Telco Customer Churn prediction

Using Machine Learning

Work Flow

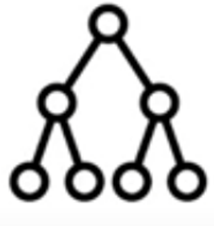
Define Objectives

Data Collection

EDA

Data Preprocessing

Data Preparation



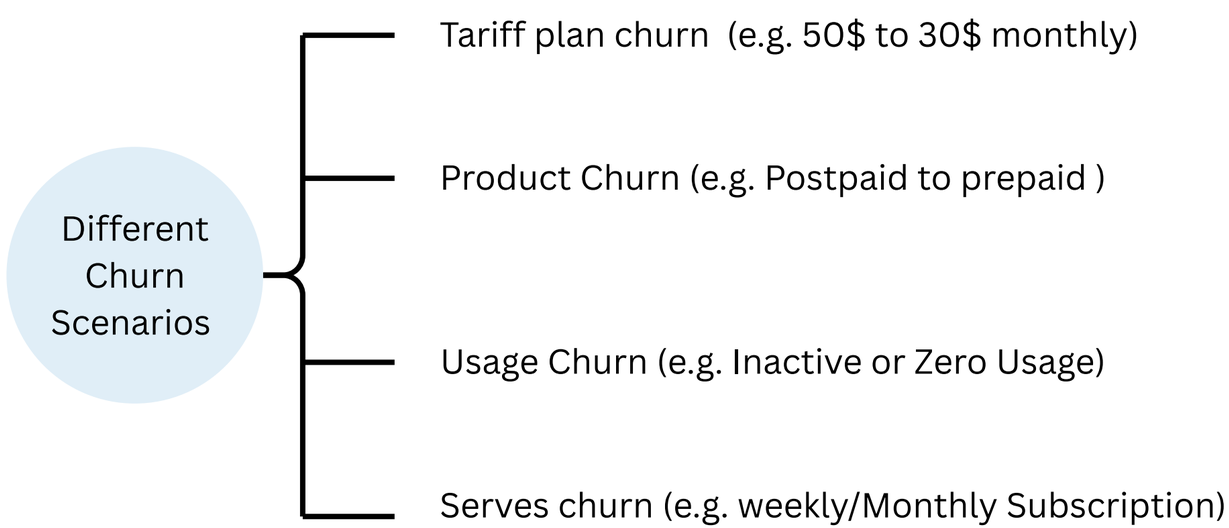
model Building

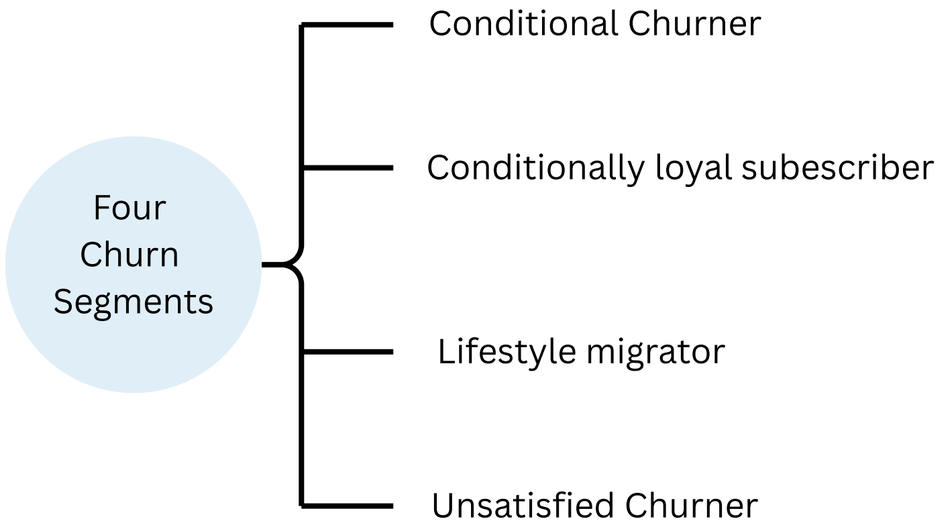
model Evaluation

Step 1:Define Objects

Why we need to manage churn ?

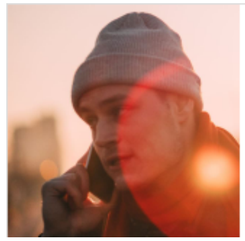
- churn is a key driver of EBITDA margin and an industry-wide challenge .
- A churn customer provides less revenue or zero revenue and increases competitor market share .
- Increase acquisition cost for the service provider if the customer churned to competition. It costs up to 5 times as much for an service Provider to acquire a new subscriber as to retain an existing one.





our objective is to use Machine Learning model to predict customer churn , It helps Serves provider companies to take appropriate actions

Step 2:Data Collection



Telco Customer Churn

Focused customer retention programs

kaggle.com

```
[185] telco_data.describe()
```

	SeniorCitizen	tenure	MonthlyCharges
count	7043.000000	7043.000000	7043.000000
mean	0.162147	32.371149	64.761692
std	0.368612	24.559481	30.090047
min	0.000000	0.000000	18.250000
25%	0.000000	9.000000	35.500000
50%	0.000000	29.000000	70.350000
75%	0.000000	55.000000	89.850000
max	1.000000	72.000000	118.750000

SeniorCitizen is actually a categorical hence the 25%-50%-75% distribution is not proper

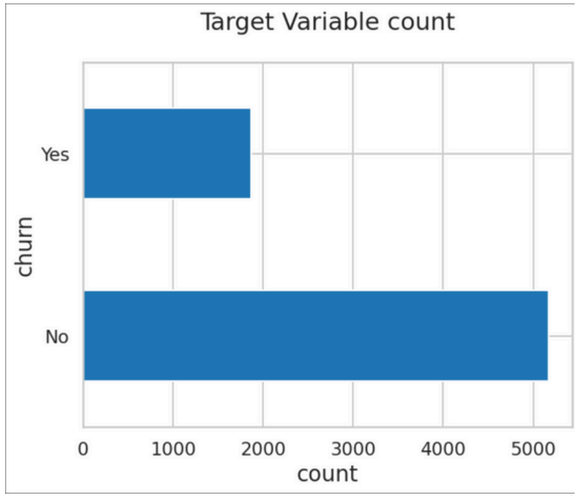
Step 3:Data Preparation :

Two Types of variables :

- Numerical
- categorical

we take Type in considerations when we analyze the data

No 73.463013
Yes 26.536987



- Data is highly imbalanced, ratio = 73:27
- the model may be overfitted and has low accuracy
- Hendel it using oversampling or down sampling

Step 3:Data Preparation :

Data Cleaning

Missing Data - Initial Intuition

- Here, we don't have any missing data.

General Rules for Handle Missing Values:

- Delete Rows/Columns : This method we commonly used to handle missing values. Rows can be deleted if it has insignificant number of missing value Columns can be delete if it has more than 75% of missing value.
- Replacing with mean/median/mode: This method can be used on independent variable when it has numerical variables. On categorical feature we apply mode method to fill the missing value.

As there's no thumb rule on what criteria do we delete the columns with high number of missing values, but generally you can delete the columns, if you have more than 30-40% of missing values. But again there's a catch here, for example, Is_Car & Car_Type, People having no cars, will obviously have Car_Type as NaN (null), but that doesn't make this column useless, so decisions has to be taken wisely.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 21 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   customerID        7043 non-null   object 
 1   gender            7043 non-null   object 
 2   SeniorCitizen     7043 non-null   int64  
 3   Partner           7043 non-null   object 
 4   Dependents        7043 non-null   object 
 5   tenure            7043 non-null   int64  
 6   PhoneService      7043 non-null   object 
 7   MultipleLines     7043 non-null   object 
 8   InternetService   7043 non-null   object 
 9   OnlineSecurity    7043 non-null   object 
 10  OnlineBackup      7043 non-null   object 
 11  DeviceProtection  7043 non-null   object 
 12  TechSupport       7043 non-null   object 
 13  StreamingTV       7043 non-null   object 
 14  StreamingMovies   7043 non-null   object 
 15  Contract          7043 non-null   object 
 16  PaperlessBilling  7043 non-null   object 
 17  PaymentMethod     7043 non-null   object 
 18  MonthlyCharges    7043 non-null   float64
 19  TotalCharges      7043 non-null   object 
 20  Churn             7043 non-null   object 
dtypes: float64(1), int64(2), object(18)
memory usage: 1.1+ MB
```

Step 3:Data Preparation :

Data Cleaning

Convert incorrect data types

Since the % of these records compared to total dataset is very low ie 0.15%, it is safe to ignore them from further processing.

So to clean data we drop all nulls

```
[203] telco_data_copy['TotalCharges']=pd.to_numeric(telco_data_copy['TotalCharges'],errors='coerce')
```

```
telco_data_copy.isnull().sum()
```

	0
customerID	0
gender	0
SeniorCitizen	0
Partner	0
Dependents	0
tenure	0
PhoneService	0
MultipleLines	0
InternetService	0
OnlineSecurity	0
OnlineBackup	0
DeviceProtection	0
TechSupport	0
StreamingTV	0
StreamingMovies	0
Contract	0
PaperlessBilling	0
PaymentMethod	0
MonthlyCharges	0
TotalCharges	11

Step 3:Data Preparation :

Derived Metrics

Derived metrics create a new variable from the existing variable to get a insightful information from the data by analyzing the data.

- Feature Binning
- Feature Encoding

Feature Binning: converts or transform continuous/numeric variable to categorical variable. It can also be used to identify missing values or outliers.

```
#tenure Feature Binning
labels=['{0}-{1}'.format(i,i+11)for i in range(1,72,12)]
telco_data_copy['tenure_groups']=pd.cut(telco_data_copy['tenure'],range(1,80,12),right=False,labels=labels)
telco_data_copy['tenure_groups'].value_counts()
```

tenure_groups	count
1-12	2175
61-72	1407
13-24	1024
25-36	832
49-60	832
37-48	762

Equal Width Equal width separate the continuous variable to several categories having same range of width.

Step 3:Data Preparation :

Derived Metrics

Derived metrics create a new variable from the existing variable to get a insightful information from the data by analyzing the data.

- Feature Binning
- Feature Encoding

Feature Encoding: help us to transform categorical data into numeric data.

```
telco_data_copy['Churn'] = np.where(telco_data_copy.Churn == 'Yes',1,0)
telco_data_copy.head()
```

Label encoding Label encoding is technique to transform categorical variables into numerical variables by assigning a numerical value to each of the categories.

```
[211] # One-Hot Encoding for catogoral data
      telco_data_dummies = pd.get_dummies(telco_data_copy,dtype=int)
      telco_data_dummies.head()
```

One-Hot encoding This technique is used when independent variables are nominal. It creates k different columns each for a category and replaces one column with 1 rest of the columns is 0. Here, 0 represents the absence,

Step 3:Data Preparation :

Exploratory Data Analysis

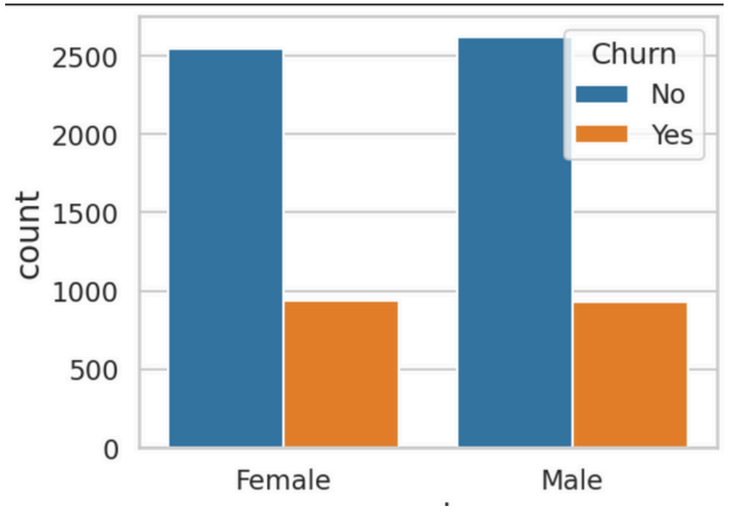
Univariate Analysis:

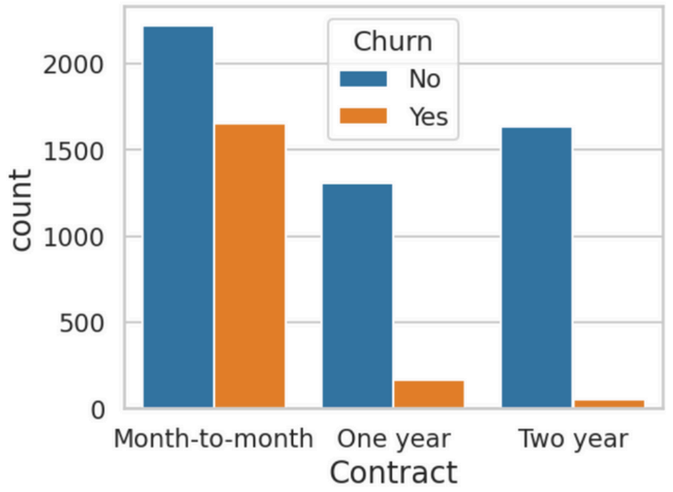
- deal with analyzing one feature at a time.
- The main purpose of univariate analysis is to make data easier to interpret.
- Mainly, Histogram is used to visualize univariate.

Frequency Measure: Measure of frequency include frequency table, where you tabulate how often a particular category or particular value appear in the dataset.

Insight: Gender is not an affecting variable alone , but Contract is

Categorical Data





Step 3:Data Preparation :

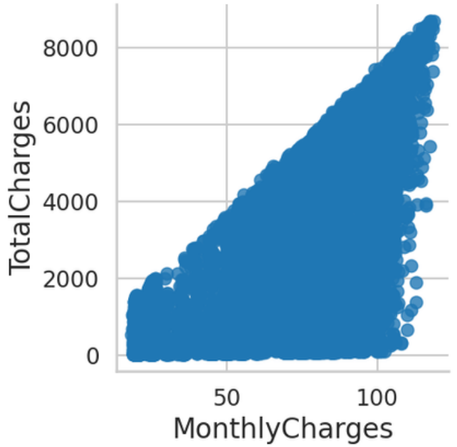
Exploratory Data Analysis

Univariate Analysis:

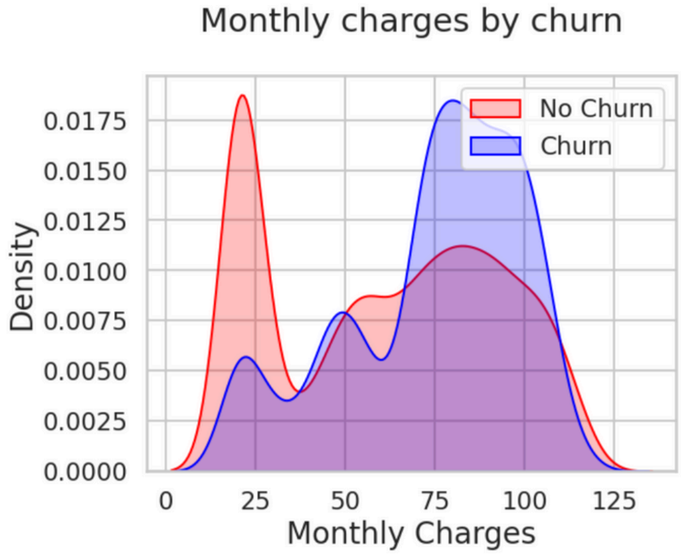
- deal with analyzing one feature at a time.
- The main purpose of univariate analysis is to make data easier to interpret.
- Mainly, Histogram is used to visualize univariate.

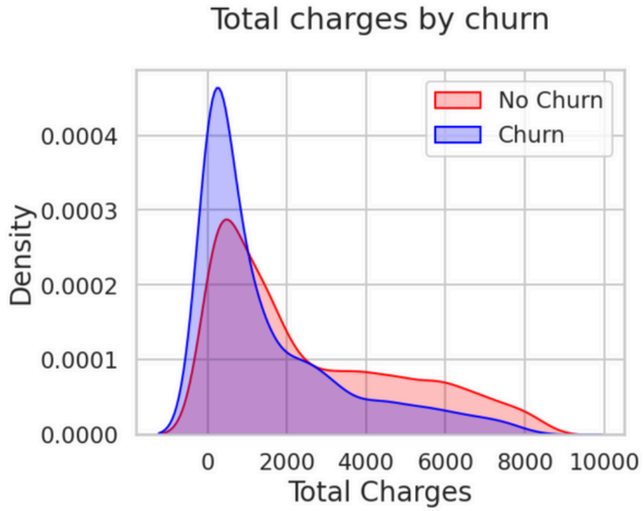
Insight:

- Churn is high when Monthly Charges are high
- as higher Churn at lower Total Charges
- Higher Monthly Charge and Lower Total Charge are linked to High Churn.



Continuous data





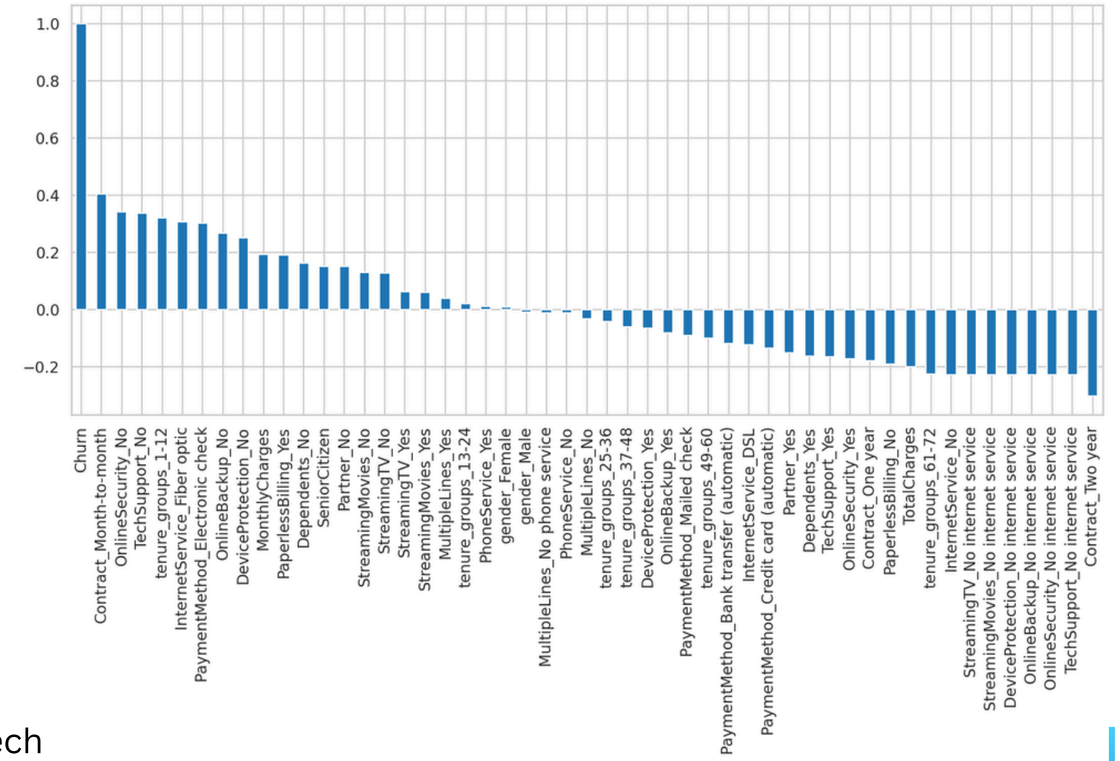
Step 3:Data Preparation :

Exploratory Data Analysis

Bivariate Analysis: Data which has two variables ,you often want to measure the relationship that exists between these two variables.

Correlation: Correlation measure the strength as well as the direction of the linear relationship between the two variables. Its range is from -1 to +1.

- If one increases as the other increases, the correlation is positive
- If one decreases as the other increases, the correlation is negative
- If one stays constant as the other varies, the correlation is zero



Insight:

- HIGH Churn seen in case of Month to month contracts, No online security, No Tech support, First year of subscription and Fiber Optics Internet.
- LOW Churn is seen in case of Long term contracts, Subscriptions without internet service.
- Factors like Gender, Availability of PhoneService and # of multiple lines have almost NO impact on Churn.

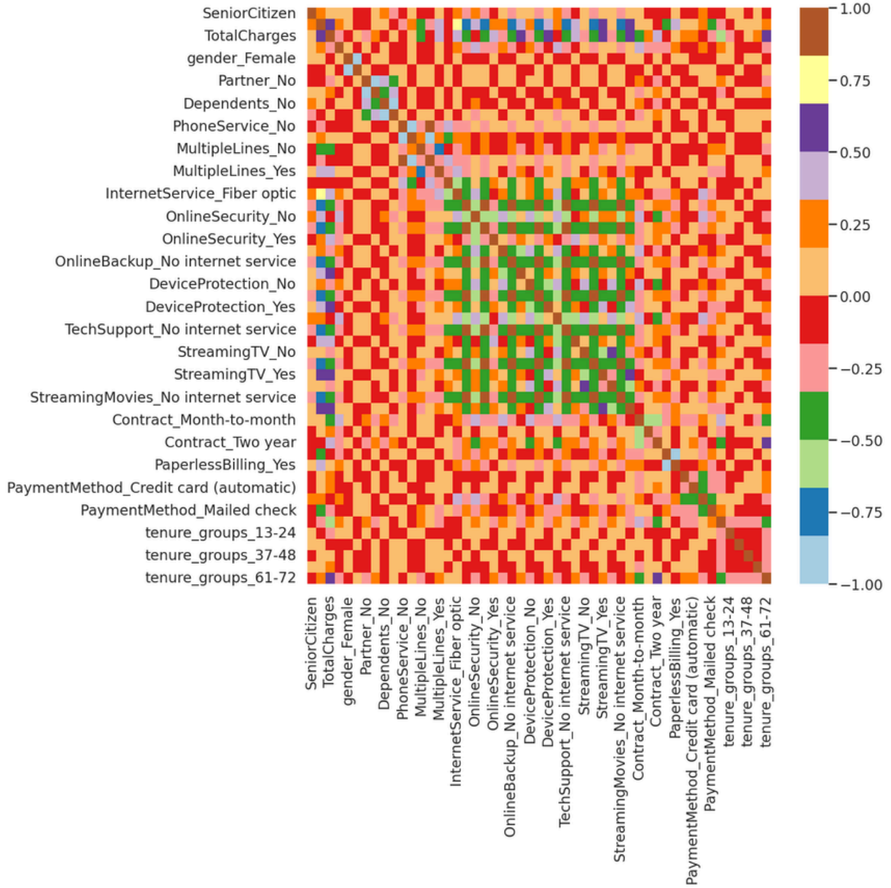
Step 3:Data Preparation :

Exploratory Data Analysis

Bivariate Analysis: Data which has two variables ,you often want to measure the relationship that exists between these two variables.

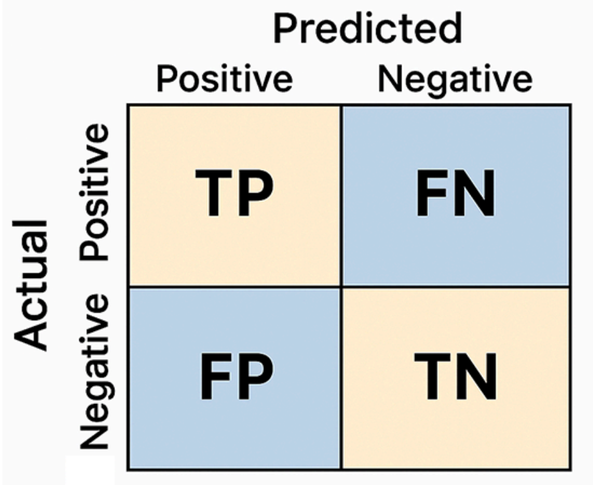
Correlation: Correlation measure the strength as well as the direction of the linear relationship between the two variables. Its range is from -1 to +1.

- If one increases as the other increases, the correlation is positive
- If one decreases as the other increases, the correlation is negative
- If one stays constant as the other varies, the correlation is zero



Step 4:Model Building

1.Create Feature X (All columns except Churn)& Label Y (Churn which is our target)
2.split data into 80% training and 20% testing
3.choose classifier



Step 5:Model Evaluation

Evaluation metrics are used to measure the quality of the statistical or machine learning model.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$Sensitivity = \frac{TP}{TP + FN}$$

$$F1\ score = 2\ x\ \frac{Precision * Recall}{Precision + Recall}$$

DecisionTreeClassifier

Decision Trees (DTs) are a supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

	precision	recall	f1-score	support
0	0.82	0.91	0.86	1026
1	0.65	0.46	0.54	381
accuracy			0.79	1407
macro avg	0.74	0.69	0.70	1407
weighted avg	0.78	0.79	0.78	1407

As you can see that the accuracy is quite low, and as it's an imbalanced dataset, we shouldn't consider Accuracy as our metrics to measure the model, as Accuracy is cursed in imbalanced datasets

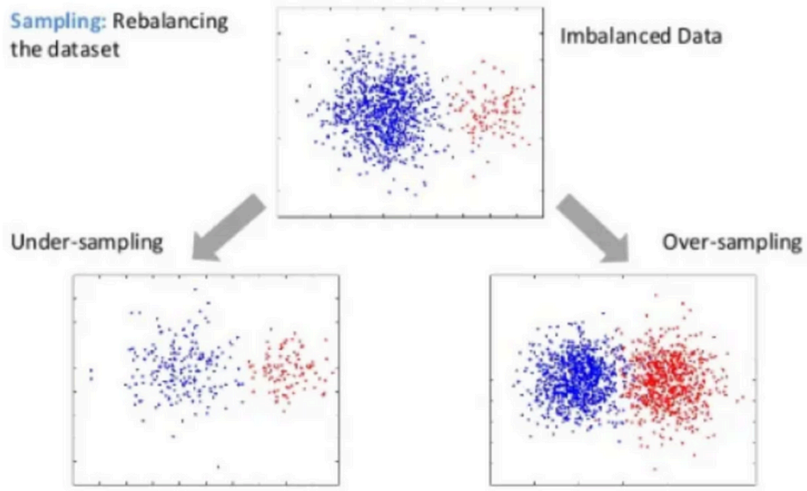
.

Hence, we need to check recall, precision & f1 score for the minority class, and it's quite evident that the precision, recall & f1 score is too low for Class 1, i.e. churned customers.

DecisionTreeClassifier

Resampling:

- Upsampling which consists in over-sizing the minority class by adding observations
- Downsampling which consists in down-sizing the majority class by removing observations until the dataset is balanced




```
0.9356521739130435
              precision    recall  f1-score   support

           0       0.94      0.92      0.93       525
           1       0.93      0.95      0.94       625

    accuracy                           0.94      1150
   macro avg       0.94      0.93      0.94      1150
weighted avg       0.94      0.94      0.94      1150
```

Now we can see quite better results, i.e. Accuracy: 94 %, and a very good recall, precision & f1 score for minority class.

Random Forest Classifier

is a method that combines the predictions of multiple decision trees to produce a more accurate and stable result.
It can be used for both classification and regression tasks.

	precision	recall	f1-score	support
0	0.82	0.92	0.87	1026
1	0.69	0.47	0.56	381
accuracy			0.80	1407
macro avg	0.76	0.70	0.71	1407
weighted avg	0.79	0.80	0.79	1407

Before Upsample

	precision	recall	f1-score	support
0	0.96	0.93	0.94	541
1	0.94	0.97	0.95	633
accuracy			0.95	1174
macro avg	0.95	0.95	0.95	1174
weighted avg	0.95	0.95	0.95	1174

After Upsample

With RF Classifier, also we are able to get quite good results, infact better than Decision Tree.

Step 4:Model Deployment :

```
[1]
Prediction: Churn
Prediciton Probability: [[0.11093977 0.88906023]]
```

THANK YOU