Customer Churn Analysis

With the development and growth of the telecommunication industry, the service providers are inclined more towards expansion of the subscriber base. To survive in the competitive environment, the retention of existing customers has become a huge challenge. It has been observed that keeping an existing customer is far less expensive than acquiring a new customer.

A high churn rate could adversely affect profits and can obstruct the growth. Churn rate is an important factor in the telecommunications industry. In most areas, many of these companies compete, making it easy for people to transfer from one provider to another. The churn rate not only includes when customers switch carriers but also includes when customers terminate service without switching. This measurement is most valuable in subscriber-based businesses in which subscription fees comprise most of the revenues.

**Problem Statement**

Customer retention is a big challenge in telecommunication industry. But somehow it can be achieved with good customer service and products. But the most effective way for a company to prevent attrition of customers is to truly know them. The vast volumes of customer data from IBM Sample Data Sets are provided to us which we’ll be using to build churn prediction models. Knowing who is most likely to defect means that a company can prioritise focused marketing efforts on that subset of their customer base.

**Data Set**

**Link to the dataset:** <https://github.com/dsrscientist/DSData/blob/master/Telecom_customer_churn.csv>

The customer churn dataset provided to us contains 7043 rows ,20 independent features and the dependent feature that is churn. We aim to predict if the current associated customer will continue using our services or not. Therefore, our target feature contains 2 values i.e. Yes or No. This implies that it is a classification problem, and we will be training the classification models to predict the desired outputs.

The dataset is consisting of different types of features which are mentioned below:

**customerID – contains unique id’s of the customers.**

**gender - gender of a person.**

**SeniorCitizen - if the customer is senior citizen or not.**

**Partner - customer is having a partner or not.**

**Dependents - whether the customer relies on somebody else or not.**

**tenure - total amount of time the customer is associated with the company**

**PhoneService - Does the customer has phone service.**

**MultipleLines - Does the customer has multiple lines.**

**InternetService - Kind of internet service customer is using.**

**OnlineSecurity – Do they have online security.**

**OnlineBackup – Do they use online Backup .**

**DeviceProtection – Do they have any device protection**

**TechSupport - Do they have tech support.**

**StreamingTV - Do they use streaming tv feature.**

**StreamingMovies – Do they use streaming movies feature.**

**Contract - What kind of contract are they having.**

**PaperlessBilling - opted paperless billing.**

**PaymentMethod - Kind of payment method.**

**MonthlyCharges - Amount charged monthly**

**TotalCharges - Total amount charged**

**Churn - customer churn**

**Content of the Article**

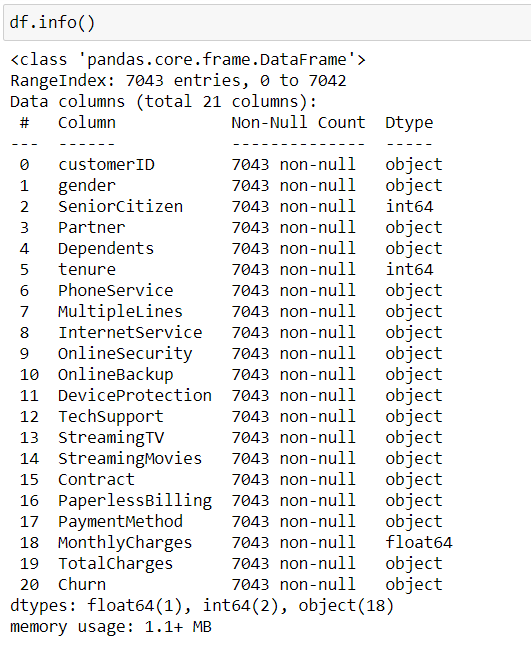
In this article we’ll be focussing on below mentioned topics:

1. Data cleaning and handling missing values
2. Exploratory data analysis
3. Data-Preprocessing
4. Model Buiding
5. Cross validation scores
6. Hyper parameter tuning of the selected model
7. Saving the model
8. Conclusion

First, we’ll import the essential libraries which we’ll be using in various data analysis.

Afterwards we’ll be importing our data set and will store it in a variable “df”.

Now, We’ll check the basic information associated with the dataset.

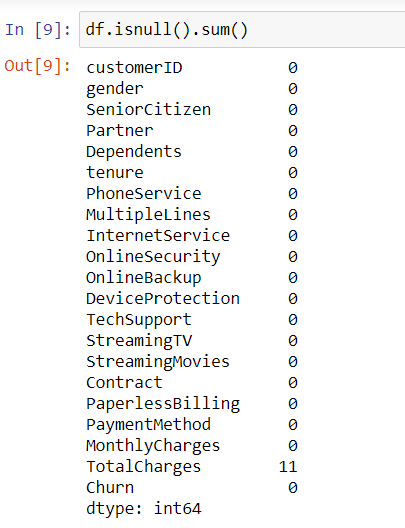


By using df.info() we obtained the following information about each feature.as we can see that all the features are mixture of object, integer and float data type. Around 18 features are of object data type. 1 feature is of float data type and rest of them are integer data type. We also observed that the count of non- null values of every feature is 7043 which is exactly same as the number of rows, which means no null values are present in any of the feature. We’ll check this in our next step.

1. **Data cleaning and handling missing values**

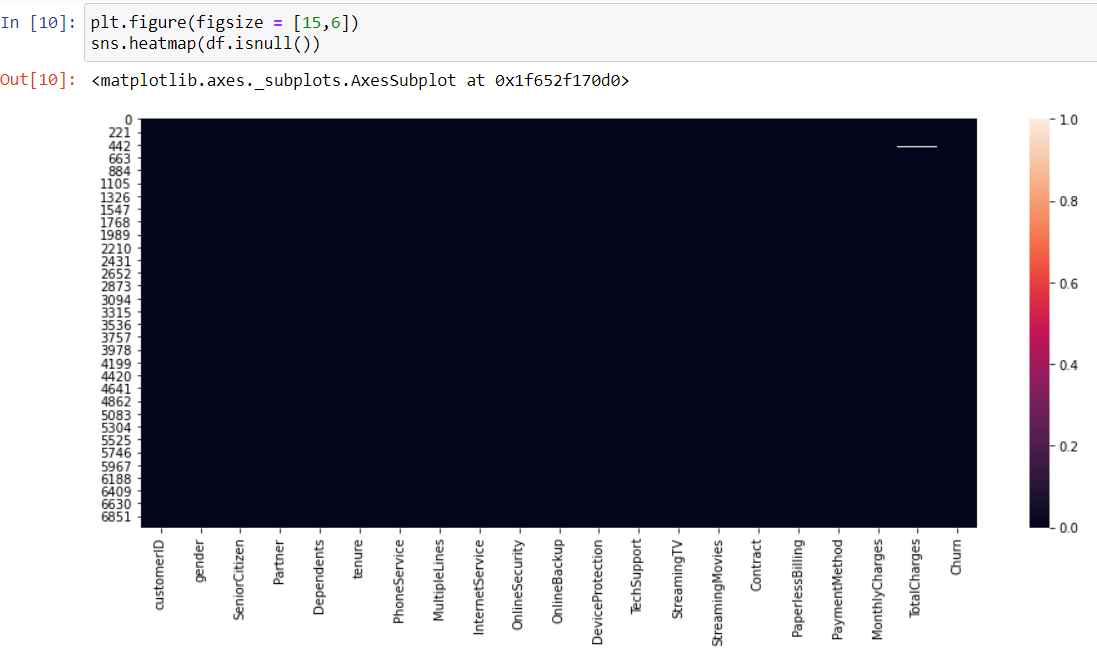
In this section we’ll see whether our dataset contains null values or not using 2 methods, mentioned below :

1. By using the command “df.isnull().sum()”.This returns the number of null values present in each features.



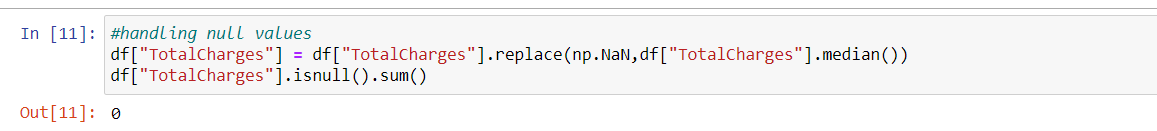
Here we observed that 11 null values are present in the Total Charges feature, earlier which showed there are no null values. So we’ll be handling this missing values

1. By using heatmaps visualization method.



Since there are white spots or boxes (represents null values) present in the graph. therefore, null values are present in the feature 'Total Charge'.

Now, I’ll remove these null values using the following method:

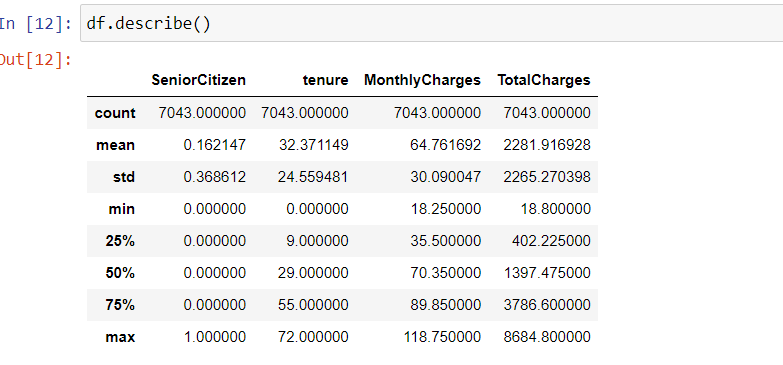


Here I used .replace function to replace all the null values(np.NaN) with the median of the TotalCharges feature.

1. **Exploratory data analysis**

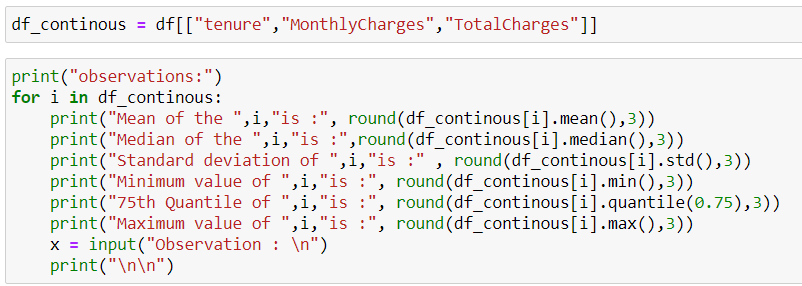
In the beginning of the article I used df.info() to get basic information about the dataset.

Now , I used df.describe() function to get the statistical analysis of the data set.

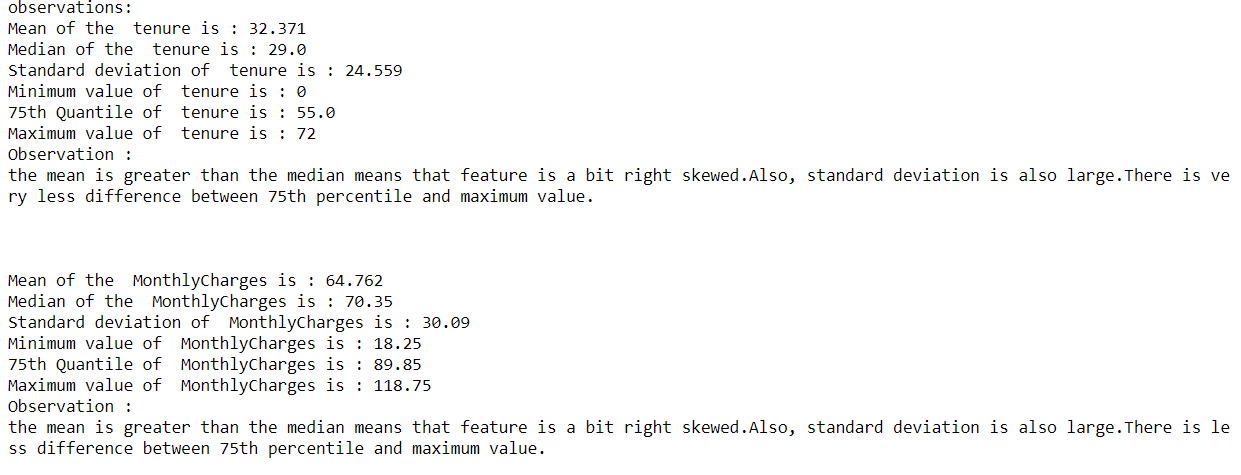


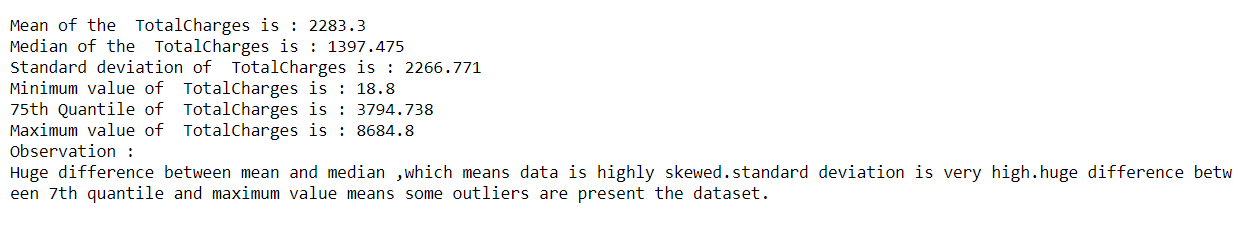
The first observation I made was Senior citizen comprises of 0 and 1 which implies that the customer is not a senior or a senior citizen respectively. Right now its of int type so we'll be converting it to object type.

For writing other observation ,first I created a list which contains only continuous data and then I used a piece of code mentioned below.



The observations made were as follows:





After this I used visualization tools to get insights from the categorical feature(those features consists of only 0’s and 1’s ,Yes or No or other non-numerical values)

Firstly ,I made a list of all categorical features and then plotted them using count plot.

Then I analysed the following features individually and developed these following insights.

**Gender:**

The gender features consist of 2 categories i.e. Male and Female . Also, I observed that the count of both the categories are almost equal.



**Dependents**

This feature is also consisting of two categories. Here I observed that the most of the individuals are not dependent.



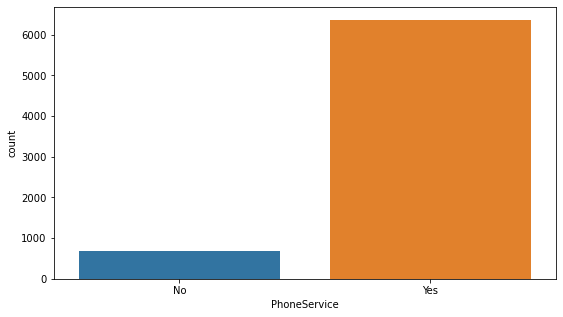
**Partner**

This feature is also consisting of two categories. Here also we can see that the ratio of yes and no is almost same.



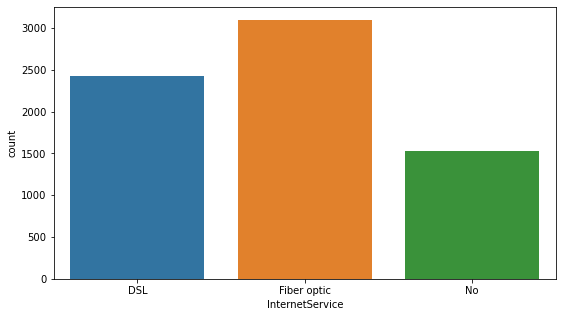
**PhoneService**

This feature is also consisting of two categories. Most of the individuals prefer phone services.



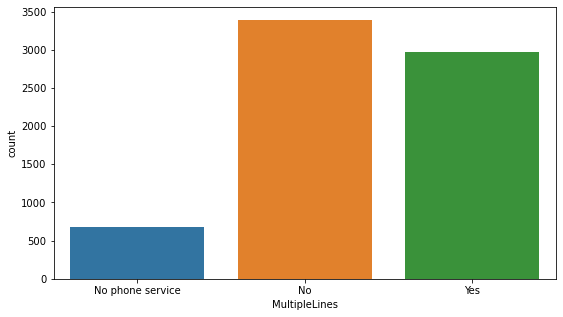
**Internet Service**

As we can see that the feature is consisting of three categories. Where most of the individuals prefer Fiber optic internet service.



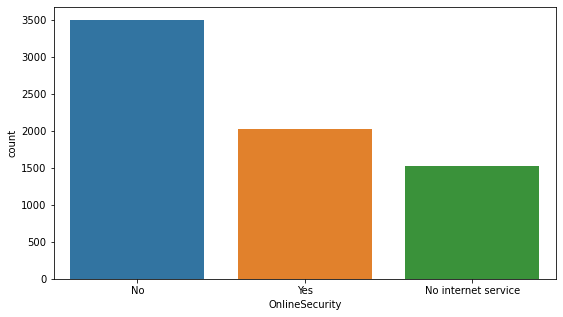
**Multiple Lines**

Here we can see that more than 3000 individuals has no multiple lines but at the same time around 3000 people also has multiple lines



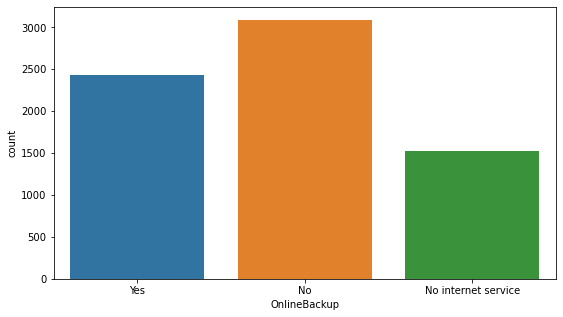
**Online Security**

Here we can see that most of the individuals do not have online security but still many people prefer having online security.



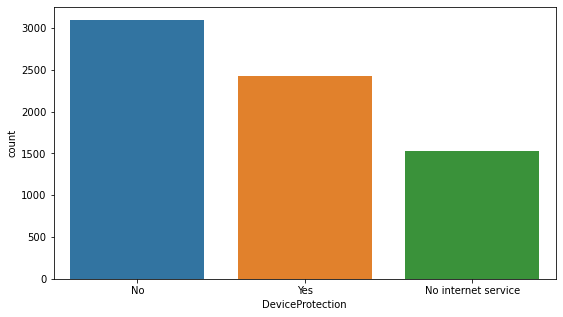
**Online Backup**

Here we can see that most of the individuals do not prefer having online backup but still around 2500 people prefer having online backup.



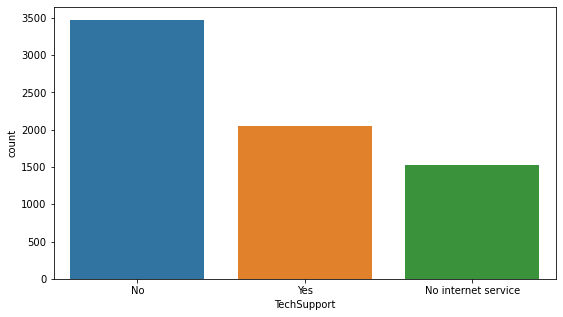
**Device Protection**

Here we can see that most of the individuals do not have any device protection. And around 2500 people prefer having Device protection.



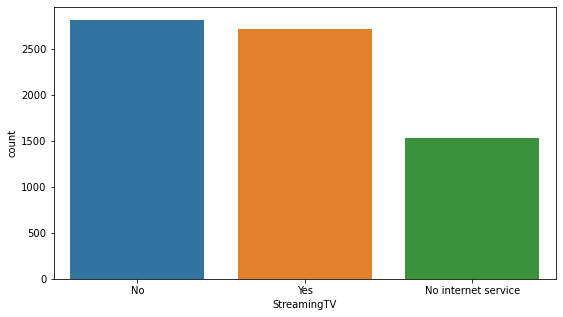
**Tech Support**

Here we can see that most of the individuals do not prefer any tech support. Still less than 2000 people prefer having tech support.



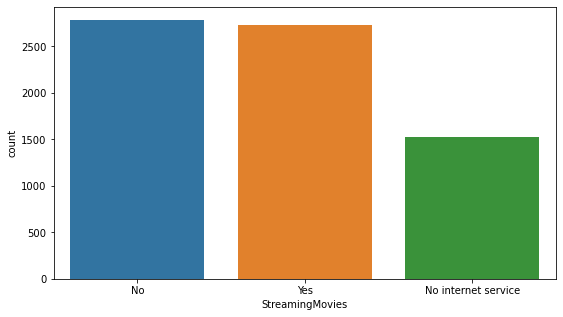
**Streaming TV**

The number of individuals who are streaming TV are almost similar to those who do not use streaming tv service.



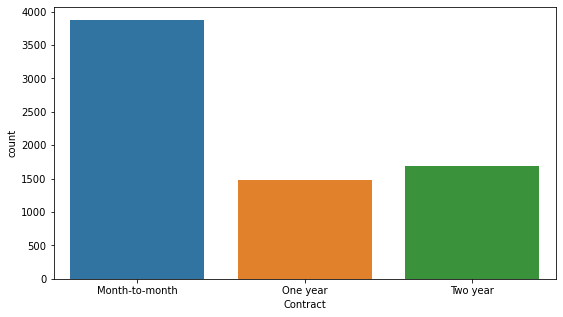
**Streaming Movies**

The number of individuals who are streaming movies are almost similar to those who do not use streaming movies service.



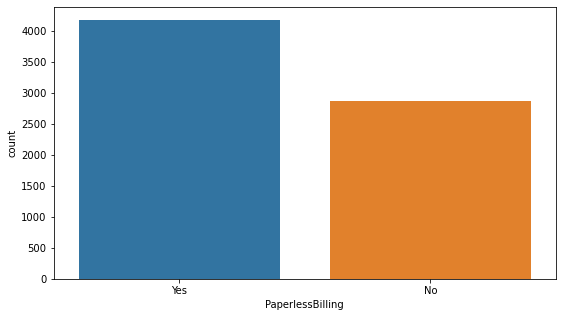
**Contract**

Here we saw that most of the individuals prefer month to month contract over other contracts. Surprisingly people prefer two year contract over one year contract.



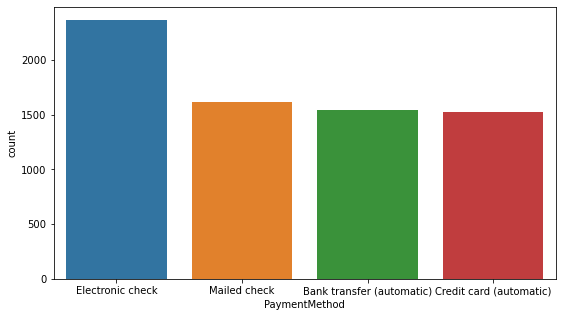
**Paperless Billing**

Here we saw that most of the customers prefer paperless billing. And around 2500 customers still prefer paper billing.



**Payment Method**

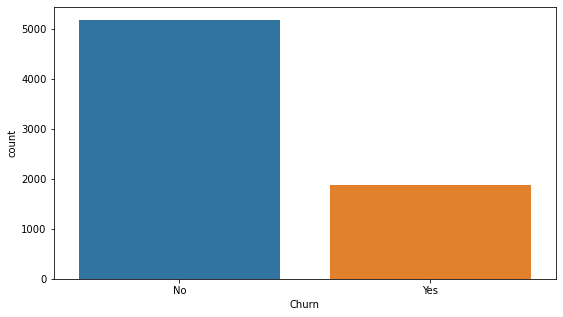
This feature consists of 4 categories. In this feature most of the people prefer Electronic check over other payment methods.



**Customer Churn**

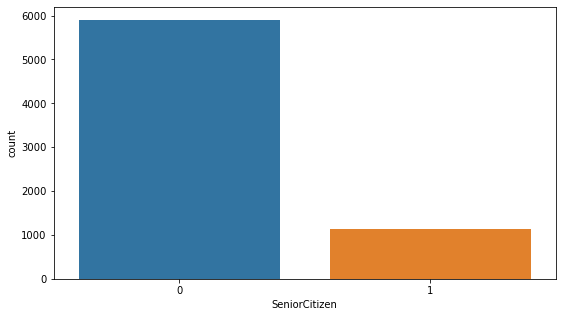
This feature is our Target feature which consist of two categories, i.e., Yes or no. the most of the customers didn’t stop using the services of this company, which is a good sign for the company.

But having so much imbalance in the two categories, there are chances that our model might get biased for one category. So, to overcome this problem I used sampling technique to balance the data in further steps.



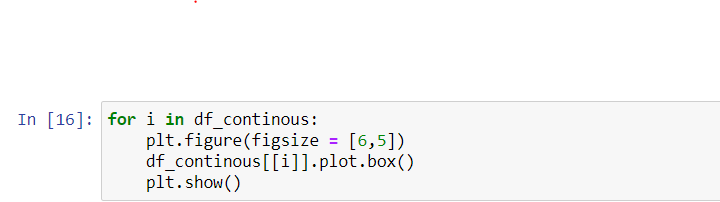
**Senior citizens**

This feature consists of 2 categories. The customers are either senior citizen or not .And I observed that maximum number of the customers are not senior citizen.

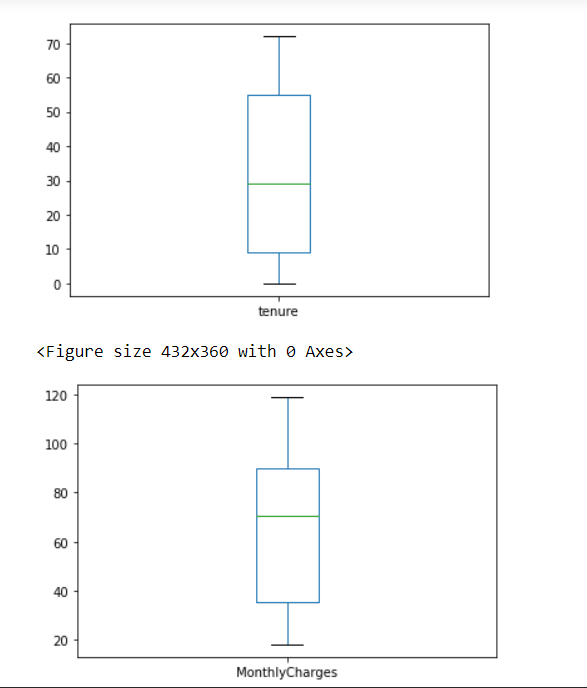


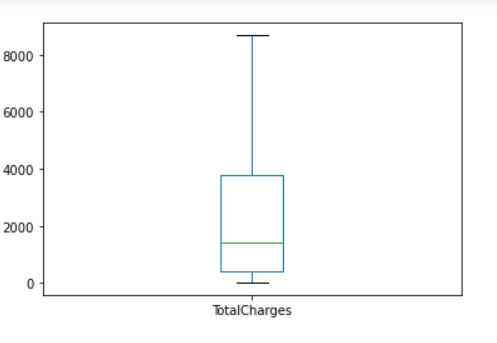
After that we checked that if there are any outliers present in the continuous features or not.

* To plot the box plot of different features at one time I used the following code,



* The following output was obtained after running the code:

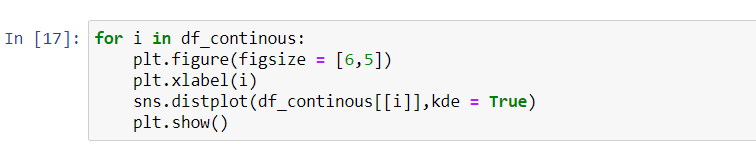




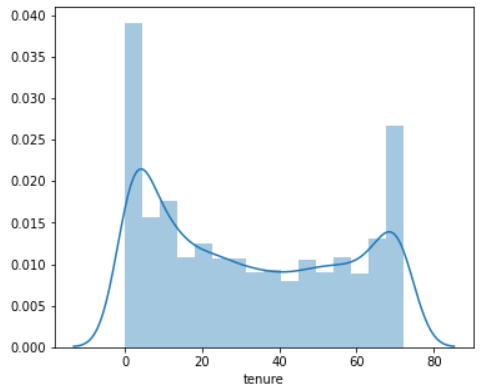
As I observed that no outlier was present in any of the above features. So I skipped the outlier removal step and proceeded for further analysis.

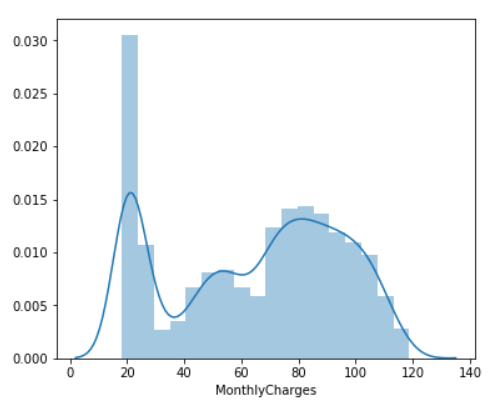
After that we used distribution plot to check the distribution of the data among the feature :

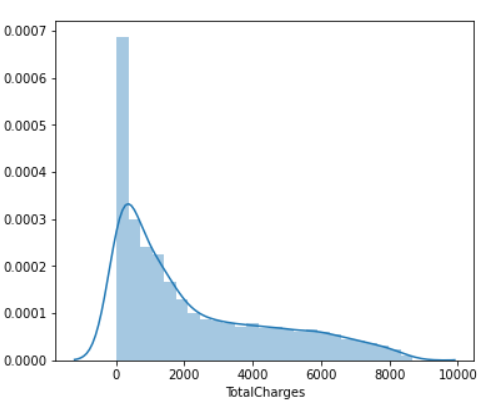
* I again used the following code to plot the distribution plot



The following output was obtained,







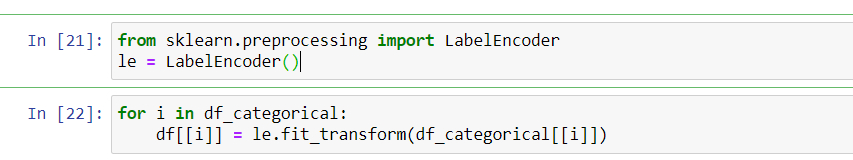
The following observations were made from the above distribution plot:

* The data present in the Total charges is highly right skewed.

1. **) Data Pre-processing:**

* **Label Encoder**

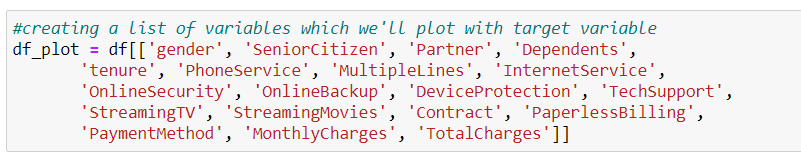
First, I used label encoder to convert the categorical features.

****

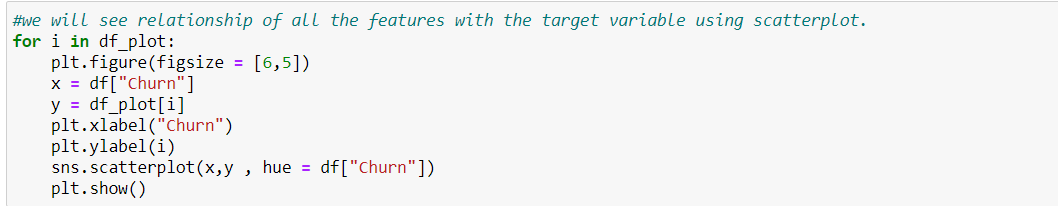
In the above code, I first imported label encoder from the sklearn.preprocessing library and then fit the categorical features(df\_categorical) into label encoder.

* After that I checked the relationship of the independent features with the dependent feature.

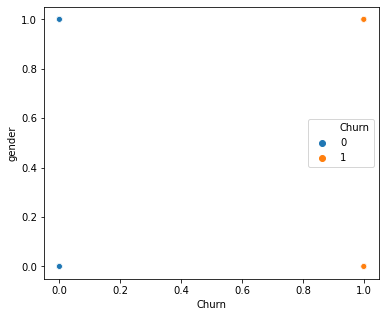
For that I first created a list of all the independent features



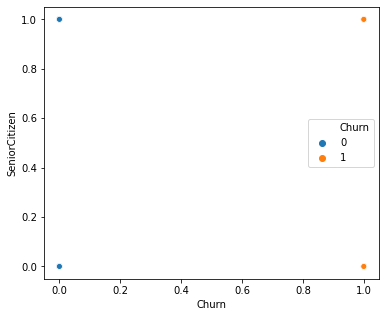
After that I used for loop to plot the scatterplot between the dependent and independent feature. And I got the following results.



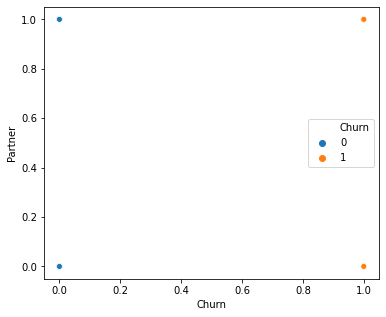
1.Gender



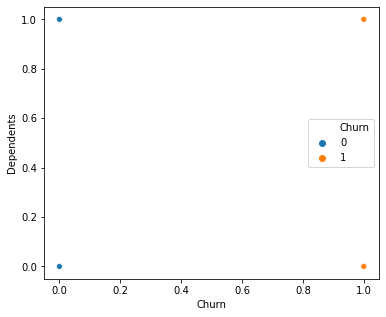
2.Senior citizen



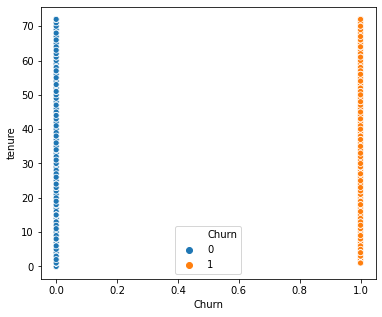
3.Partner



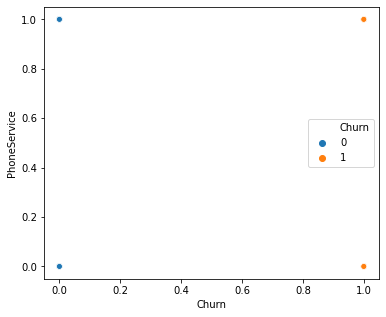
4.Dependets



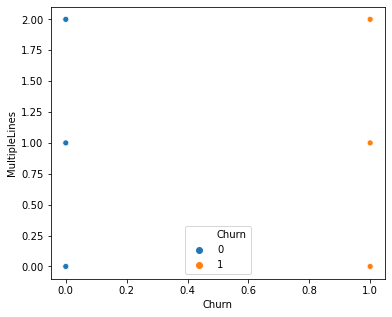
5.Tenure



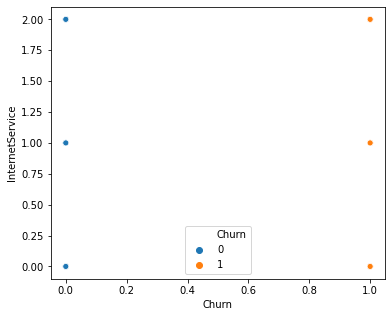
6.Phone Service



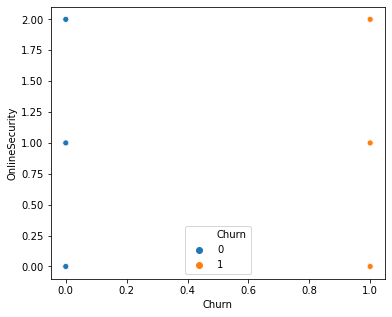
7.Multiple lines



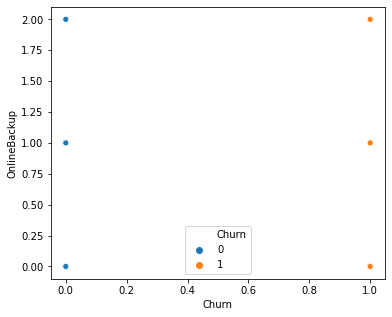
8.Internet Service



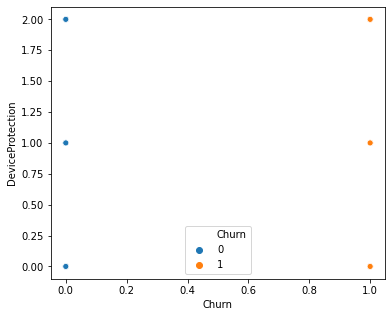
9.Online Security



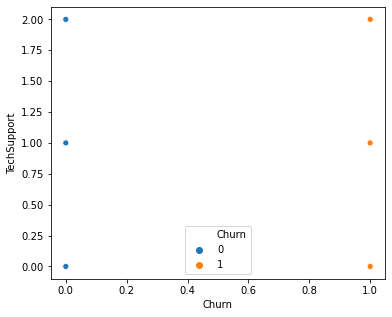
10.Online Backup



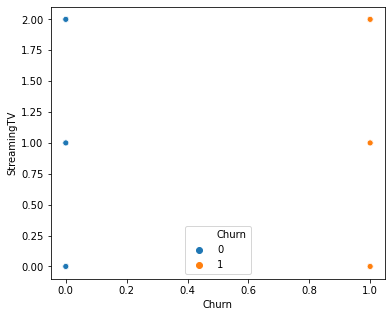
11.Device Protection



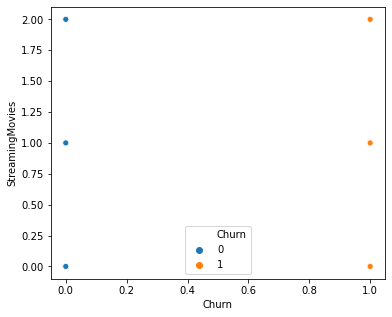
12.TechSupport



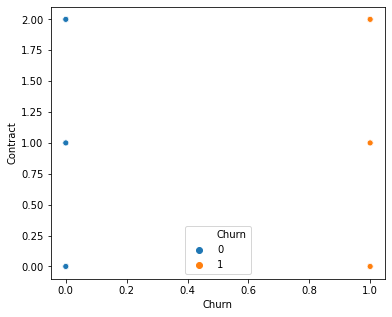
13.StreamingTV



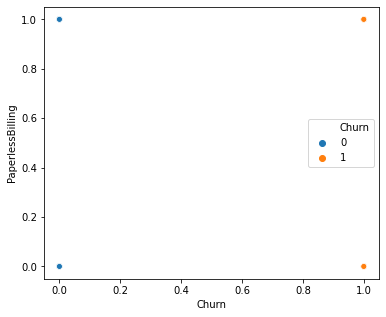
14.StreamingMovies



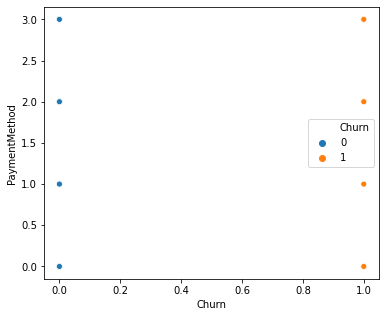
15.Contract



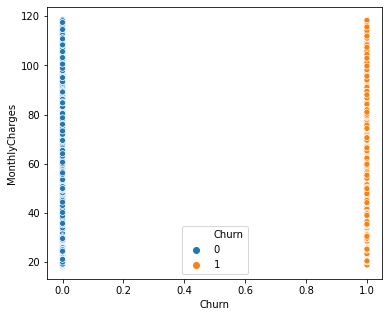
16.PaperlessBilling



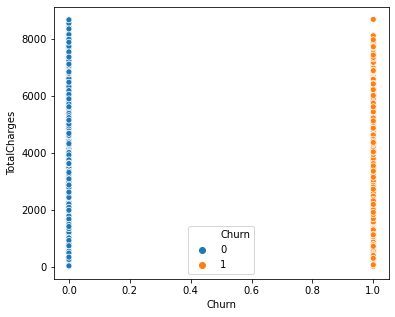
17.Payement Method



18.Monthly Charges



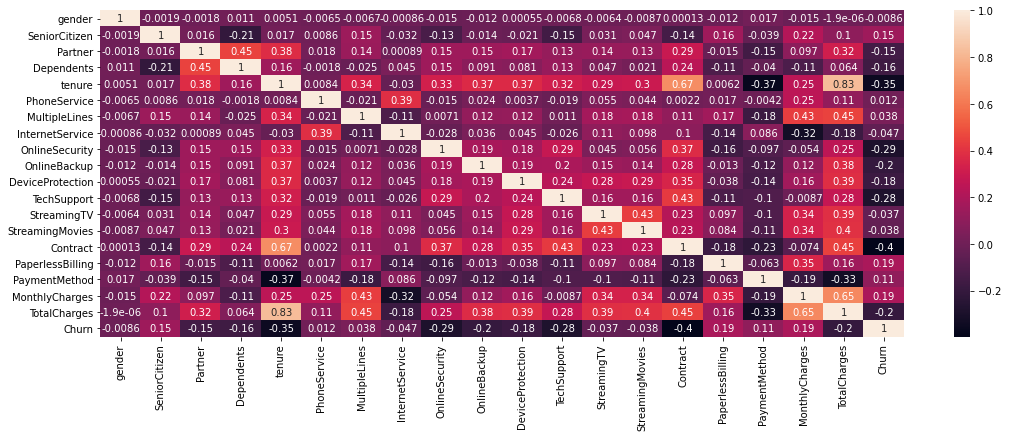
19.Total Charges



After plotting all the scatterplot between the independent and dependent variable, I didn’t get any insights or any trend that explains the behaviour of the customers who discontinued the services or those who are still using them.

You might have observed that I used few exploratory data analysis tools after the data pre-processing, so, I used correlation matrix to check the relationship between the dependent and independent variable.

* **Correlation**

****

* The following observations were made from the heatmap of correlation matrix:

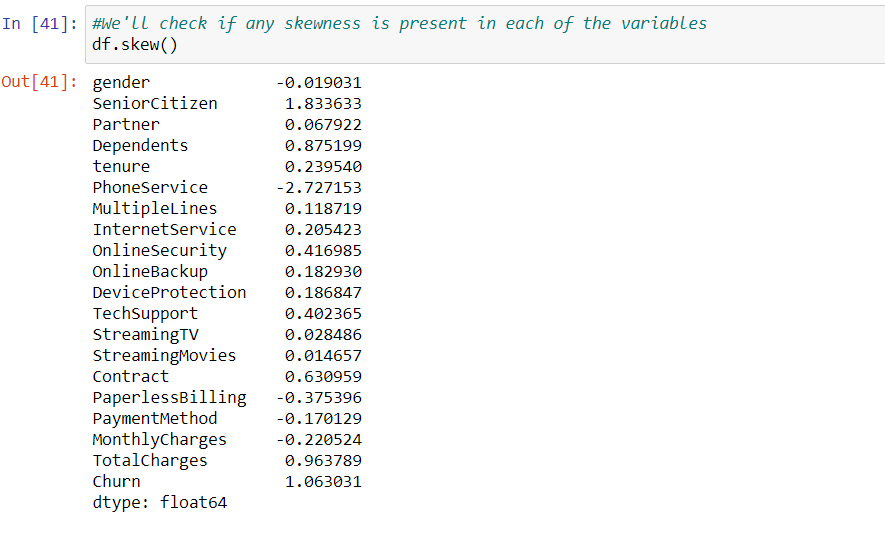
MonthlyCharges and Contracts are highly correlated with target feature "Churn" with corelation value (0.193356) and (-0.396713) respectively. Whereas gender is least correlated to the target feature.

* **Skewness Treatment**

**Checking for skewness and removing the skewness.**

After the getting useful insights from the EDA, I then checked the skewness present in the data set and removed it, because removing outliers and skewness is important for increasing the accuracy of the model.

To check the skewness I first used df.skew() to check the skewness present in the dataset.



We decided to keep the skewness between +/- 0.5. So, we checked the skewness in each of the feature.

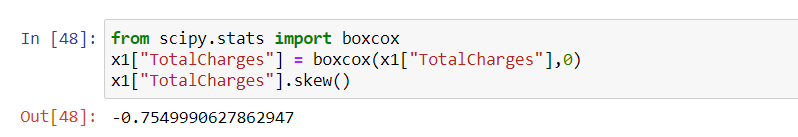
I observed that most of the skewness that is greater than 0.5 or less than -0.5 is present in the categorical features. So, we won’t be removing any skewness from these features.

Further I noticed that skewness is present in the TotalCharges feature. Since it contains continuous or say numeric data, so, I removed the skewness from this feature using the following skewness removing techniques :

1. I will split the data into two categories that is independent and dependent features and will store them in two variables X (independent features) and Y (Dependent Feature).

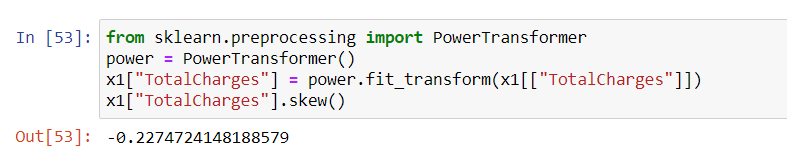


1. After that I first used boxcox function to remove the skewness. And I got the following output



I observed that the skewness is still present in the feature, So I tried removing it by using the same function but it threw an error that the data must be positive. Since we can apply boxcox function only on positive set of data so I used PowerTranformer to remove the further skewness.

1. Using PowerTranformer we got the following output.



As we can see that after using powertransformer, we are finally able to remove the skewness from the TotalCharges Feature. So now, we proceeded to our next step, that is, standardizing and scaling the whole dataset

* **Scaling the data**

To avoid the model from getting biased its important to scale our data. For scaling the dataset I used minmaxscaler() function. Remember that we only scale or numerical data so I used a for loop on the list(df\_continous) which I had already made in the beginning of the project.

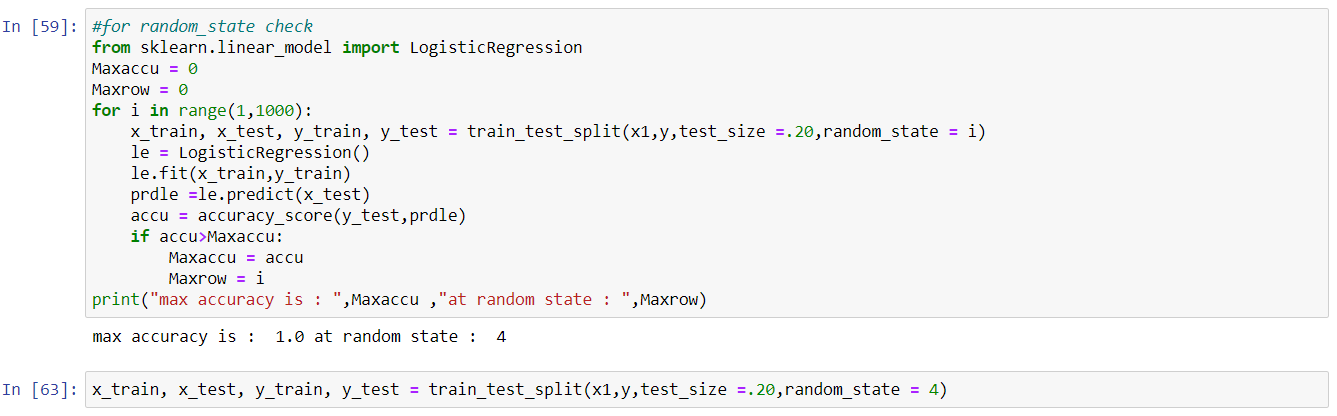


Once the scaling is done the next step is to start building our model.

**4.) Model Building**

Here I used different classification algorithms for building the model, Fitting the model and then predicting the outputs.

* So, the first step is to split the data into testing and training part. For that I used train\_test\_split method.



You can see that I used a few lines of code to get the random state on which our model is giving maximum accuracy.

* After splitting the data set,We used the following classification algorithms for model building,

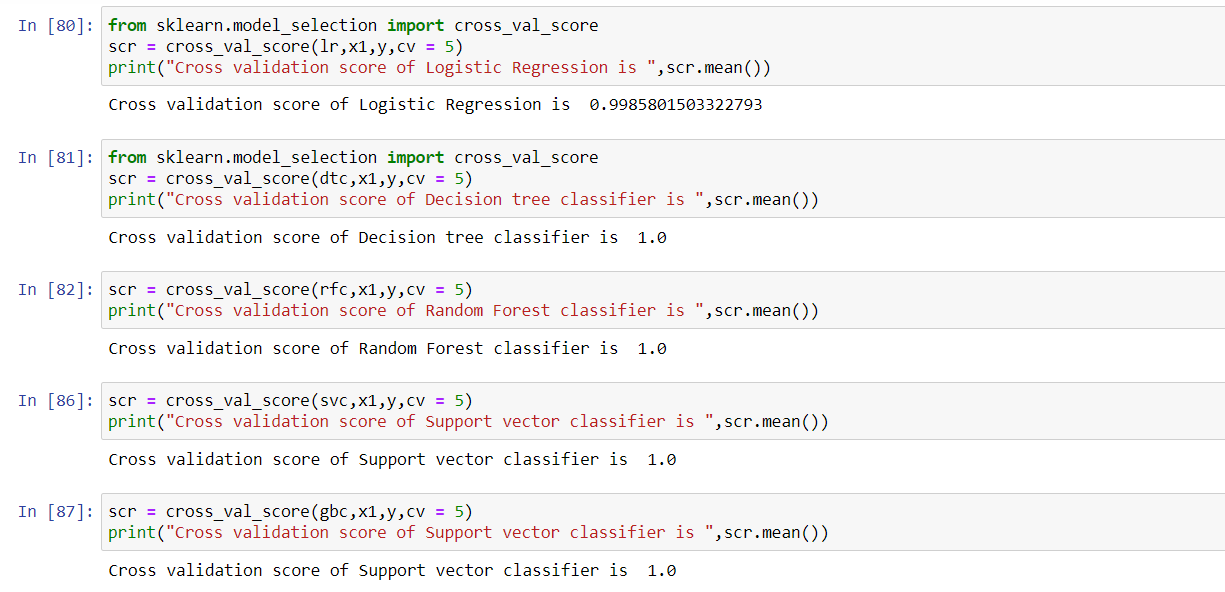
1. Logistic Regression.
2. Random forest classifier
3. Decision Tree classifier
4. Support Vector classifier
5. Gradient Boosting classifier

* After that we plotted AUC ROC curve of each of the algorithm to check whether the model is learning efficiently or not.

After checking the accuracy and auc roc curve of all the models I made this conclusion that, almost the models we tested for model building shows 100 percent accuracy. So, we'll use cross validation score to check which model is actually learning efficiently or not.

**5.) Cross – Validation Score**

Using cross\_val score function I checked the cross validation score of every model and got the following results.

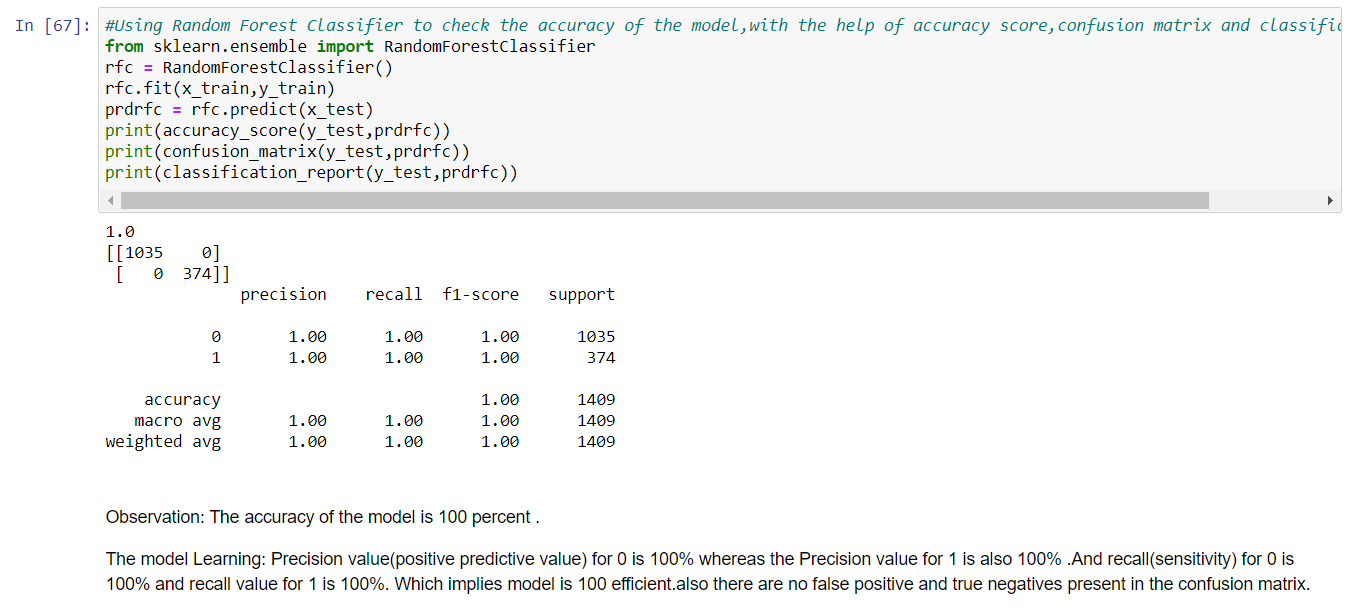


After looking at the cross validation score of each model the following conclusion was made,

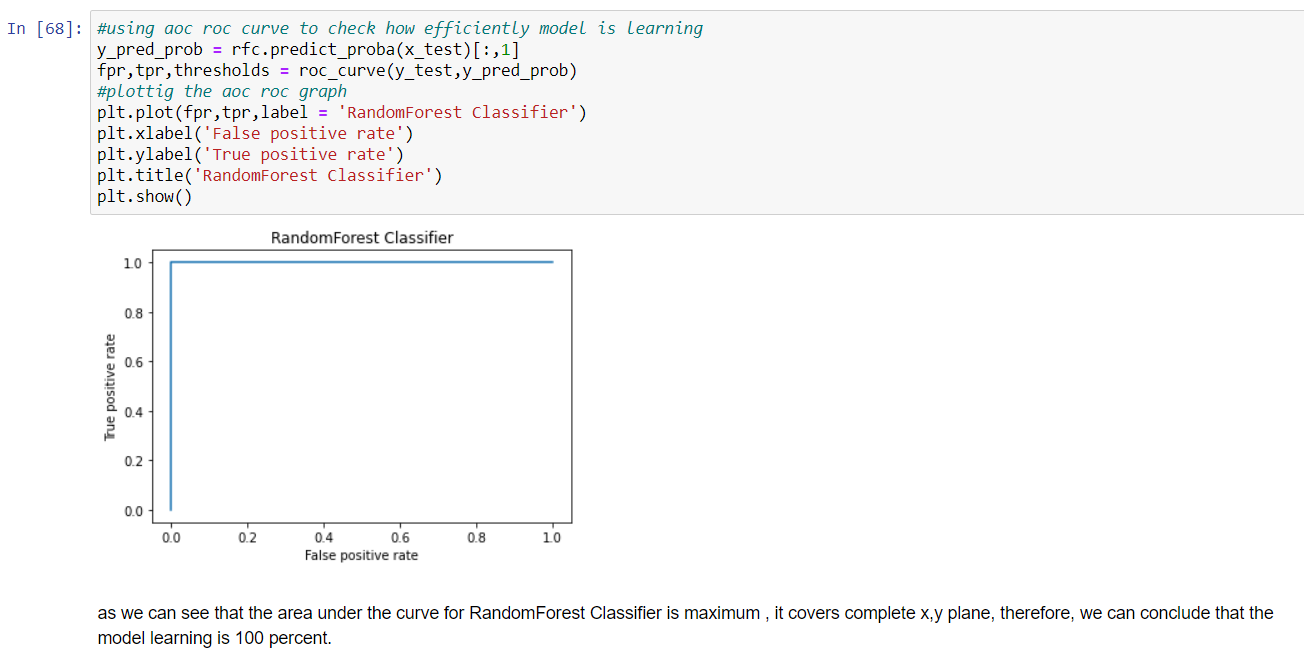
From the above cross validation scores of different models we conclude that Decision tree classifier , Random forest classifier and gradient bosting classifier shows same cross validation score. So, we can choose any of the model.

So, I chose Random Forest Classifier algorithm for the model building.

I’ll share the accuracy score and AUC ROC Curve of Random Forest classifier with you,



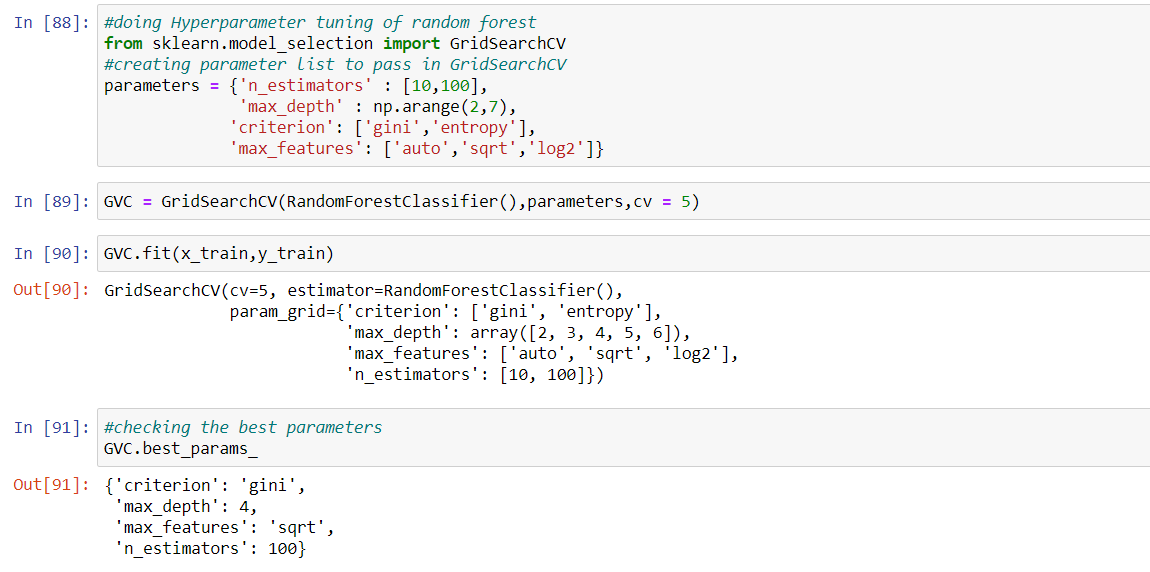
* AUC ROC CURVE OF RANDOM FOREST CLASSIFIER



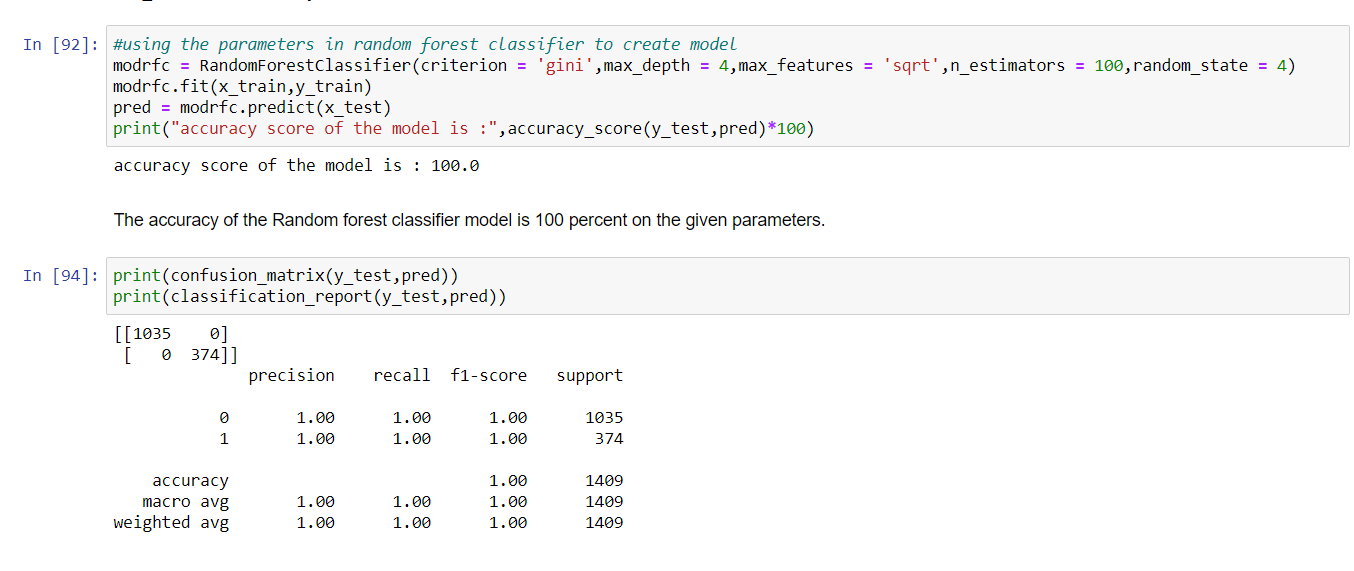
Now , we’ll proceed to the next step that is hyperparameter tuning of the model in order to further increase the accuracy of the model.

**6.) Hyperparameter Tuning**

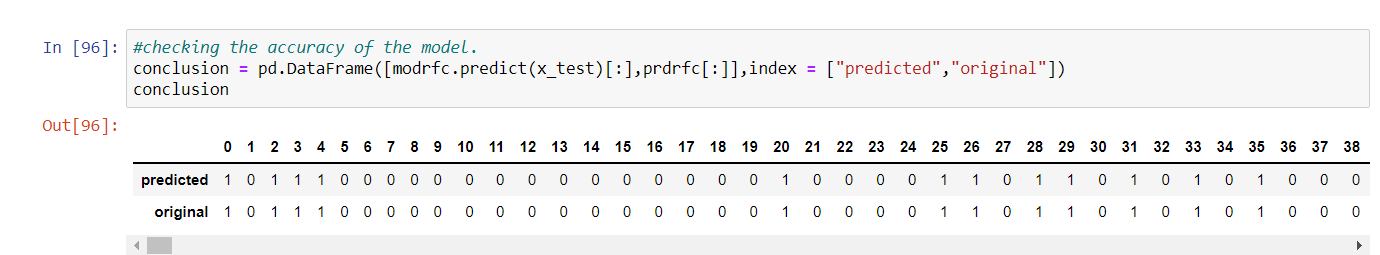
For hyperparameter tuning I created the list of parameters of the random forest classifier and tested the model on those parameters ,After that I took the best parameters from this list and used them to increase the efficiency of my model.



Creating the model using best parameters

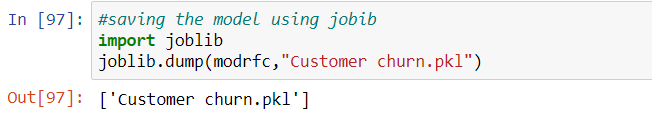


After successfully making the final model, I used some set of data to check whether the model actually working efficiently, making accurate prediction or not. For that I used following set of codes.



**7.) Saving the model**

To save the model I used joblib library .



**8.) Conclusion**

From here we concluded that the model is actually predicting efficiently and making perfect prediction on the given set of data. Further we saw that the accuracy of the model on the given set of data is 100 percent. This model can genuinely help the company in predicting whether the new customer will be associated with them, what type of services they should provide to them. Not only this, this model also help them in making better decisions and providing better services to their existing customer, so that they don’t switch to other company.