***Article on* :**

**Customer Churn Analysis**

* **Problem Definition:**

Here, we are having a sample dataset from IBM about the Customers of a telecom company, where we have to make prediction about the churn of the customers through modelling of the data.

As it is a very big problem for the companies that their customers leave their services and opt someone else’s service. So companies want to identify those customers well in advance and try to retain them for longer by identifying their needs and what they actually want from the company.

* **Data Analysis:**

In Data Analysis phase we have to do a lot of things. You can say that there are sub-phases of this phase that includes: Data Cleaning, Visualisation, etc.

* **Data Cleaning:**

In Data Cleaning, our first job is to have a look at data and find out its shape.

And by using **df.shape**, we came to know that the data is having 7043 rows and 21 columns.

With that, we also come to know about the Data Types of the columns. Where we can see clearly that there are – 1 float, 2 int, and 18 object columns.

It’s a matter of surprise that although the data type of some columns are object whereas they have to be int. So, it was a duty to change the data types of such columns.

After converting the data type of the columns we will ***drop*** some of the columns as they are not needed, e.g., CustomerID.

* **Using some Visualization:**

After cleaning the data we try to draw some information from the cleaned data set.

At first, we have plotted some Pie Charts and Count plots to show the **Churn** columns value counts that how many have churned from the service and how many have not.

From the figures which we have got, it was clear that there are about only one-fourth of the population which is churning but the rest are continuing with the service provider.

Besides that, we also have plotted a Histogram to show that how many customers may or may not leave in a particular Tenure which is being converted in Months.

From the figure it was evident that those who are with very less Tenure have almost equal chances to Churn. And as the Tenure increases the customers have less tendency to Churn out of the service.

After that we plotted a side-by-side Histogram again to see Customer Leaving and Not Leaving with respect to the “Monthly Charges”.

With less monthly charges there are very less number of churning instances of customers but as the monthly charges increases the instances of churn have also increased.

After that we have analysed ‘Contract Type Analysis’ which is of three kinds –

1. Month-to-Month contract
2. One-year contract
3. Two-year contract

And through this we saw that there is an extremely high chance of customer churn from ‘Month-to-Month contract’, almost 43%. After that ‘One-year contract’ in which there is almost 11% chance of customer churn.

And at last, customers with ‘Two-year contract’ in which only about 3% of customers churn.

After that we analyzed the customer churn with respect to the ‘Gender’ of the customers. And in this we didn’t saw any major difference on churn rate. Churn rate is almost equal in both the case.

Then we analysed the data for customer churn with respect to ‘Dependents’. And we found that customers with dependents have fewer tendencies to churn in comparison to Customers with no dependents.

* **EDA Concluding Remarks:**

Through the EDA we have seen that there are many factors for churn. Some factors contribute more some less. But we have to consider all the factors which have even the slightest of influence on the customer churn.

* **Pre-processing Pipeline:**

In this phase, the first thing which I have done is to create a function to look at the unique values of each column which is of object data type only.

When we run that code we get a detailed list of the columns and the unique values each of the column have.

We can see that some of the columns have 'No internet service' or 'No phone service', that can be replaced with a simple "No". So, we did that. And after that we get only two kind of unique values i.e., either ‘Yes’ or ‘No’.

After doing that we replace the unique values which is in the form of ‘Yes’ and ‘No’ to 1 and 0.

For which we have created a variable “yes-no-columns” and stored all the variable names in it.

So, now we have replaced all the ‘Yes’, ‘No’ values with 1 and 0. And we did same for the

‘Gender’ column where we replace Male from 1 and Female from 0.

And now for the remaining columns which still have the ‘object’ values we have used the “One Hot Encoding Scheme. We know that using this scheme it creates different feature variables for the unique values it contains.

As in the case of ‘Contract’ type in which we have three unique values ‘Month-to-Month’ , ‘One Year’, ‘Two Year’.

And now if we check the DataFrame we find that all the columns are with numeric format and in range 0 to 1. But there are some columns which is not following this. For that we have used “sklearn” library MinMaxScaler. After using that we have all the values of those columns also in the range of 0 to 1.

After doing all this we **split** the data in **Train and Test**. And for that we use the “sklearn’s” **Train-Test-Split.**

* **Building ML Models:**

Here we have built three models, namely – Logistic Regression, Random Forest, and Support Vector Machine.

All the three models have almost performed equally just the Logistic Regression Model performed slightly well with 80% of value.

* **Concluding Remarks**:

On the conclusion we can say that whatever the ML models we have made and applied on the data they all have performed equally well except Logistic Regression which have done slightly better than the rest two models.