharges'] ,fit=norm);
```



```
sns.distplot(data['tenure'] ,fit=norm);
```

source : Author

From the above plot we can conclude that

* Monthly Charges have Multimodal distribution with Leptokurtic nature.

* Total Charges have distribution of Leptokurtic in nature and have positively skewness.

* Tenure have Bimodal distribution with Leptokurtic nature . And the distribution is little bit right skewed.

* Customers with low TotalCharges, MonthlyCharges and Tenure are in majority.

* Count of customer decrease approximately exponentially

as Total Charges increases.

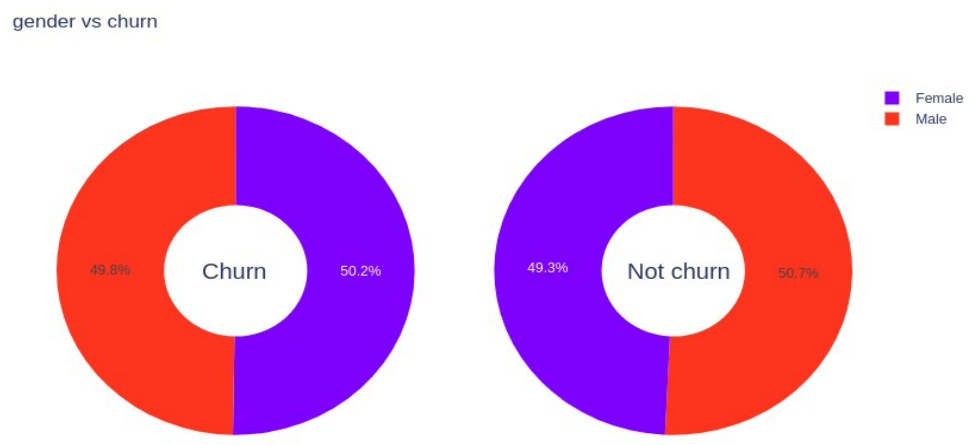
* Count of customers with intermediate tenure is almost constant.

4.2 BIVARIATE ANALYSIS:

In bivariate analysis we will find the relationship between every features with the target feature. For this we will plot piechart representing relationship betwen target variable and every features. Also we will find the correlation of every features with target variable.

Pie Chart are shown below:

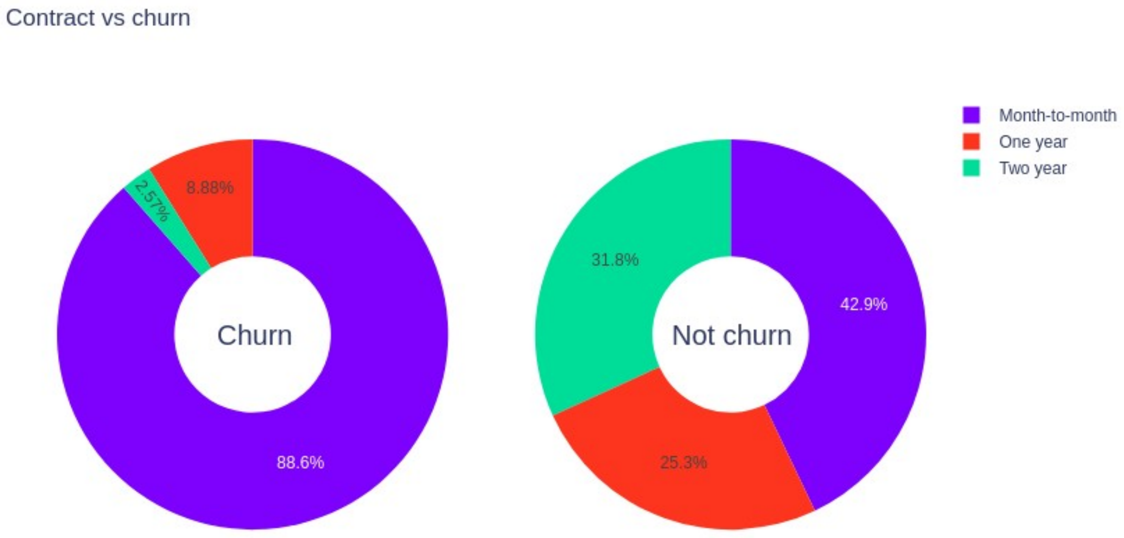
1. GENDER VS CHURN



CONCLUSION:

Above figure shows that Male and Female customers are equally expected to churn. Hence Gender is not an indicative of churn according to given data.

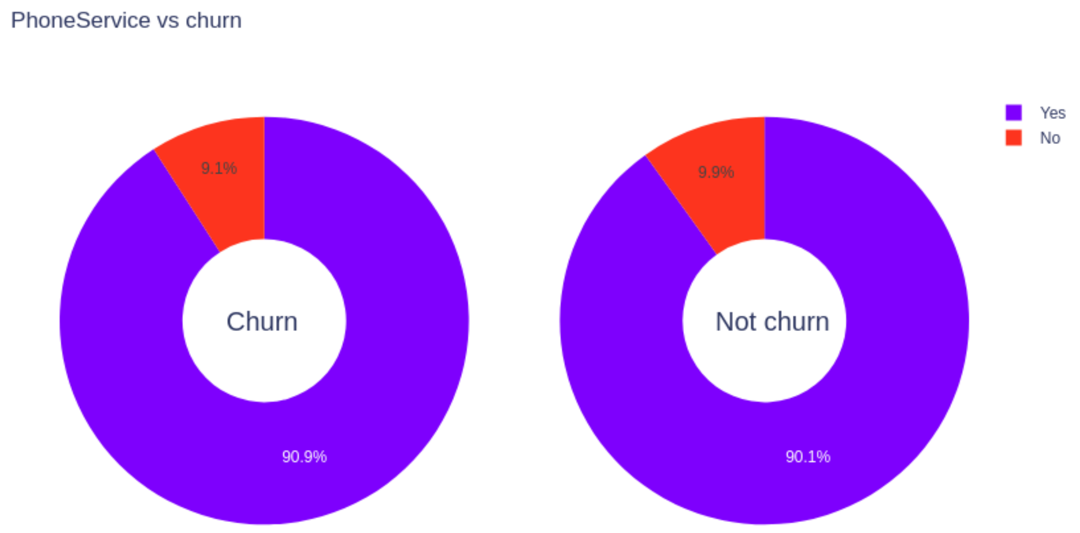
2.CONTRACT VS CHURN:



CONCLUSION:

Customers having month to month contract are more expected to churn then customers having one year and two year contract.

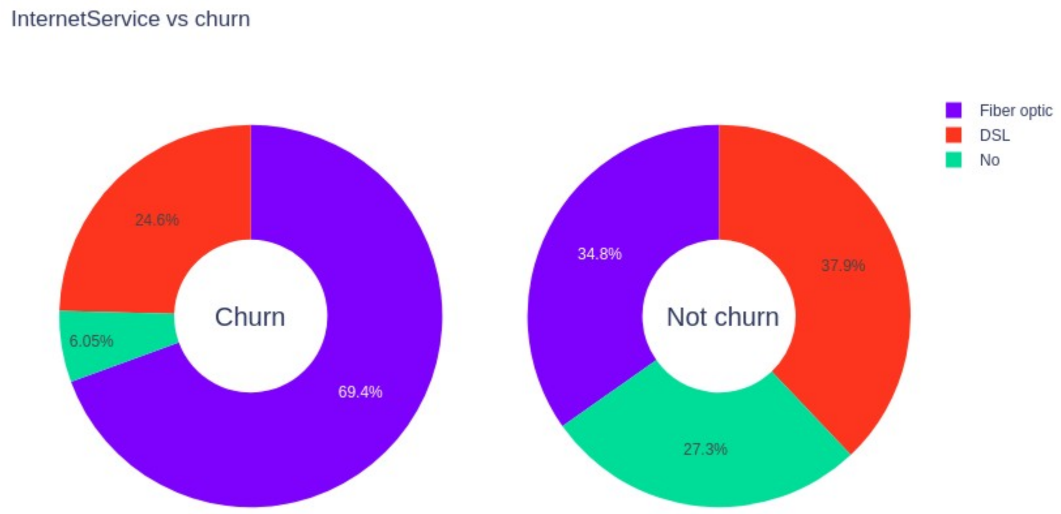
3. PHONE SERVICE VS CHURN:



CONCLUSION:

Customer having phone service is more likely to churn. In our data set approximately 91% customer using phone service is likely to churn.

4. INTERNET SERVICE VS CHURN



CONCLUSION:

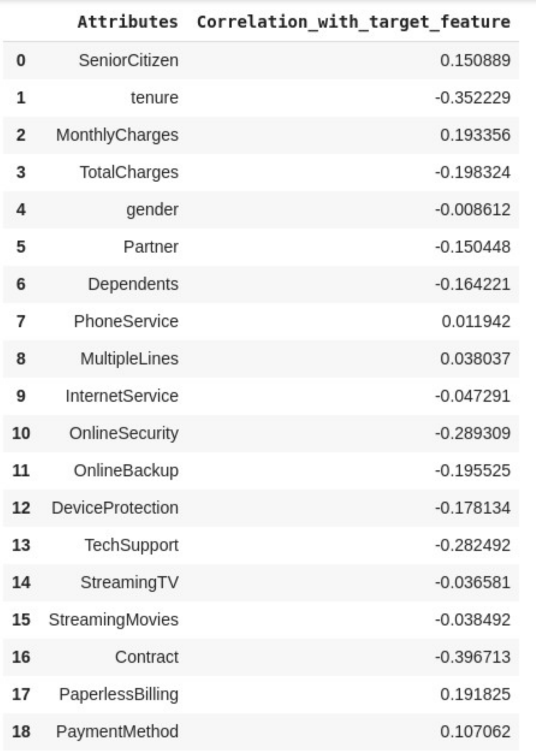
source of image: Author

1.Customer without internet has low a churn rate.

2.Customer with optical fibre based internet service is more expected to churn.

Since the datasets contains 21 features which is not possible to show piechart of each one so for further visualization you can refer to notebook created by us whose link is given below: https://colab.research.google.com/drive/1XV5YVq2zfyBQ9N14vIxMzGSY6yt582rX

Now we will move to explore linear dependency of each features with the target feature. For this we calculate correlation of every single feature with the target feature which is shown below:



	Attributes	Correlation_with_target_feature
0	SeniorCitizen	0.150889
1	tenure	-0.352229
2	MonthlyCharges	0.193356
3	TotalCharges	-0.198324
4	gender	-0.008612
5	Partner	-0.150448
6	Dependents	-0.164221
7	PhoneService	0.011942
8	MultipleLines	0.038037
9	InternetService	-0.047291
10	OnlineSecurity	-0.289309
11	OnlineBackup	-0.195525
12	DeviceProtection	-0.178134
13	TechSupport	-0.282492
14	StreamingTV	-0.036581
15	StreamingMovies	-0.038492
16	Contract	-0.396713
17	PaperlessBilling	0.191825
18	PaymentMethod	0.107062

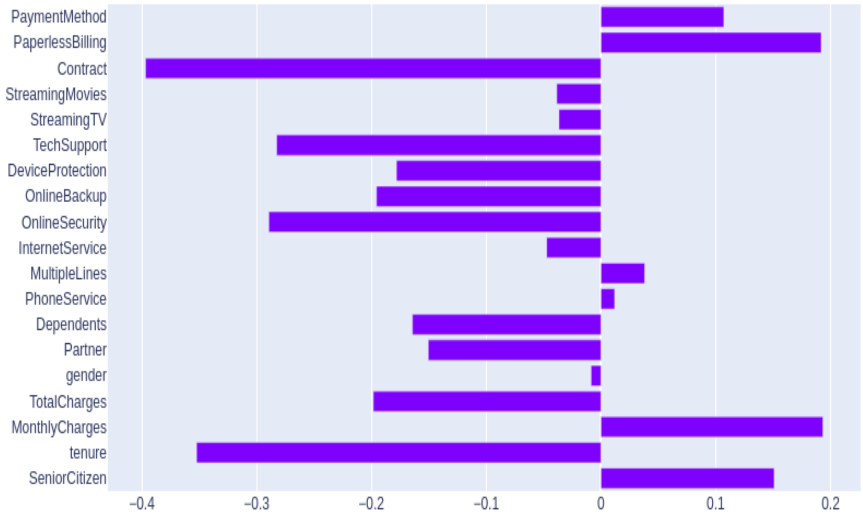


fig. showing correlation of every individual features with target feature

source: Author

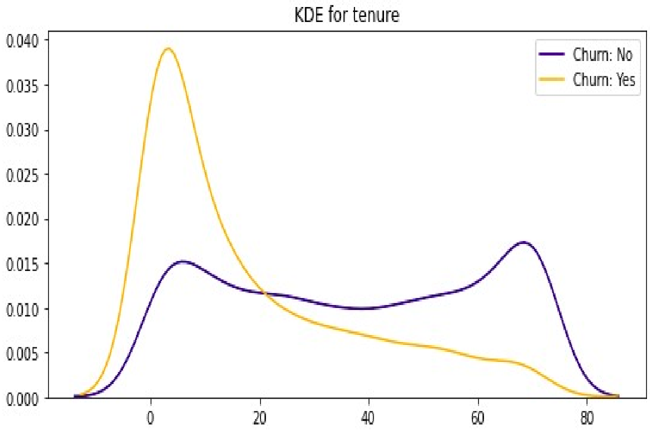
CONCLUSION:

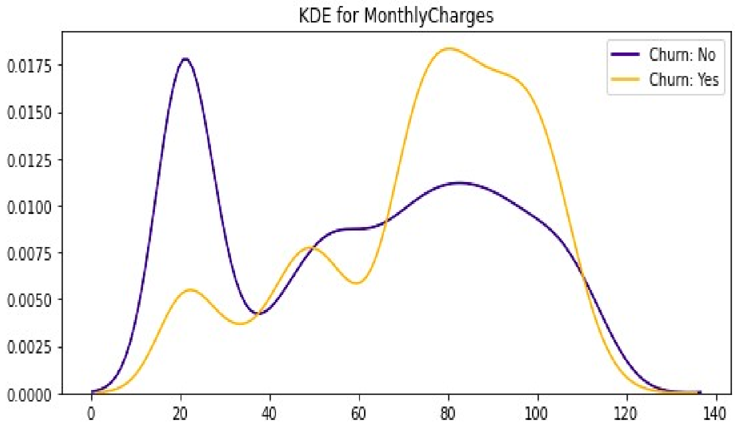
From above figure we conclude that the correlation coefficient of contract with churn is highest among all other features which shows that churn is linearly dependent on contract the most.

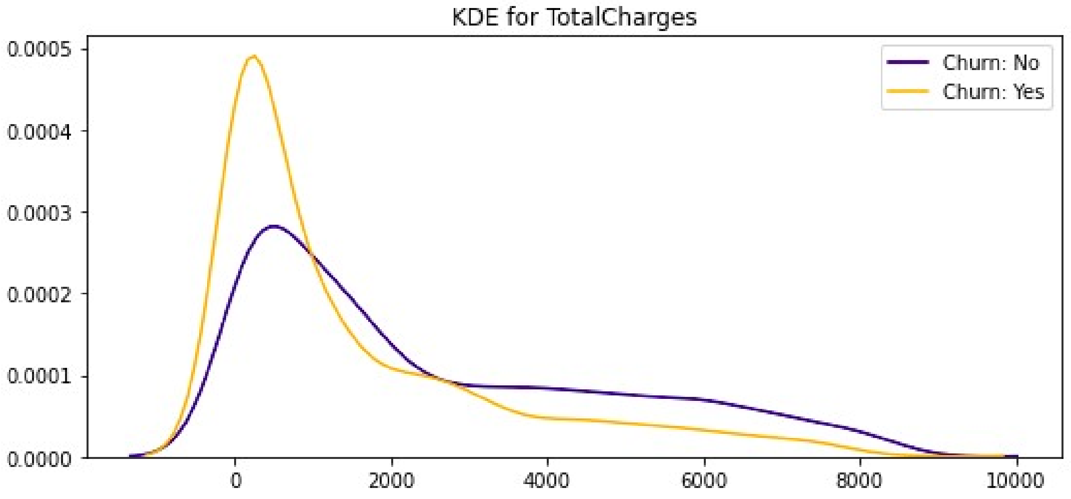
Then tenure, total Charges, Tech Support, Online security also have linear dependency with churn with corrleation coeffircent ranging from (0.38 - 0.28) .

Features having very low correlation coefficient with churn does not means that they are of no use, There may be nonlinear dependency among them.

Next we group the numerical feature on the basis of churn and not churn then plot their distribution, we find some interesting pattern among them which is shown below:







CONCLUSION:

1. Customers having small tenure value are more expected to churn.

2. Customers having low monthly charge value is less expected to churn and the customers having high monthly charge are more expected to churn.

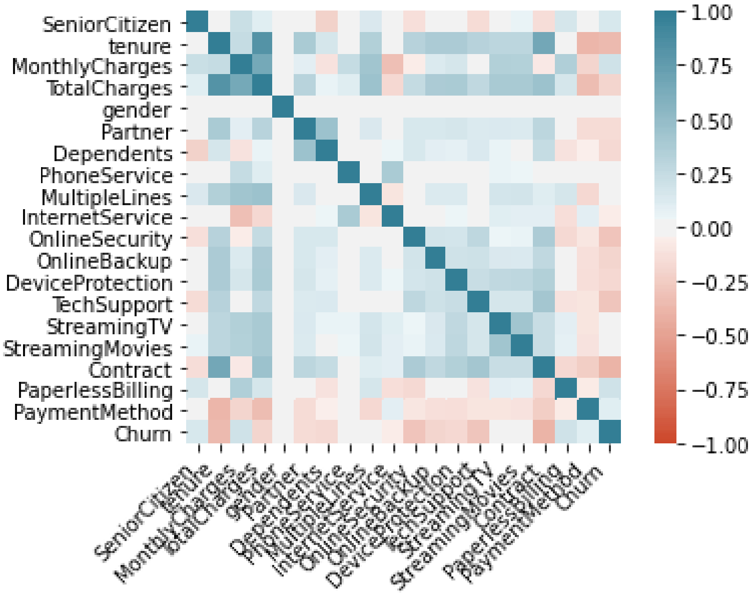
3. It is interesting to see that customers having low TotalCharges are more expected to churn wich is just opposite to that in case of Monthly charges.

(KDE plot is basically a probablity distribution plot)

4.2 MULTIVARIATE ANALYSIS:

In multivariate analysis we will see the correlation coefficient among all the features and will drop one feature if a group of two features have high correlation coefficent. Because both the features contain almost same information and it is proven that correlated feature make the model worse.

For this we plot correlation matrix among all the feature:



source of image: Author

fig. Correlation-heatmap showing correlation-coefficient among every feature

CONCLUSION:

* From the above plot we can see that tenure have high correlation coefficent with TotalCharges and contract. If without these feature, model have better performance then we will simply drop those features. Also monthly charges and Total charges have high correlation coeffiecient which is obvious as it can be linearly related.

Chapter 6

Predictive Modelling:

5.1 Splitting the data into train and test data.

We have used sklearn train test split library to split the the data into 30 percent test data and 70 percent train data. Also to insure that the distribution of target feature remain same in both train and test we did stratified split with the data. Then fit MinMax scaler on train data (for preprocessing). Same MinMax scaler is then applied to transform the test data.

After that we used different model for prediction on test data and find the accuracy, precision, and recall on the test data.

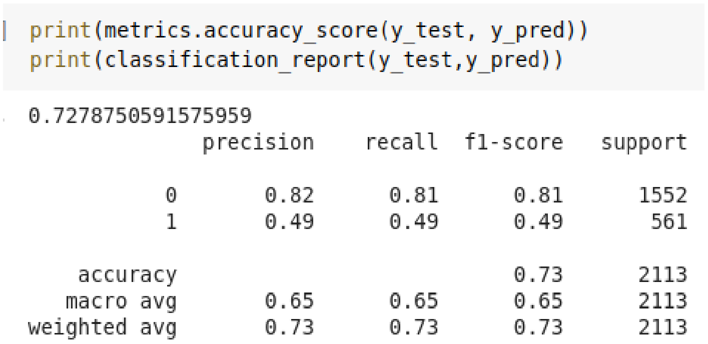
Summary of all the outcomes of the models are given below:

5.2 Models:

5.2.a Decision Tree:

Metric used for evaluation: Accuracy, Precision, Recall.

Below diagram shows the outcome of decison tree model:

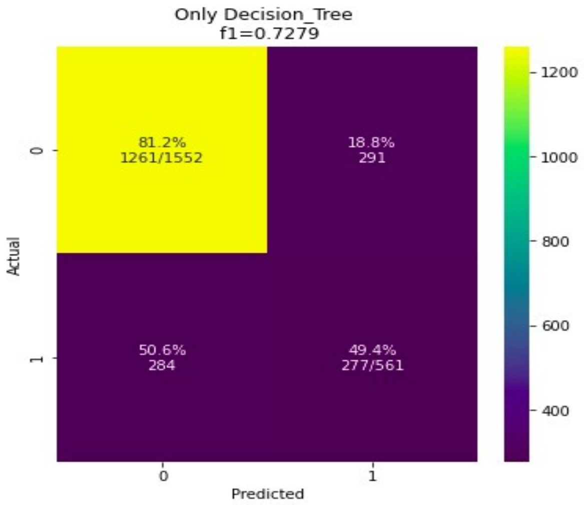


```
print(metrics.accuracy_score(y_test, y_pred))
print(classification_report(y_test,y_pred))
```

```
0.7278750591575959
              precision    recall  f1-score   support

           0       0.82      0.81      0.81      1552
           1       0.49      0.49      0.49       561

    accuracy                           0.73      2113
   macro avg       0.65      0.65      0.65      2113
weighted avg       0.73      0.73      0.73      2113
```



source of image: Author

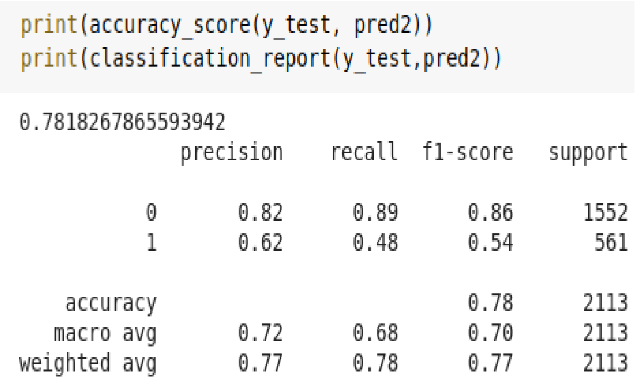
As we can clearly see that recall and precision of our model for minority class (1 in our case) is very poor. Overall accuracy that we have got is approx 73% .

We then try some advance algorithm for our task.

5.2.b Random Forest: (Advance form of decision tree which create bunch of decision tree, then ensamble all of them to make prediction).

Metric used for evaluation: Accuracy, Precision, Recall.

Below diagram shows the outcome of Random forest:

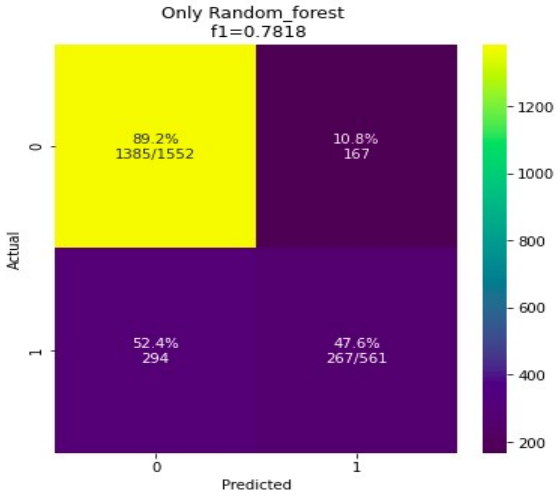


```
print(accuracy_score(y_test, pred2))
print(classification_report(y_test,pred2))
```

```
0.7818267865593942
              precision    recall  f1-score   support

           0       0.82      0.89      0.86      1552
           1       0.62      0.48      0.54       561

    accuracy                           0.78      2113
   macro avg       0.72      0.68      0.70      2113
weighted avg       0.77      0.78      0.77      2113
```



source of image: Author

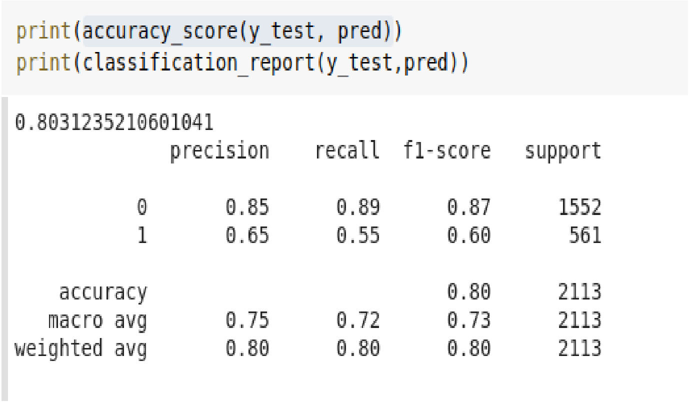
As we can see that accuracy increases to 78% . Precision of Minority class (1 in our case) has also increased to 62% but recall of minority class got lesser then earlier model.

We have also experimented with other models such as light Gbm, Xgboost but accuracy does not improve in all those cases. But finally a simple Logistic Regression works best for our task.

5.2.c Logistic Regrssion:

Metric used for evaluation: Accuracy, Precision, Recall.

Below is the snapshot of outcomes of logistic regression:

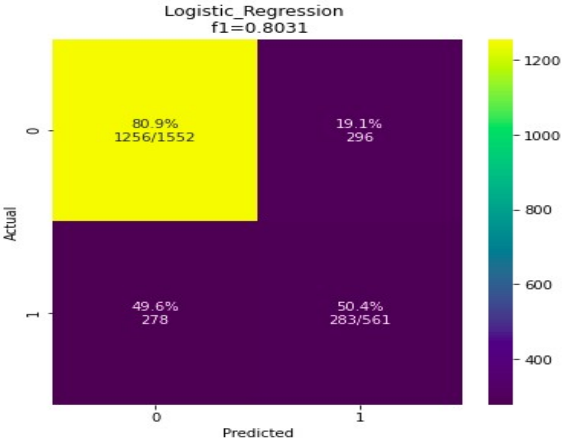


```
print(accuracy_score(y_test, pred))
print(classification_report(y_test,pred))
```

```
0.8031235210601041
              precision    recall  f1-score   support

           0       0.85      0.89      0.87      1552
           1       0.65      0.55      0.60       561

    accuracy                           0.80      2113
   macro avg       0.75      0.72      0.73      2113
weighted avg       0.80      0.80      0.80      2113
```



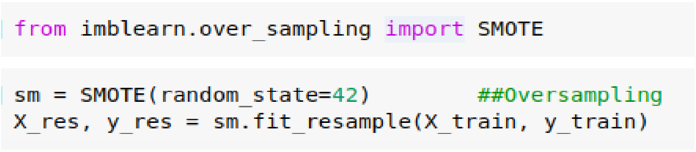
source: Author

As we can see that accuracy improves a lot, also precision and recall of minority class have improved a lot.

Since Logistic Regression works best for our task, we select this model for further experiment.

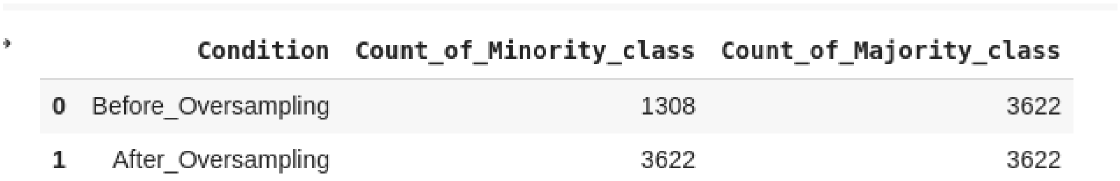
5.3 Over Sampling with SMOTE:

Smote is an inbuilt library in imblearn which generate sythetic data based on the distribution of training data to equilize the number of minority class with number of mazority class.



```
from imblearn.over_sampling import SMOTE
```

```
sm = SMOTE(random_state=42)        ##Oversampling
X_res, y_res = sm.fit_resample(X_train, y_train)
```

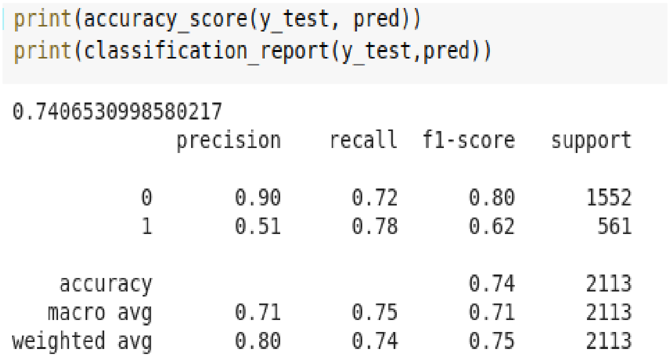


	Condition	Count_of_Minority_class	Count_of_Majority_class
0	Before_Oversampling	1308	3622
1	After_Oversampling	3622	3622

source: Author

Now we have balance dataset with equal number of minority and majority class.

We then fit the oversampled data to logistic regression and obtained the following result:

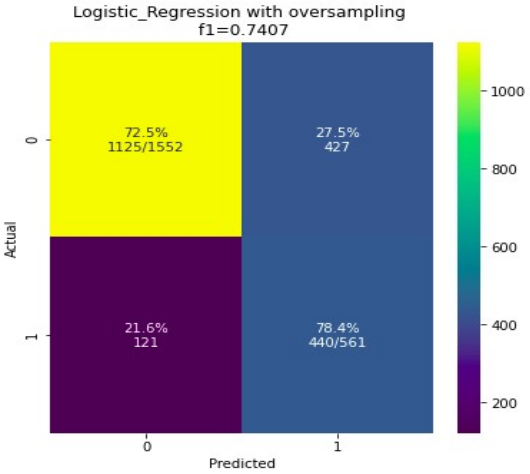


```
print(accuracy_score(y_test, pred))
print(classification_report(y_test,pred))

0.7406530998580217
              precision    recall  f1-score   support

           0       0.90      0.72      0.80      1552
           1       0.51      0.78      0.62       561

    accuracy                           0.74      2113
   macro avg       0.71      0.75      0.71      2113
weighted avg       0.80      0.74      0.75      2113
```



After oversampling we can see that there is huge increment in the recall of minority class(1 in our case), But overall accurcay decrease in this case.

After this we also try to predict the test data with class weight. In sklearn library class_weight is already available which find out what weight should be feed into the model for every class. After calculating class_weight we again train the Light GBM model with class weight and obtained the accuracy of 76% which is still poorer then classical logistic regression.

To see all those experiment you can refer to the notebook mentioned earlier in this section.

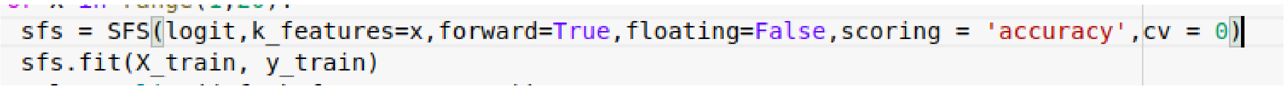
We then did some feature selection which leds to increase in the accuracy.

5.4 FEATURE SELECTION:

For feature selection we used Wrapper forward propagation method and obtained the best subset of whole feature. There is already a library named SequentialFeatureSelector

which do the same job. Below is some snapshot of our code or you may visit notebook created by us whose link is given below:

https://colab.research.google.com/drive/1XV5YVq2zfyBQ9N14vIxMzGSY6yt582rX

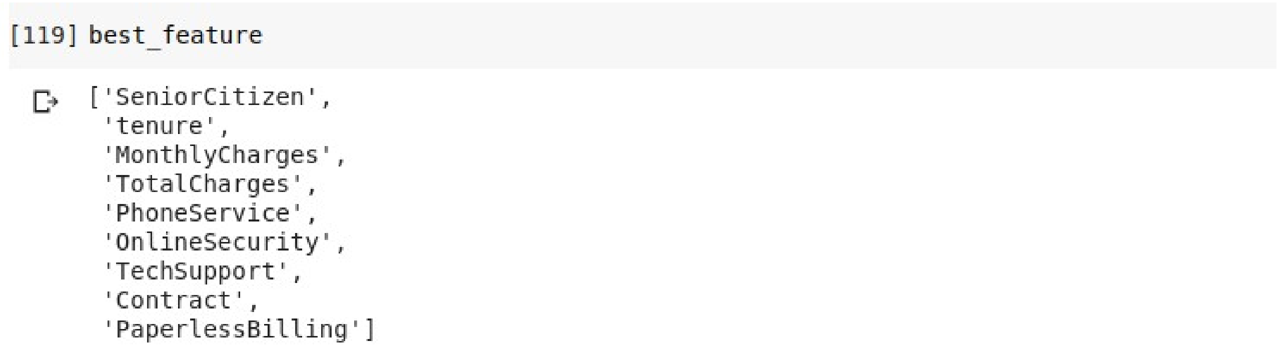


```
 sfs = SFS(logit,k_features=x,forward=True,floating=False,scoring = 'accuracy',cv = 0)
 sfs.fit(X_train, y_train)
```

Here logit represent model (logistic Regression in our case), k_features represent maximum number of feature we want from over all feature. For forward propagation, Forward is kept True and for backward propagation it is kept as False. Scoring criteria is selected as accuracy. And we keep number of cross validation as 0 as it takes lot of time for feature selection.

Since we don't kow the exact number of best features for our model hence we iterate over range 1 to total number of feature and select the best subset of feature which perform best for our model.

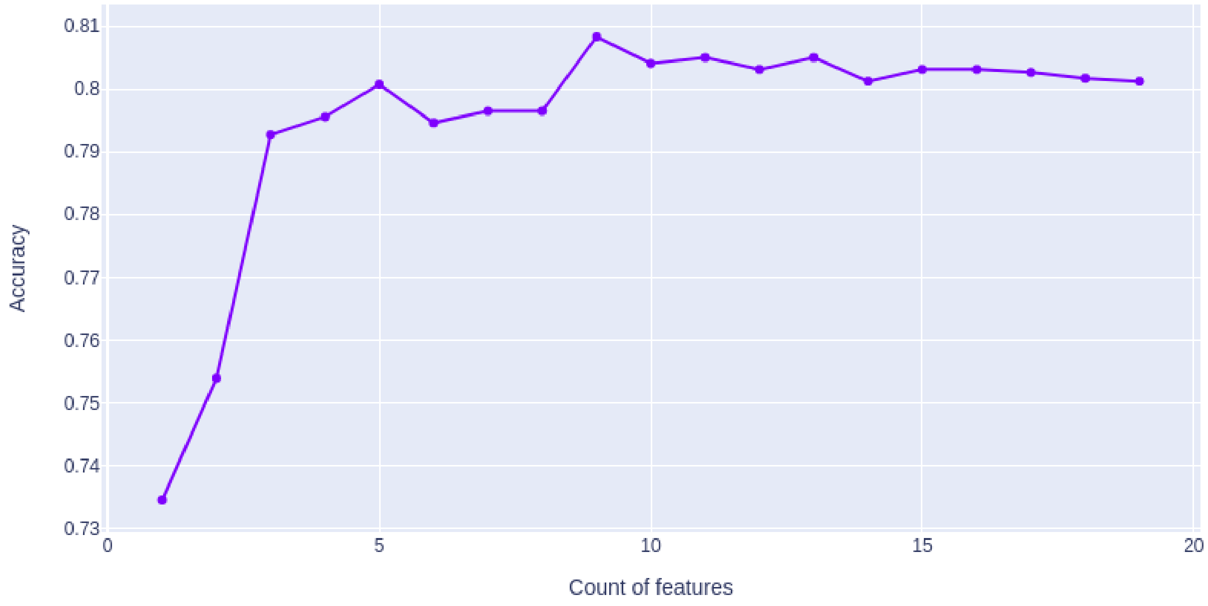
The Best Feature we obtained are :



```
[119] best_feature

['SeniorCitizen',
 'tenure',
 'MonthlyCharges',
 'TotalCharges',
 'PhoneService',
 'OnlineSecurity',
 'TechSupport',
 'Contract',
 'PaperlessBilling']
```

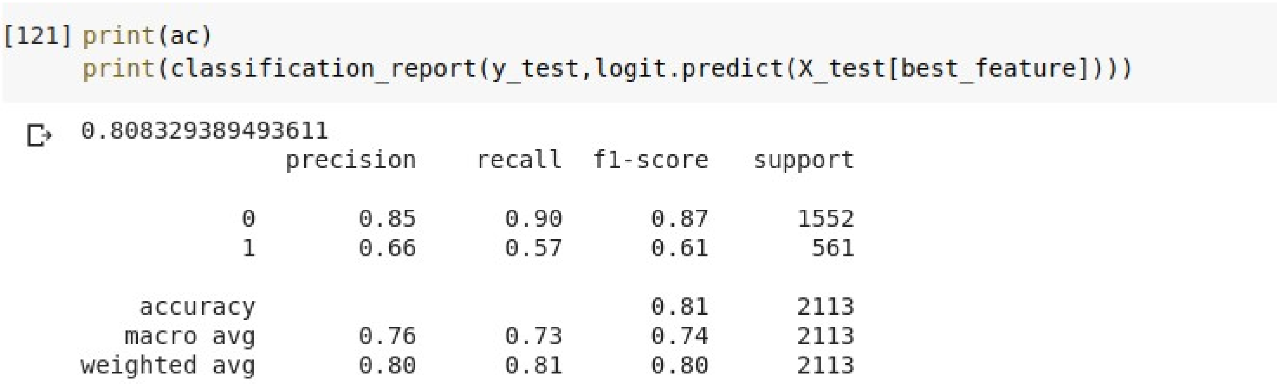
And then we plot the count of features vs accuracy and obtain the result which is shown below:



source: Author

From above diagram we can see that best subset of features contain total 9 features which leds to improve in performance of our model.

Below is the snapshot of what we have obtained after fitting the model on above selected subset of feature:



```
[121] print(ac)
      print(classification_report(y_test,logit.predict(X_test[best_feature])))
```

```
0.808329389493611
              precision    recall  f1-score   support

           0       0.85      0.90      0.87      1552
           1       0.66      0.57      0.61       561

    accuracy                           0.81      2113
   macro avg       0.76      0.73      0.74      2113
weighted avg       0.80      0.81      0.80      2113
```

We are able to get improve in accuracy with improve in precision and recall of minority class as well.

For now we are going to wrap up this section here. For future work you can refer to conclusion section.

Chapter 7

Conclusion:

We tried every possiblity and reach at follwing conclusion:

6.1. The best feature that helps in churn prediction are:

a. Senior Citizens: It is a kind of categorical feature that tells whether customers are senior-citizens or not. If a customers is a senior citizen then that particular row contain 1 else 0.

b. Tenure: It is a kind of numerical feature that tells the time duration, customer is associated with the bussiness.

c. Monthly Charges: It is a kind of numerical feature that tells the total monthly charges customer have to pay to company.

d. Total Charges: It is a kind of numerical feature that tells the total charges customer have to pay to company.

e. Phone Service: It is a kind of categorical feature that tells whether customers have phone services or not.

f. Online Security: It is a kind of categorical feature that tells whether customers have taken online security or not.

g. Tech Support : It is a kind of categorical feature that tells whether customers have access to technical support or not.

h. Contract : It is also a kind of categorical feature that tells what type of contract, customer have taken with the bussiness, such months, one year, two year and others.

i. Paper Less Billing: It is also a kind of categorical feature that tells whether customers have access to paperless billing or not.

On training the model on above feature we get best performance of our model:

i. Accuracy Score that we obtained is 80.83%.

ii. Precision of minority class is 66%.

iii. Recall of of minority class is 57%.

iv. Precision of majority class is 85%.

v. Recall of majority class is 90%.

WHAT ELSE WE CAN DO:

1. Since the dataset we have is very small. We can look for more data to improve the performance of models.

2. We can do ensambling, Stacking and Blending for further emprovement in accuracy.

3. Hyper-parameter tunning of all the tree's model such as Random Forest, XGBoost, Light GBM.

4. We can apply some new methods of feature selection.

Chapter 8

References:

[1]. Swetha Amaresan, “What Is Customer Churn？”

[2]. https://en.wikipedia.org/wiki/Customer_attrition

[3]. https://en.wikipedia.org/wiki/Churn_rate

[4]. Source of Dataset: https://www.kaggle.com/blastchar/telco-customer-churn/download