Plan:

I. Introduction
II. Business Understanding
1) Determine business objectives
2) Assess situation
3) Determine data mining goals
4) Project plan
III. Data Understanding
1) Initial data collection
2) Data description
3) Data Exploration
4) Data quality
IV. Data preparation
1) Data cleaning
2) Feature selection
3) Data transforming
a.Feature encoding
b.Feature scaling
c.Data augmentation
4) Dimensionality reduction
V. Modeling
1) Selection of modeling techniques
2) Generation of test designs
3) Building models
4) Assessing models
VI. Evaluation
1) Results evaluation
2) Process review
3) Determining next step
VII. Deployment
VIII. Conclusion

I. Introduction :

Customer churn is the percentage of customers that stopped using a certain company's product or service or even switched to the adversary company during a certain time frame. This defines an important issue in the telecommunications industry. Since the client is the greatest asset that a company can have, therefore keeping an already established customer compared to trying to find a new client is more beneficial to a company, regarding the financial aspect especially. In this project, we're trying to find the best solution to customer churn using the latest machine learning technologies for a better and a more accurate result using a near perfect model referring to customer analysis behavior. As our project plan, we applied The **CR**oss **I**ndustry **S**tandard **P**rocess for **D**ata **M**ining (*CRISP-DM*) method; a process model with six phases that naturally describes the data science life cycle.

II. Business understanding:

Nowadays, with mobile telecommunication being the most relevant way of communication across the globe, Telecommunication companies find themselves facing a huge problem of market saturation which makes finding a new customer a much harder and costly task than keeping returning customers. This issue made the rivalry between companies intense and advanced due to the latest technologies used to predict accurate models of customer behavior and target the weak links in a company's client base. The investment in customer churn issue and taking a close look into customer behavior is a prior necessity in the telecommunication industry as it could end up affecting the revenue numbers and influence policy decisions by the impact that could one customer apply to another to quit as well as adding extra costs especially in advertisement when keeping old subscribers is already cheaper.

1. Determine business objectives:

Our priority is making our business the most effective in the marketplace.

We reached out many solutions as compared and analyzed other company's customer churn and extracted insights in order to find better solutions and develop a more effective model. Being able to learn about the customer's behavior faster than the competitor is our highest goal ,

We will be analysing the common and obvious customer tendency that would lead to the churn and predicting the kind of subscribers who are more likely to switch the operator or to definitely quit and this by using machine learning tools . We are aiming to develop a model that predicts the customers with higher probability to switch lines.

By gaining information about customers from individual demographics to details of usage of service

, we'd be able to recognize which services are less likely to be consumed which lead to implementing effective retention strategies by identifying the services that would fulfill customer's preferences. As a result, we're looking forward to attempting a growth rate higher than churn rate and decreasing more than 2% churn which is equivalent to 10% reduction in costs .

2. Assess situation:

We, Belkis Baccar , Donia Ksia, Fadhel Shel, Issam Ben Moussa, Latyfa Sassi and Mohamed Khalil Chakroun, are data science students aiming to do churn analysis to bring out solutions and identify the customers that will eventually quit or switch companies.

First of all, we will start with the data preparation to a better understanding of the data using the 'Teleco-customer-churn' database, followed by applying classification methods such as K-Nearest Neighbors, Tree classifier and logistic regression referring to cross validation for best practices. We will use Python as the program language in the platform anaconda Jupyter Labs utilizing the Pandas , scikit.learn, Matplot.lib, Numpy and Seaborn libraries .

This project will normally be achieved by the 6th of January, we will deliver a report of [20-30] pages and a notebook well commented and detailed.The dataset we are working with contains at least 2129 rows, 33 columns and 1Mo for its size. In the next step, data understanding, we will be cleaning the database for an accurate model.

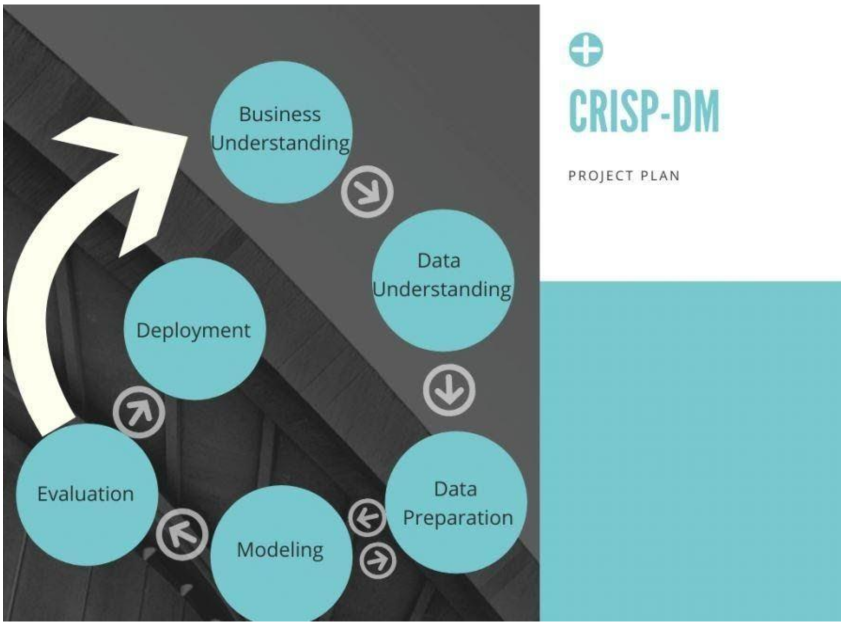
When it comes to the risks of this project, the dataset understanding phase can be difficult to do perfectly at the first try giving the different types of columns we have to work with.

3. Determine data mining goals:

When it comes to the data mining goals, we will be comparing models based on the F1 score to resolve the classification issues in order to reach the most accurate and effective method and model giving us the perfect solution for the customer churn problem. We are aiming for a F1 score in the interval [0.80..1].

$$F_\beta = (1+\beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}.$$

4. Project plan:



III. Data Understanding :

The *Telco customer churn* data contains information about a fictional telco company that provided home phone and Internet services to 7043 customers in California in Q3. It indicates which customers have left, stayed, or signed up for their service. Multiple important demographics are included for each customer, as well as a Satisfaction Score, Churn Score, and Customer Lifetime Value (CLTV) index.

1. Initial data collection

The dataset we are studying in this project is *Telco customer churn*. This dataset is published online with public access for learning purposes.

2. Data description

The format of the dataset we are putting in use is Comma Separated Values (CSV). It contains 7043 rows and 33 columns putting it at a size of 1.66 M. The column types vary from 10 numerical columns to 23 String values. The dataset can be divided to 4 sections:

- Demographics
- Location
- Services
- Status

Demographics

CustomerID(String): A unique ID that identifies each customer.

Count(Int64): A value used in reporting/dashboarding to sum up the number of customers in a filtered set.

Gender(String: Categorical): The customer’s gender: Male, Female

Age(Int64): The customer’s current age, in years, at the time the fiscal quarter ended.

Senior Citizen(String: Categorical) : Indicates if the customer is 65 or older: Yes, No

Partner(String: Categorical): Indicates the customer has a partner: Yes, No.

Dependents(String: Categorical) : Indicates if the customer lives with any dependents: Yes, No. Dependents could be children, parents, grandparents, etc.

Location Features:

Country(String): The country of the customer’s primary residence. Unique value: United States

State(String): The state of the customer’s primary residence. Unique value: California

City(String: Categorical): The city of the customer’s primary residence. 1129 different values

Zip Code(String: Categorical): The zip code of the customer’s primary residence. 1652 different values

Lat Long(String): The combined latitude and longitude of the customer’s primary residence.

Latitude(Float64): The latitude of the customer’s primary residence.

Longitude(Float64): The longitude of the customer’s primary residence.

Services Features :

Tenure in Months(int64): Indicates the total amount of months that the customer has been with the company by the end of the quarter specified above.

Phone Service(String: Categorical): Indicates if the customer subscribes to home phone service with the company: Yes, No

Multiple Lines(String: Categorical): Indicates if the customer subscribes to multiple telephone lines with the company: Yes, No, No phone service

Internet Service(String: Categorical): Indicates if the customer subscribes to Internet service with the company: No, DSL, Fiber Optic.

Online Security(String: Categorical): Indicates if the customer subscribes to an additional online security service provided by the company: Yes, No, No Internet service.

Online Backup(String: Categorical): Indicates if the customer subscribes to an additional online backup service provided by the company: Yes, No, No Internet service.

Device Protection Plan(String: Categorical): Indicates if the customer subscribes to an additional device protection plan for their Internet equipment provided by the company: Yes, No, No Internet service

Streaming TV(String: Categorical): Indicates if the customer uses their Internet service to stream television programing from a third party provider: Yes, No, No Internet service.

Streaming Movies(String: Categorical): Indicates if the customer uses their Internet service to stream movies from a third party provider: Yes, No, No Internet service.

Contract(String: Categorical): Indicates the customer's current contract type: Month-to-Month, One Year, Two Year.

Paperless Billing(String: Categorical): Indicates if the customer has chosen paperless billing: Yes, No

Payment Method(String: Categorical): Indicates how the customer pays their bill: Bank transfer(Automatic),Credit Card (Automatic), Mailed Check, Electronic Check.

Monthly Charge(Float64): Indicates the customer's current total monthly charge for all their services from the company.

Total Charges(String): Indicates the customer's total charges, calculated to the end of the quarter specified above.

Status Features:

Churn Label(String: Categorical): Yes = the customer left the company this quarter. No = the customer remained with the company. Directly related to Churn Value.

Churn Value(Int64): 1 = the customer left the company this quarter. 0 = the customer remained with the company. Directly related to Churn Label.

Churn Score(Int64): A value from 0-100 that is calculated using the predictive tool IBM SPSS Modeler. The model incorporates multiple factors known to cause churn. The higher the score, the more likely the customer will churn.

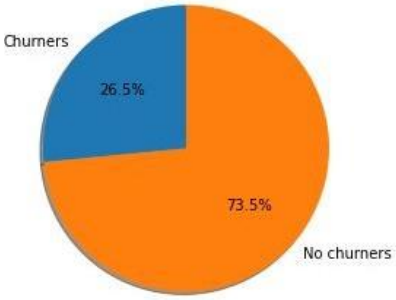
CLTV(Int64): Customer Lifetime Value. A predicted CLTV is calculated using corporate formulas and existing data. The higher the value, the more valuable the customer. High value customers should be monitored for churn.

Churn Reason(String): A customer's specific reason for leaving the company. Directly related to the Churn Category.

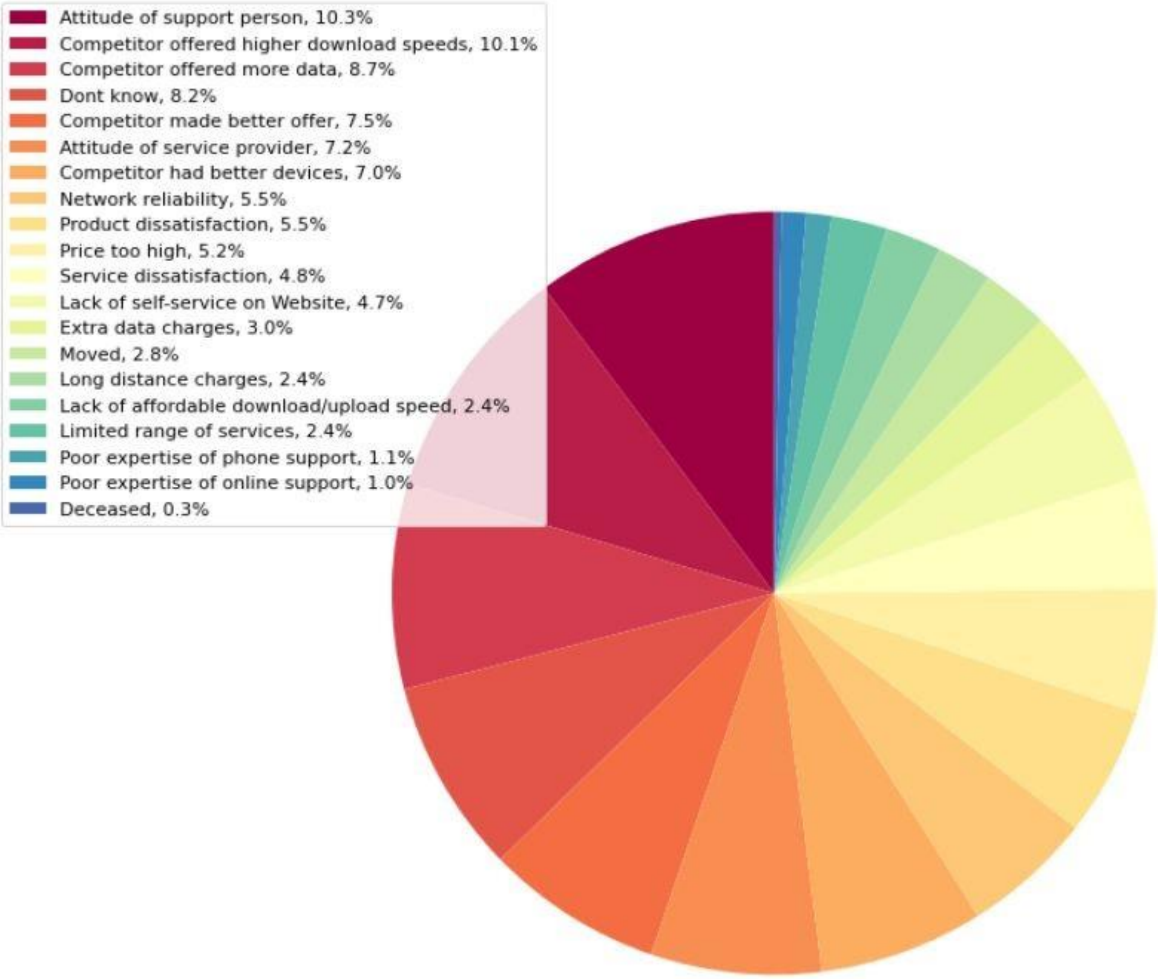
3. Data Exploration

In this part, we explored the dataset categorical features and these are the observations and plots we detected and created.

Churners : from 7043 clients, only 1869 are churners.



Churn Reason:

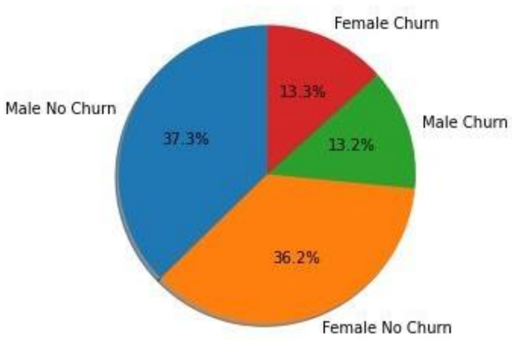


→ The attitude towards clients plays an important role in customer satisfaction.

Gender :

Male no churners : 2625
Female no churners : 2549
Male churners : 930
Female churners : 939

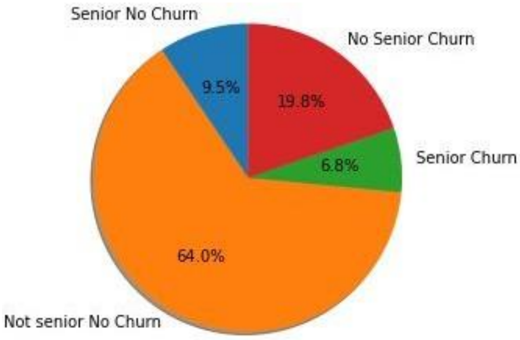
- □ Gender has no influence over the results.



Senior Citizen :

Yes no churners : 666
No no churners : 4508
Yes churners : 476
No churners : 1393

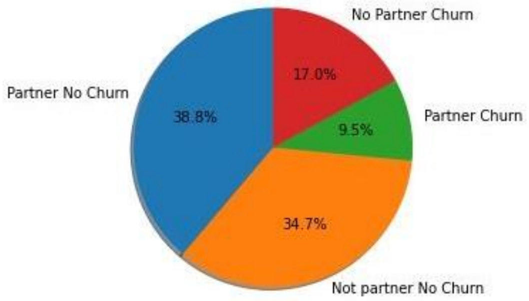
- □ The younger generation tends to churn more.



Partner :

Yes no churners : 2733
No no churners : 2441
Yes churners : 669
No churners : 1200

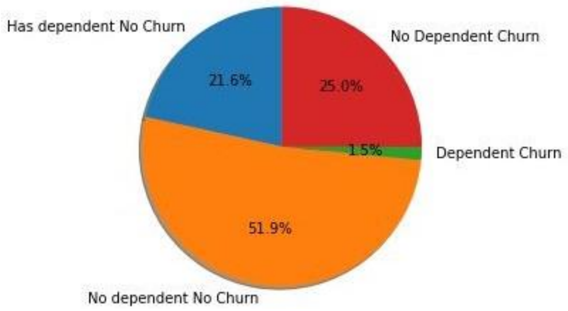
- □ Single customers have the tendency to churn more.



Dependant :

Yes no churners : 1521
No no churners : 3653
Yes churners : 106
No churners : 1763

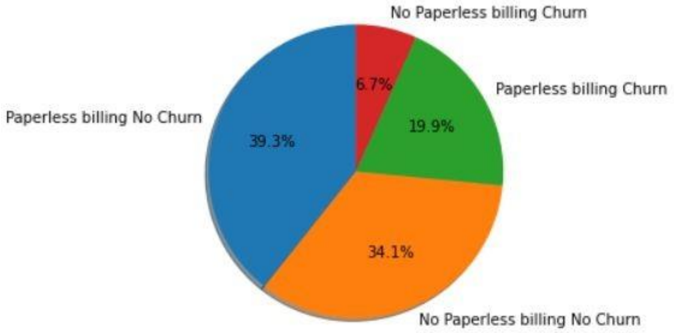
- □ Customers without kids tend to churn more



Paperless billing :

Yes no churners : 2771
No no churners : 2403
Yes churners : 1400
No churners : 469

- □ Customers with paperless billing churn more

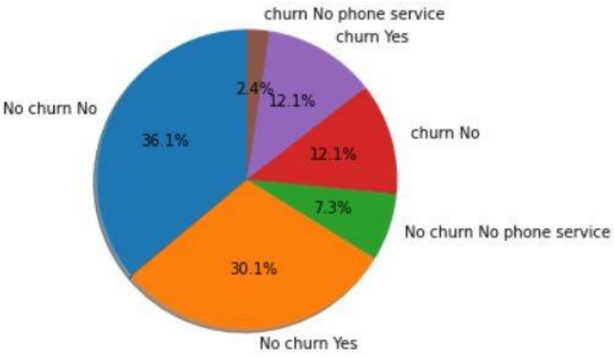


Multiple Lines :

No no churners : 2541
Yes no churners : 2121
No phone service no churners : 512

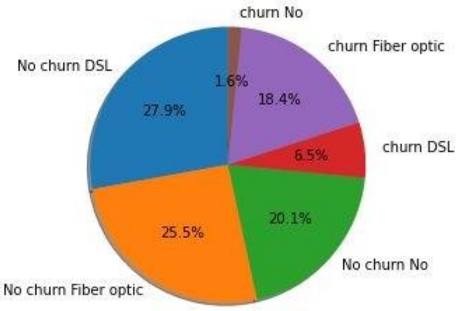
No churners : 849
Yes churners : 850
No phone service churners : 170

- □ Multiple Lines has no influence over the results.



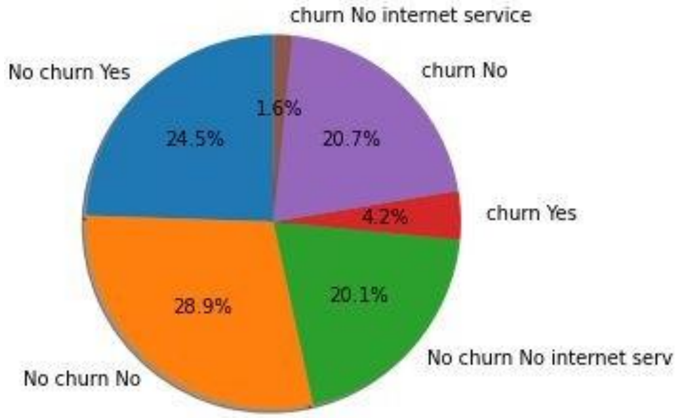
Internet Service :
DSL no churners : 1962
Fiber optic no churners : 1799
No no churners : 1413
DSL churners : 459
Fiber optic churners : 1297
No churners : 113

- □ Customers with fiber optic churn more



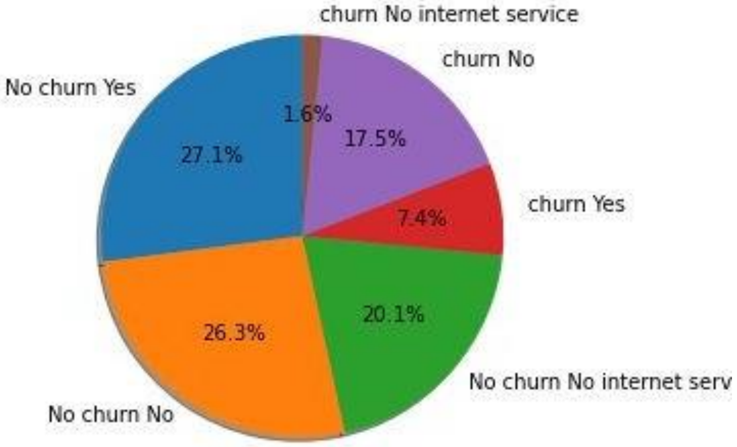
Online Security:
Yes no churners : 1724
No no churners : 2037
No internet service no churners : 1413
Yes churners : 295
No churners : 1461
No internet service churners : 113

- □ Customers without online security churn more



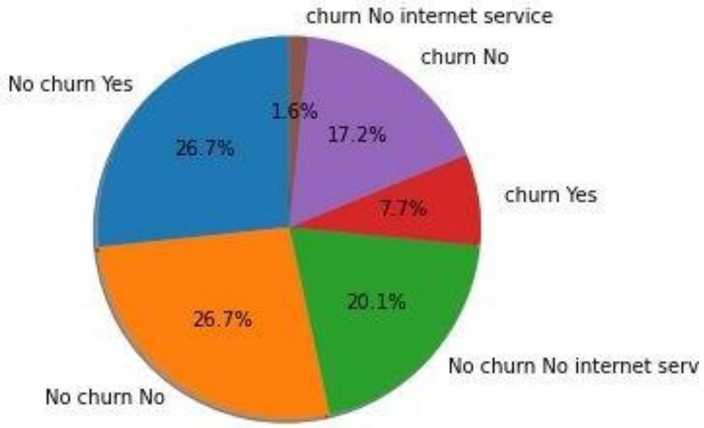
Online Backup :
Yes no churners : 1906
No no churners : 1855
No internet service no churners : 1413
Yes churners : 523
No churners : 1233
No internet service churners : 113

- □ Customers without online backup churn more



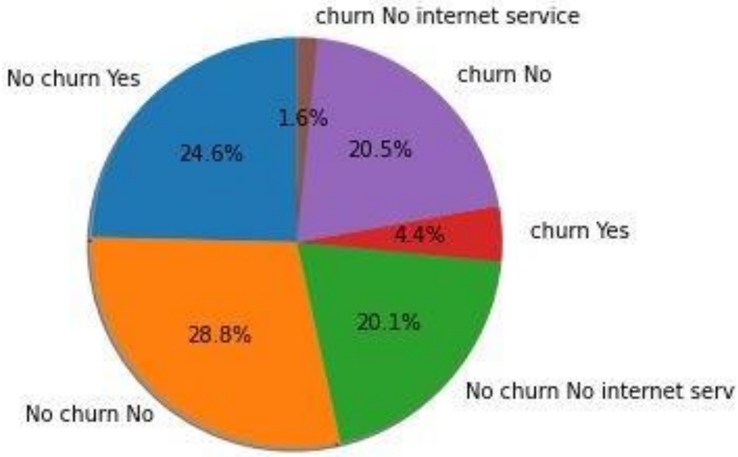
Device Protection:
Yes no churners : 1877
No no churners : 1884
No internet service no churners : 1413
Yes churners : 545
No churners : 1211
No internet service churners : 113

- □ Customers without device protection churn more



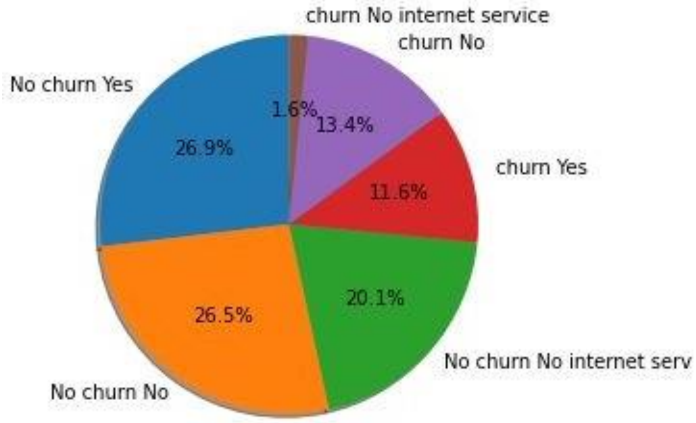
Tech Support:
Yes no churners : 1734
No no churners : 2027
No internet service no churners : 1413
Yes churners : 310
No churners : 1446
No internet service churners : 113

- □ Customers without tech support churn more



Streaming TV :
Yes no churners : 1893
No no churners : 1868
No internet service no churners : 1413
Yes churners : 814
No churners : 942
No internet service churners : 113

- □ Streaming TV has no influence over the results.



Streaming Movies :

Yes no churners : 1914

No no churners : 1847

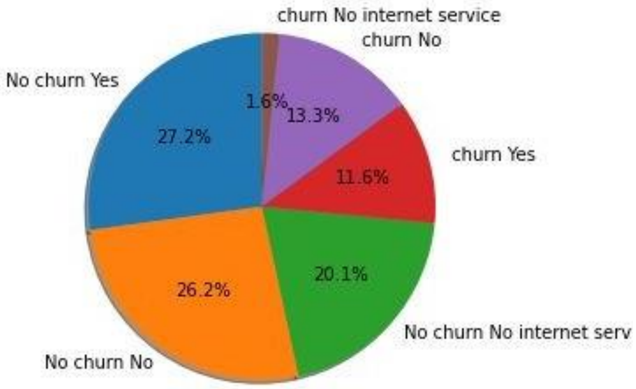
No internet service no churners : 1413

Yes churners : 818

No churners : 938

No internet service churners : 113

- □ Streaming movies has no influence over the results.



Contract :

Month-to-month no churners : 2220

Two year no churners : 1647

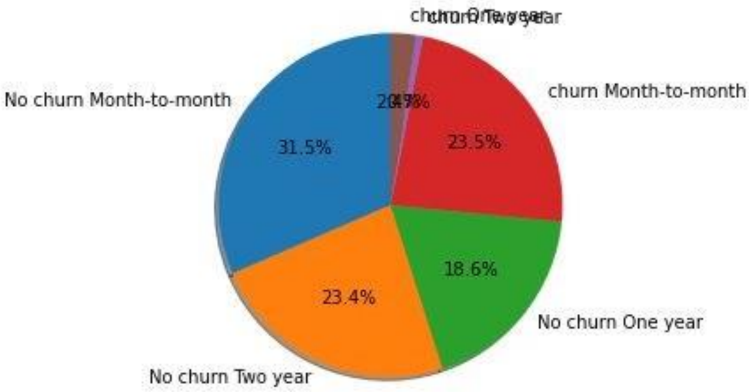
One year no churners : 1307

Month-to-month churners : 1655

Two year churners : 48

One year churners : 166

- □ Customers with month to month contracts churn more



Payment Method :

Mailed check no churners : 1304

Electronic check no churners : 1294

Bank transfer (automatic) no churners : 1286

Credit card (automatic) no churners : 1290

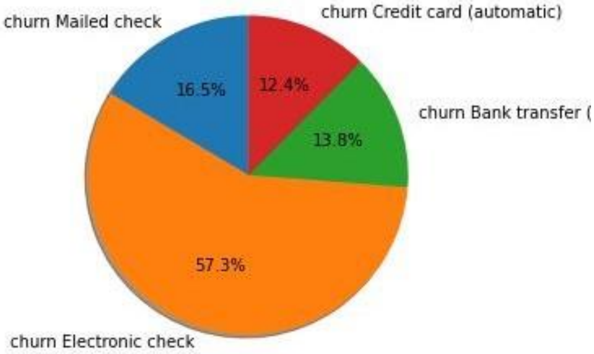
Mailed check churners : 308

Electronic check churners : 1071

Bank transfer (automatic) churners : 258

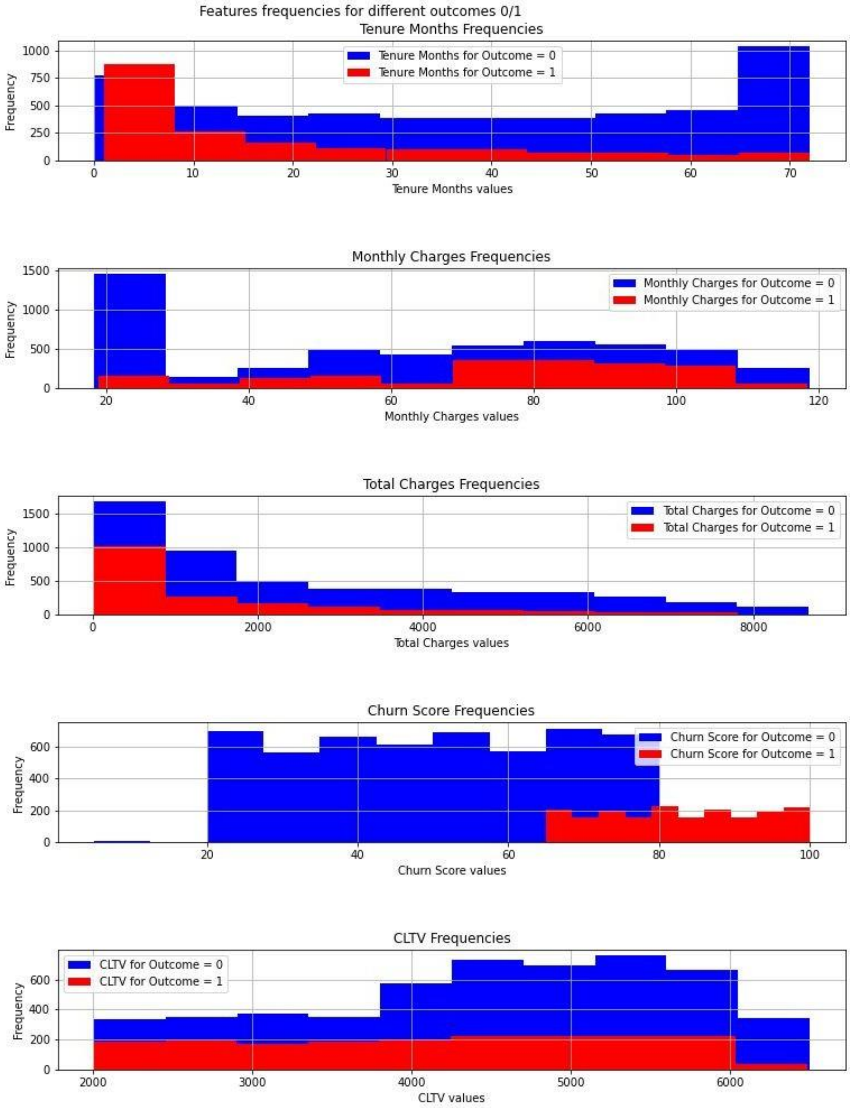
Credit card (automatic) churners : 232

- Customers who pay with electronic checks churn more



Distribution of features with continuous values:

- **New customers with a membership lasting for less than 10 months have a higher tendency to churn.**
- **Most of the churning customers have monthly charges higher than 70$.**
- **Customers that pay low monthly charges are more prone to stay.**
- **Total charges ≈ monthly charges * tenure months. Since we observed that churners have low tenure months but high monthly charges, their total charges are equal to the total charges for non churners since they have high tenure months but low monthly charges (inversed proportions) resulting the absence of a concluding argument.**
- **Clients with churn score above 80 are likely to churn.**
- **Churn scores between 65 and 80 result to unclear conclusion.**
- **CLTV is evenly distributed for churners.**



4. Data quality:

We stumbled upon some irregularities in the dataset such as unfit column types and missing values. We started with changing the type of Total charges from String to Float. Then, the 11 missing values occurred. We observed that the Churn reason column has some missing information due to the absence of the explanation for the non-churning customers.

IV. Data preparation :

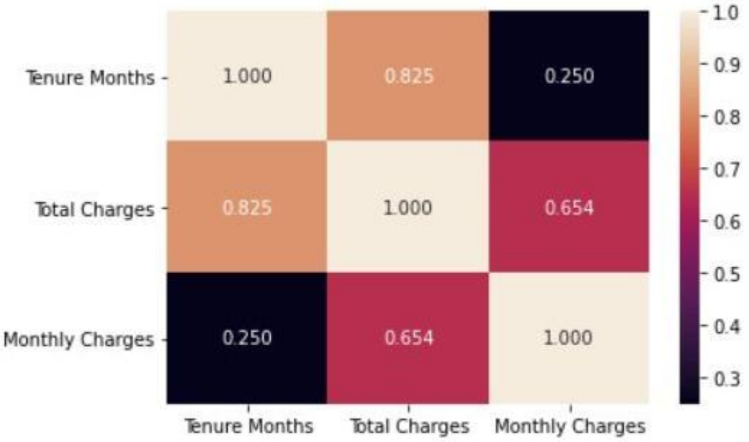
1. Data Cleaning :

Total Charges is an object type with 11 NaN rows, we should convert it to float and delete those rows. We need to drop rows with the Churn Reason “Deceased” and “Moved” because it is irrelevant to our problem and to give more accurate predictions later on.

2. Feature Selection:

In this part , we will be visualizing the correlation values between features to determine which features have no impact on our models or very small impact or correlation with Churn Value.

Earlier we said that Total Charges must be removed, the following correlation table explains the relation between our quantitative features



Total charges has a strong correlation with Monthly Charges and Tenure Months, thus we remove it.

Since our target value is a qualitative value, we established two test methods :

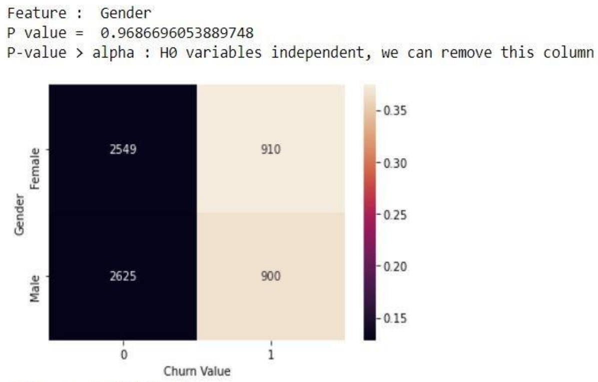
- Chi2 test for qualitative features
- Classification for quantitative features

Chi2 :

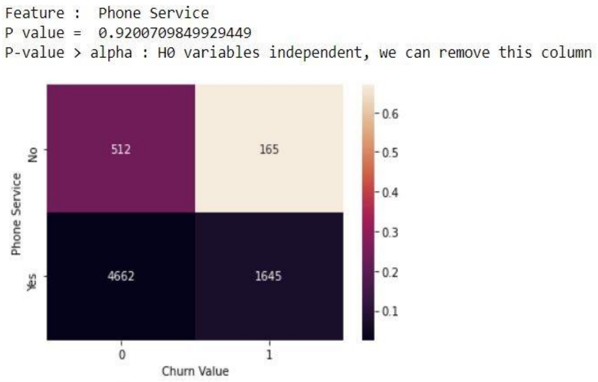
Foreach qualitative feature, we plotted the contingency table to the Churn Value (number of elements respectively to the result), and then we compared it to the independent(elementary) table.

The comparison is done by the Chi2 test, we calculate the P-value, if it is greater than 0.01 then we accept the null hypothesis : the variables are independent. Therefore, we can remove that feature.

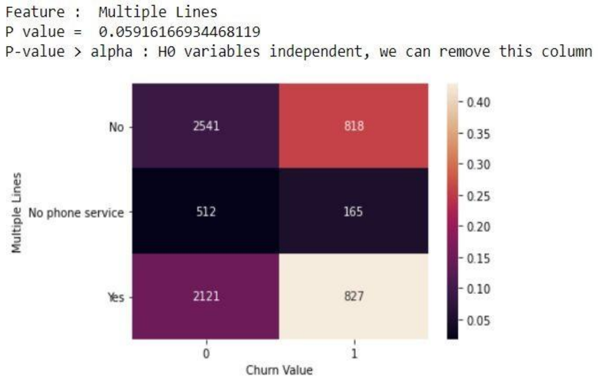
In our case, Gender, Phone Service and Multiple Lines are irrelevant to the Churn Value.



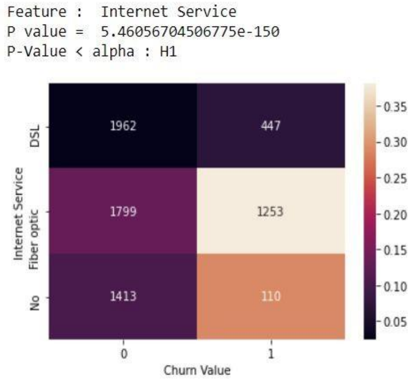
```
Feature :  Gender
P value =  0.9686696053889748
P-value > alpha : H0 variables independent, we can remove this column
```



```
Feature :  Phone Service
P value =  0.9200709849929449
P-value > alpha : H0 variables independent, we can remove this column
```



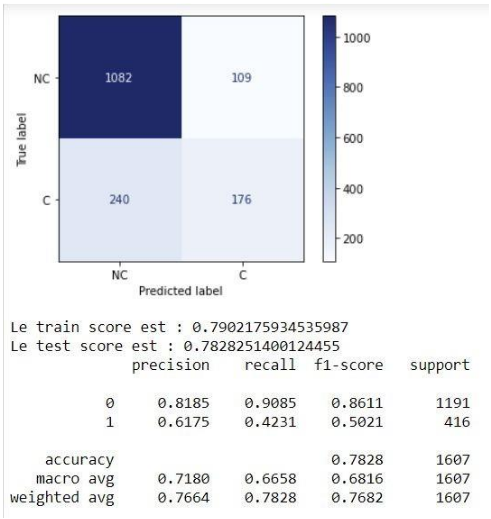
```
Feature :  Multiple Lines
P value =  0.05916166934468119
P-value > alpha : H0 variables independent, we can remove this column
```



```
Feature :  Internet Service
P value =  5.46056704506775e-150
P-Value < alpha : H1
```

For quantitative values, which are Tenure Months and Monthly Charges a classification must be done in order to evaluate their importance to the Churn Value.

The LogisticRegression model gave the following results :



```
Le train score est : 0.7902175934535987
Le test score est : 0.7828251400124455
              precision    recall  f1-score   support

           0     0.8185    0.9085    0.8611      1191
           1     0.6175    0.4231    0.5021       416

    accuracy                         0.7828      1607
   macro avg     0.7180    0.6658    0.6816      1607
weighted avg     0.7664    0.7828    0.7682      1607
```

We can conclude that these features have a good influence on the result. Therefore we keep them.

The Next Step is Dropping Irrelevant columns and columns that are not useful in modeling and that may lead to overfitting. So we dropped these columns based on:

- CustomerID : unique value for every entry
- Count: same value for every entry
- Country: same value in all rows "United States"
- State: same value in all rows " California"
- Lat Long: unique value for every entry / the combination of 2 features ('Latitude', 'Longitude') and a geolocation variable
- Latitude: unique value for every entry and a geolocation variable
- Longitude: unique value for every entry and a geolocation variable
- City: geolocation variable
- Zip Code: unique value for every entry and a geolocation variable
- Churn Label: same feature 'Churn Score' but labeled (yes or no values)
- Total Charges: this feature is the result of multiplying 2 other Features 'Monthly Charges ' and 'Tenure Months'
- Churn Reason: descriptive feature of the result , unneeded for the predicting of the churn value
- CLTV : unneeded for the predicting of the churn value as it was generated after the churning result.
- Churn Score : High correlation with the Churn Value (0.73)

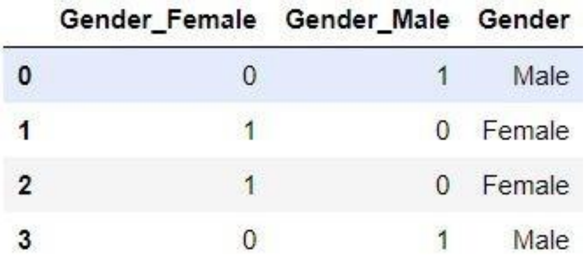
All the variable ('City','Lat Long','Latitude','Longitude','Churn Label','Zip Code') are **geolocation variables** so they are related and highly correlated with each other . So we decided to test just the correlation of the Zip Code with the Churn value which is negligible . So we removed all Geolocation variables.

3. Data Transforming :

- **Feature Encoding:**

Machine learning models can only work with numerical values. For this reason, it is necessary to transform the categorical values of the relevant features into numerical ones. We decided to use '**One-Hot encoder**' so we can transform categorical features into numerical.

This technique is used when the features are nominal(do not have any order). It creates, for each level of a categorical feature, a new variable. Each category is mapped with a binary variable containing either 0 or 1. Here, 0 represents the absence, and 1 represents the presence of that category.

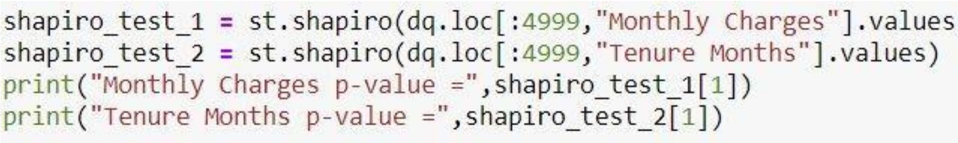


	Gender_Female	Gender_Male	Gender
0	0	1	Male
1	1	0	Female
2	1	0	Female
3	0	1	Male

- **Feature scaling:**

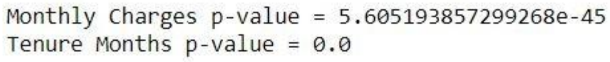
Feature scaling is a method used to normalize the range of independent variables or features of data. In data processing, it is also known as data normalization and is generally performed during the data preprocessing step. The goal is to transform features to be on a similar scale. This improves the performance and training stability of the model.

At first we will start by checking if our features are normally distributed so we decided to use the shapiro test for our Quantitative features which are 'Monthly Charges' and 'Tenure Months'.

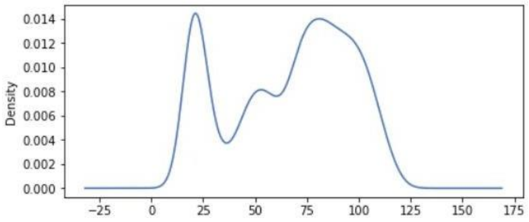


```
shapiro_test_1 = st.shapiro(dq.loc[:4999,"Monthly Charges"].values
shapiro_test_2 = st.shapiro(dq.loc[:4999,"Tenure Months"].values)
print("Monthly Charges p-value =",shapiro_test_1[1])
print("Tenure Months p-value =",shapiro_test_2[1])
```

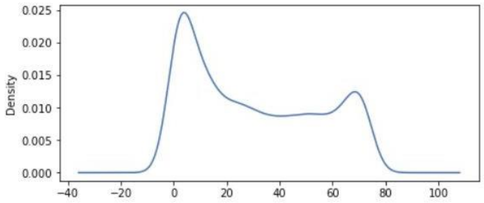
Results :



```
Monthly Charges p-value = 5.605193857299268e-45
Tenure Months p-value = 0.0
```



Monthly Charges

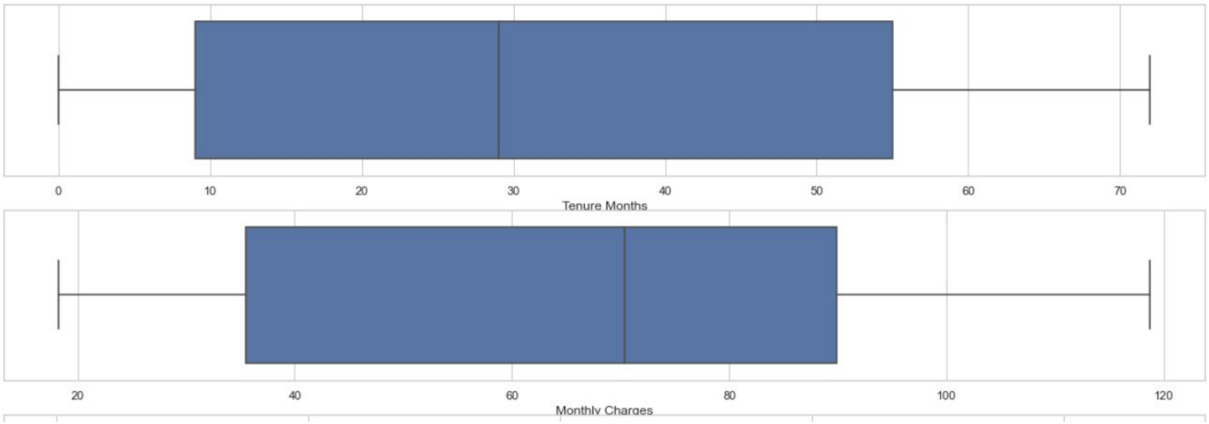


Tenure Months

As a result , we got that the quantitative features are not normally distributed , and as for the rest of the features they are qualitative features so they are not normally distributed. We conclude that we should not use the Standard scaler as it is suitable if the features are normally distributed.

The next step is checking the existence of outliers in our numerical variables.Outliers are data points that are far from other data points. In other words, they’re unusual values in a dataset. Outliers are problematic for many statistical analyses because they can cause tests to either miss significant findings or distort real results.

Outliers :



After trying all the other scalers to find the one that generates the best results, as they are all suitable for the scaling of our data, we chose the **Robust scaler** .This Scaler removes the median and scales the data according to the quantile range between the 1st quartile (25th quantile) and the 3rd quartile (75th quantile).

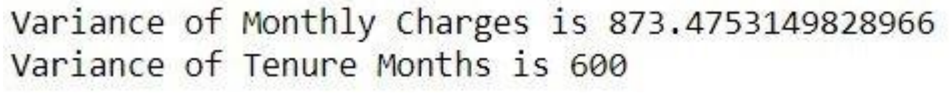
- **Data Augmentation :**

The number of churners represents only 26% of our Dataset, thus our dataset suffers from a class imbalance problem. So in order to remedy this inconvenience we used SMOTE(**Synthetic Minority Over-sampling Technique)**.SMOTE is a type of data augmentation that synthesizes new samples from the existing one (1869 samples we already have), SMOTE selects samples in the minority class that are close and then draws lines between them and new samples points are located on these lines.

SMOTE's role is to balance the dataSet going from **6984** rows to **10348** making it a 50:50 between churners and no churners.

4. Dimensionality Reduction :

After Cleaning and transforming the data , we currently have 15 Features that have important variance and important correlation value with Churn value , therefore we won't be using any Dimensionality Reduction techniques on our data.



```
Variance of Monthly Charges is 873.4753149828966
Variance of Tenure Months is 600
```

V. Modeling:

1. Selection of modeling techniques

The selection of model techniques we decided on are: Decision Tree, Logistic Regression, Naive Bayes, XGboost, Support Vector Machine, Random Forest and KNN.

- **Naive Bayes :**

Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. There is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable.

There are 3 distributions : Gaussian naïve Bayes, Multinomial naïve Bayes and Bernouill naïve Bayes.

Gaussian naïve Bayes: When dealing with continuous data, a typical assumption is that the continuous values associated with each class are distributed according to a normal (or Gaussian) distribution.

Multinomial naïve Bayes: We won't use this variation because it does not match with the scaler we're using as it generates negative values and the Multinomial naïve Bayes doesn't support that.

Bernoulli naïve Bayes: In the multivariate Bernoulli event model, features are independent Booleans (binary variables) describing inputs.

- **Logistic Regression :**

Logistic regression models the probabilities for classification problems with two possible outcomes. It's an extension of the linear regression model for classification problems.

Logistic Regression is used when the dependent variable(target) is categorical. Just like our data , To predict :

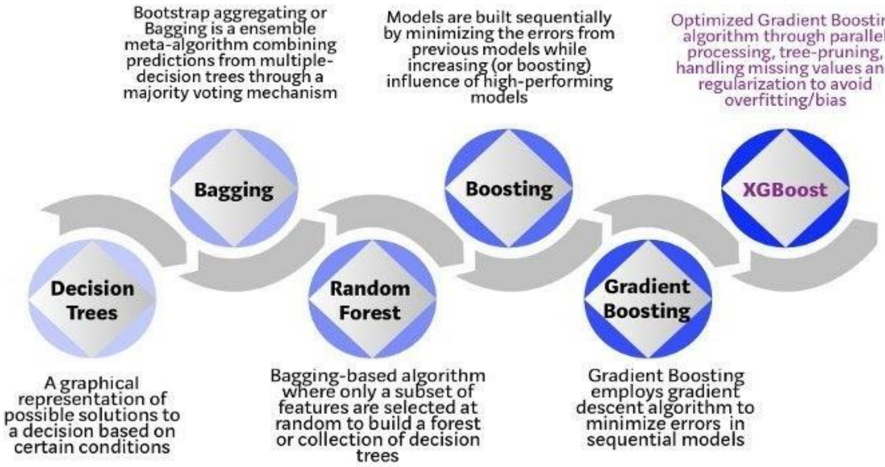
- Whether an Churn Value is (1) or (0)

If we use linear regression for this problem, there is a need for setting up a threshold based on which classification can be done. Say if the actual class is malignant, predicted continuous value 0.4 and the threshold value is 0.5, the data point will be classified as not malignant which can lead to serious consequences in real time.From this example, it can be inferred that linear regression is not suitable for classification problems. Linear regression is unbounded, and this brings logistic regression into picture. Their value strictly ranges from 0 to 1.
The logistic regression model is simply a non-linear transformation of the linear regression. The "logistic" distribution is an S-shaped distribution function which is similar to the standard-normal distribution (which results in a probit regression model) but easier to work with in most applications (the probabilities are easier to calculate). The logit distribution constrains the estimated probabilities to lie between 0 and 1.

XGBoost

XGBoost stands for “Extreme Gradient Boosting”, where the term “Gradient Boosting” originates from the paper *Greedy Function Approximation: A Gradient Boosting Machine*, by Friedman. The **gradient boosted trees** have been around for a while, and there are a lot of materials on the topic.

XGBoost is used for supervised learning problems, where we use the training data (with multiple features) xi to predict a target variable yi.
It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solves many data science problems in a fast and accurate way. The same code runs on a major distributed environment (Hadoop, SGE, MPI) and can solve problems beyond billions of examples.

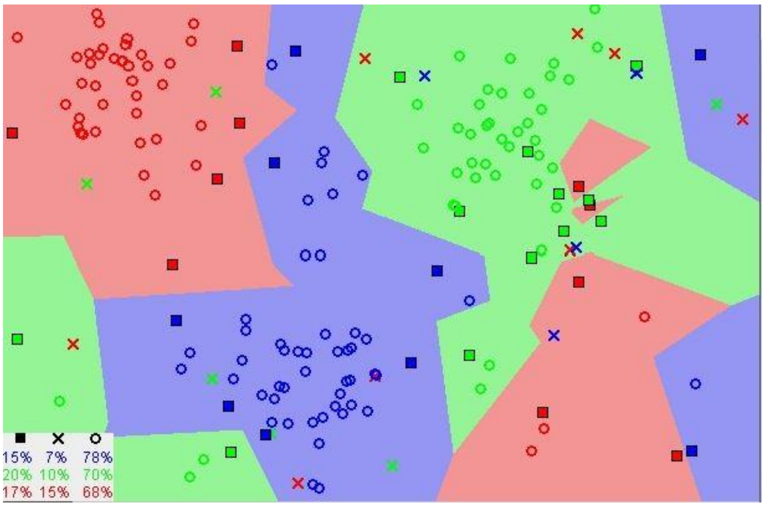

Bootstrap aggregating or Bagging is a ensemble meta-algorithm combining predictions from multiple-decision trees through a majority voting mechanism
Models are built sequentially by minimizing the errors from previous models while increasing (or boosting) influence of high-performing models
Optimized Gradient Boosting algorithm through parallel processing, tree-pruning, handling missing values and regularization to avoid overfitting/bias
Bagging
Boosting
XGBoost
Decision Trees
Random Forest
Gradient Boosting
A graphical representation of possible solutions to a decision based on certain conditions
Bagging-based algorithm where only a subset of features are selected at random to build a forest or collection of decision trees
Gradient Boosting employs gradient descent algorithm to minimize errors in sequential models

Support Vector Machine:

A support-vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks like outliers detection. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class (so-called functional margin), since in general the larger the margin, the lower the generalization error of the classifier.

- KNN :

The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other.

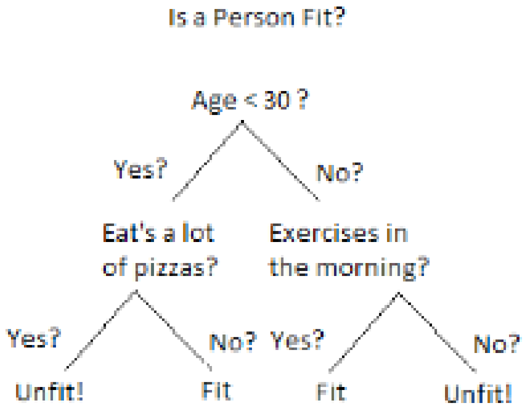


Notice in the image above that most of the time, similar data points are close to each other. The KNN algorithm hinges on this assumption being true enough for the algorithm to be useful. KNN captures the idea of similarity (sometimes called distance, proximity, or closeness) with some mathematics we might have learned in our childhood— calculating the distance between points on a graph.

- **Decision Tree**

A Decision Tree is a simple representation for classifying examples. It is a Supervised Machine Learning where the data is continuously split according to a certain parameter. TheDecision Tree consists of :

- **Nodes** : Test for the value of a certain attribute.
- **Edges/ Branch** : Correspond to the outcome of a test and connect to the next node or leaf.
- **Leaf nodes** : Terminal nodes that predict the outcome (represent class labels or class distribution).



So How does the Decision Tree algorithm work?
The basic idea behind any decision tree algorithm is as follows:

1. Select the best attribute using Attribute Selection Measures(ASM) to split the records.
2. Make that attribute a decision node and breaks the dataset into smaller subsets.
3. Starts tree building by repeating this process recursively for each child until one of the condition will match:
    - All the tuples belong to the same attribute value.
    - There are no more remaining attributes.
    - There are no more instances.

- **Random Forest :**

Random forests is a supervised learning algorithm. It can be used both for classification and regression. It is also the most flexible and easy to use algorithm. A forest consists of trees. It is said that the more trees it has, the more robust a forest is. Random forests create decision trees on randomly selected data samples, get prediction from each tree and select the best solution by means of voting. It also provides a pretty good indicator of the feature importance.

Random forests have a variety of applications, such as recommendation engines, image classification and feature selection. It can be used to classify loyal loan applicants, identify fraudulent activity and predict diseases. It lies at the base of the Boruta algorithm, which selects important features in a dataset.

What do we need in order for our random forest to make accurate class predictions?

1. **We need features that have at least some predictive power.** After all, if we put garbage in then we will get garbage out.
2. **The trees of the forest and more importantly their predictions need to be uncorrelated** (or at least have low correlations with each other). While the algorithm itself via feature randomness tries to engineer these low correlations for us, the features we select and the hyper-parameters we choose will impact the ultimate correlations as well.

2. Generation of test designs

The plan we omitted for training, testing and evaluating the models evolves around splitting the data to two sets: a training set and a testing set. We build the model on the train set, and estimate its quality on the separate test set.

3. **Building models**

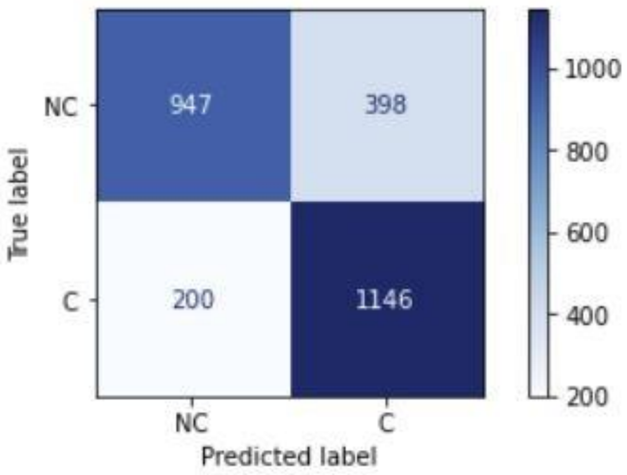
Run the modelling tool on the prepared dataset to create one or more models.

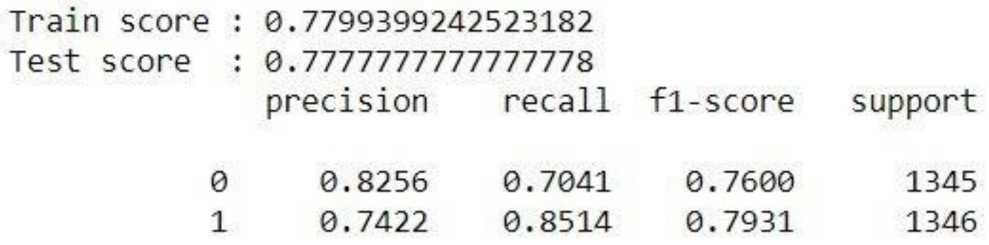
1. Parameter settings – With any modelling tool there are often a large number of parameters that can be adjusted. List the parameters and their chosen values, along with the rationale for the choice of parameter settings.
2. Models – These are the actual models produced by the modelling tool, not a report on the models.
3. Model descriptions – Describe the resulting models, report on the interpretation of the models and document any difficulties encountered with their meanings.

4. **Assessing models:**

- ## Model 1: Gaussian Naive Bayes

Visualize results on a Confusion Matrix :



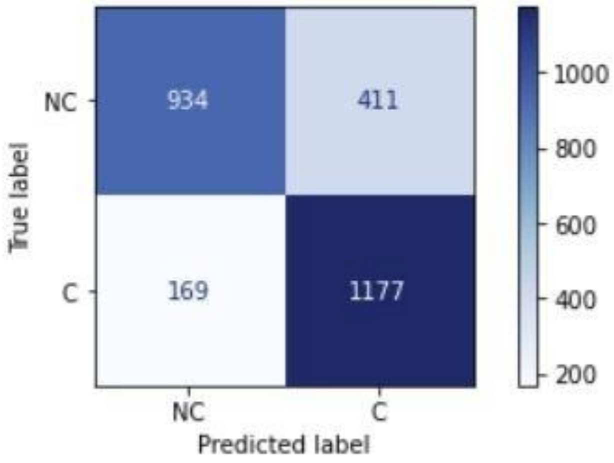


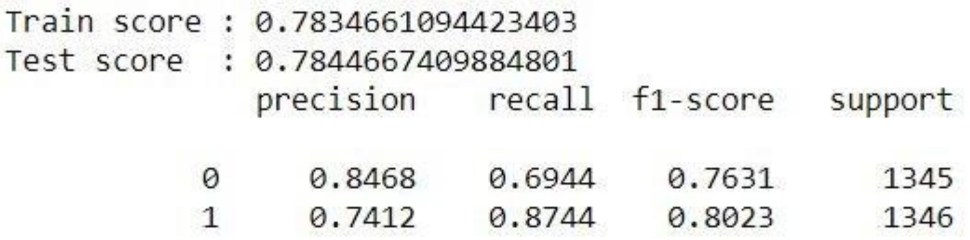
```
Train score : 0.7799399242523182
Test score  : 0.7777777777777778
              precision    recall  f1-score   support

           0     0.8256    0.7041    0.7600      1345
           1     0.7422    0.8514    0.7931      1346
```

- **Model 2 : Bernoulli Naive Bayes**

  Visualize results on a Confusion Matrix :



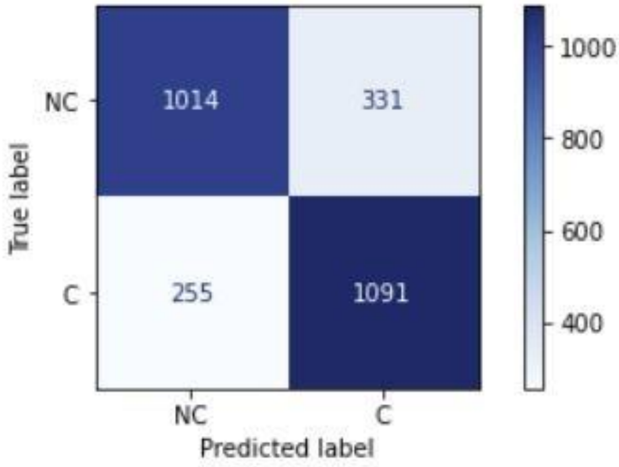


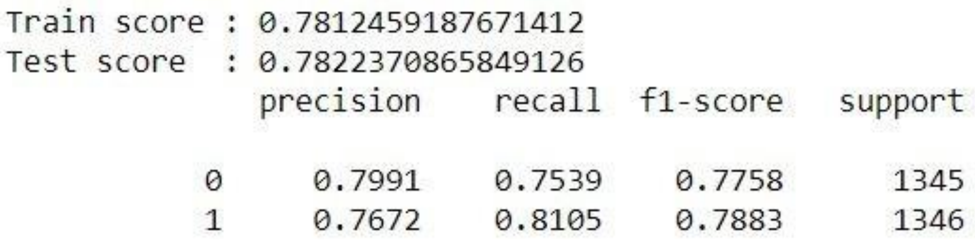
```
Train score : 0.7834661094423403
Test score  : 0.7844667409884801
              precision    recall  f1-score   support

           0     0.8468    0.6944    0.7631      1345
           1     0.7412    0.8744    0.8023      1346
```

- **Model 3 : Logistic Regression**

  Visualize results on a Confusion Matrix :



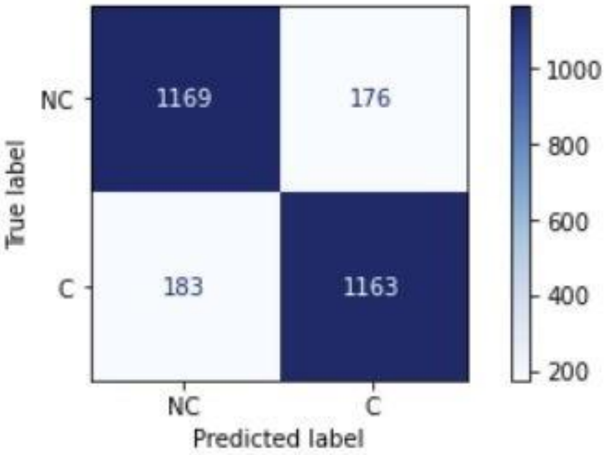


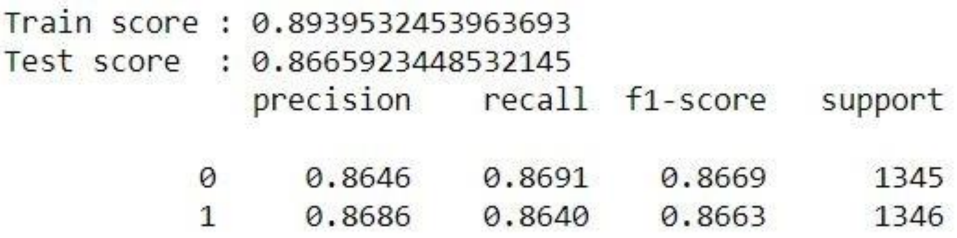
```
Train score : 0.7812459187671412
Test score  : 0.7822370865849126
              precision    recall  f1-score   support

           0     0.7991    0.7539    0.7758      1345
           1     0.7672    0.8105    0.7883      1346
```

- **Model 4 : XGBOOST**

Visualize results on a Confusion Matrix :



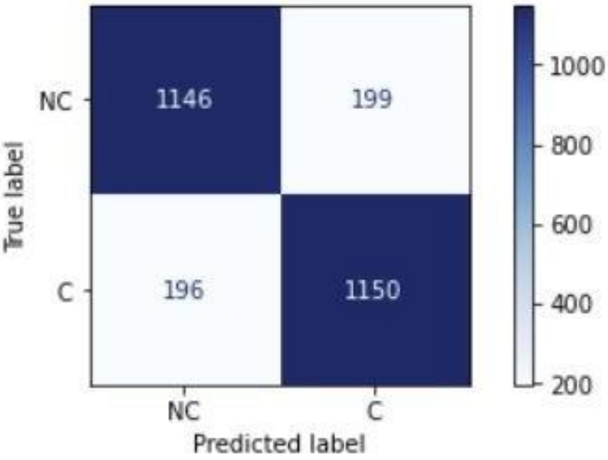


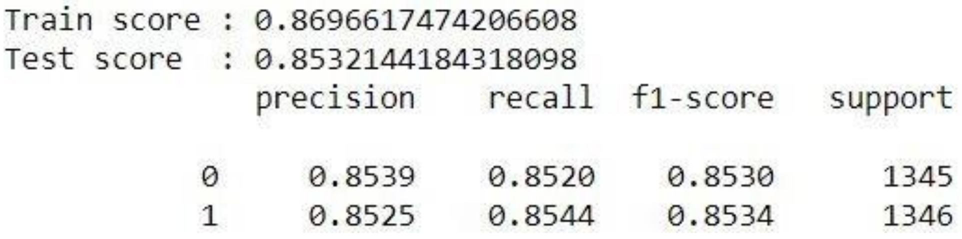
```
Train score : 0.8939532453963693
Test score  : 0.8665923448532145
              precision    recall  f1-score   support

           0     0.8646    0.8691    0.8669      1345
           1     0.8686    0.8640    0.8663      1346
```

- **Model 5 : Support Vector Machine (SVM)**

Visualize results on a Confusion Matrix :



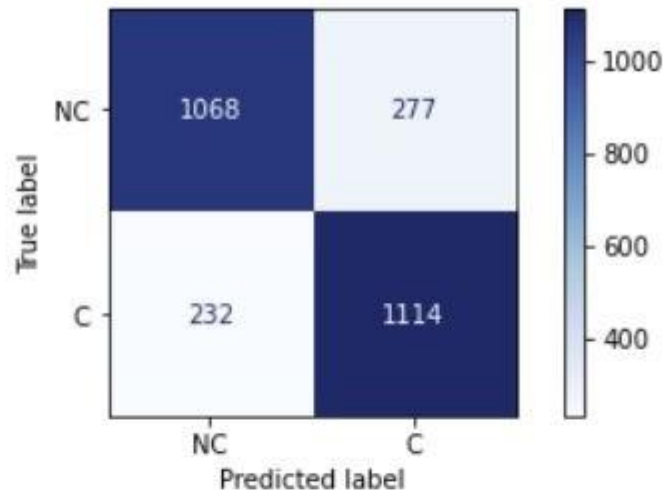


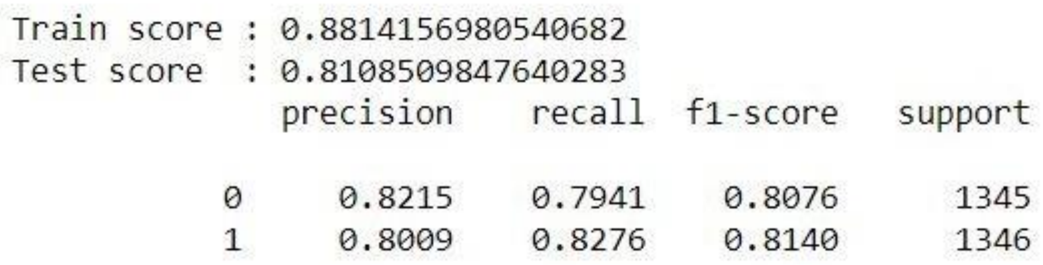
```
Train score : 0.8696617474206608
Test score  : 0.8532144184318098
              precision    recall  f1-score   support

           0     0.8539    0.8520    0.8530      1345
           1     0.8525    0.8544    0.8534      1346
```

- **Model 6 : KNeighbors**

Visualize results on a Confusion Matrix :



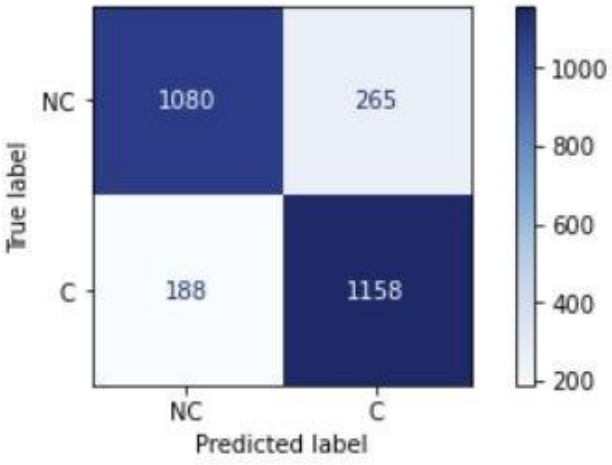


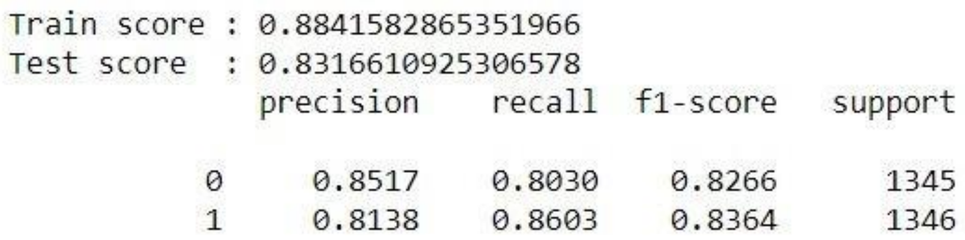
```
Train score : 0.8814156980540682
Test score  : 0.8108509847640283
              precision    recall  f1-score   support

           0     0.8215    0.7941    0.8076      1345
           1     0.8009    0.8276    0.8140      1346
```

- **Model 7 : Decision Tree**

Visualize results on a Confusion Matrix :



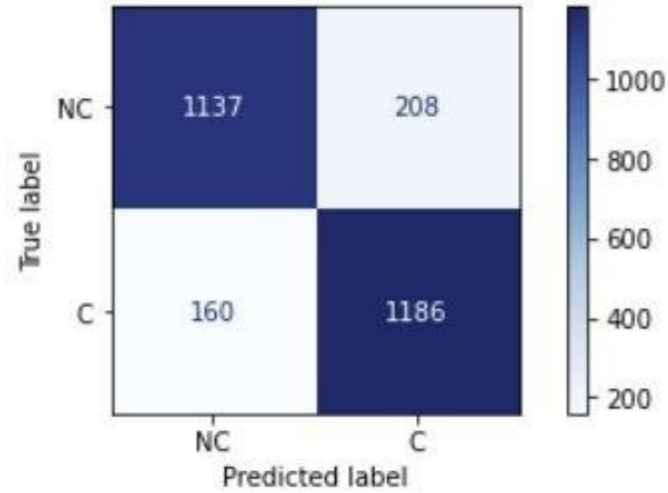


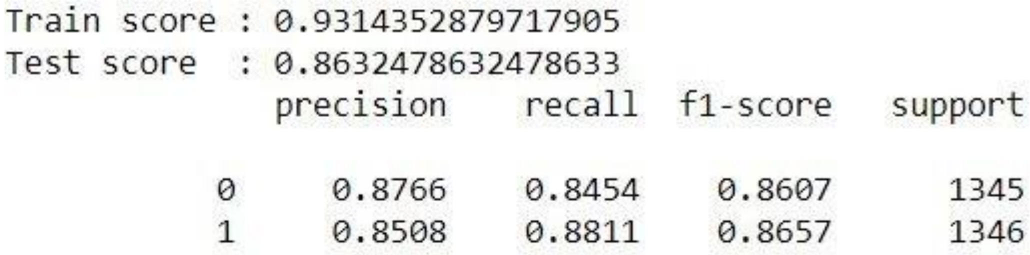
```
Train score : 0.8841582865351966
Test score  : 0.8316610925306578
              precision    recall  f1-score   support

           0     0.8517    0.8030    0.8266      1345
           1     0.8138    0.8603    0.8364      1346
```

- **Model 8 : Random Forest**

  Visualize results on a Confusion Matrix :





```
Train score : 0.9314352879717905
Test score  : 0.8632478632478633
              precision    recall  f1-score   support

           0     0.8766    0.8454    0.8607      1345
           1     0.8508    0.8811    0.8657      1346
```

VI. Evaluation:

This section analyses the results obtained from the experiments done in the research. This chapter evaluates the predictive power of the supervised machine learning models . Each model was compared by its F1-score. A comparison was also performed with the results obtained with an imbalanced dataset and sampled data using SMOTE technique.

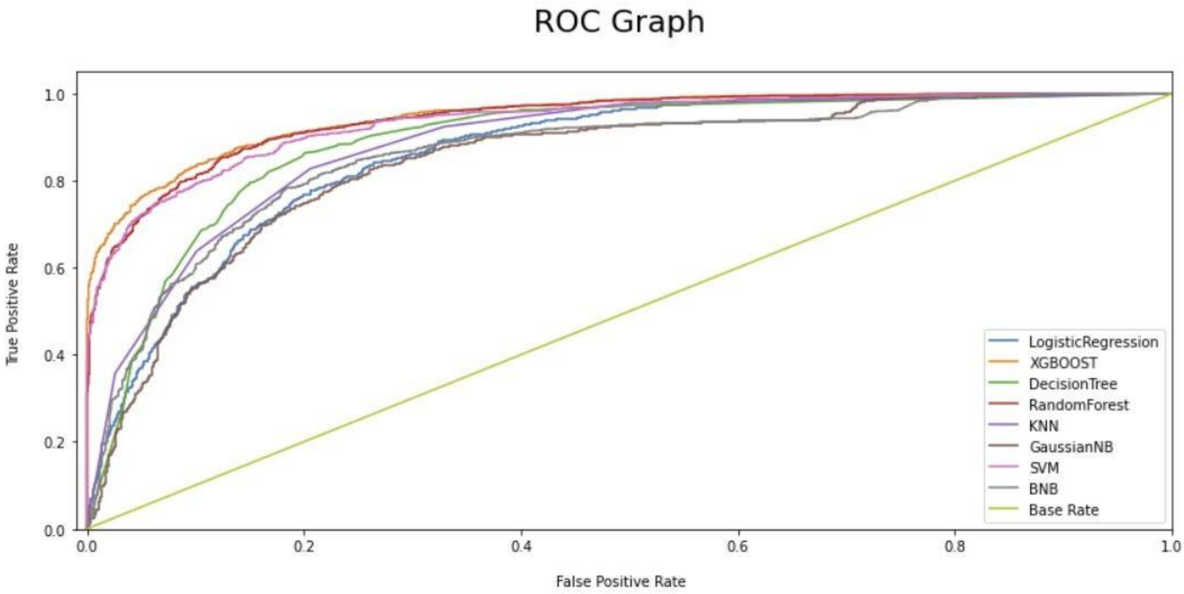
1. Results evaluation:

An experiment was performed to find the best supervised machine learning model in predicting the customer churn. In this experiment, the same set of experiments was performed twice, one with the imbalanced dataset and the other with balanced dataset using SMOTE sampling technique.

It has been observed that the supervised machine learning algorithms performed better with balance dataset in terms of F1-score ,accuracy, precision and specificity as the evaluation metrics.

In both the experiments, performed in terms of all the evaluation metrics XGBOOST machine learning algorithm has outperformed the Logistic Regression, SVM, KNN, Random forest, Decision tree and Naive bayes .

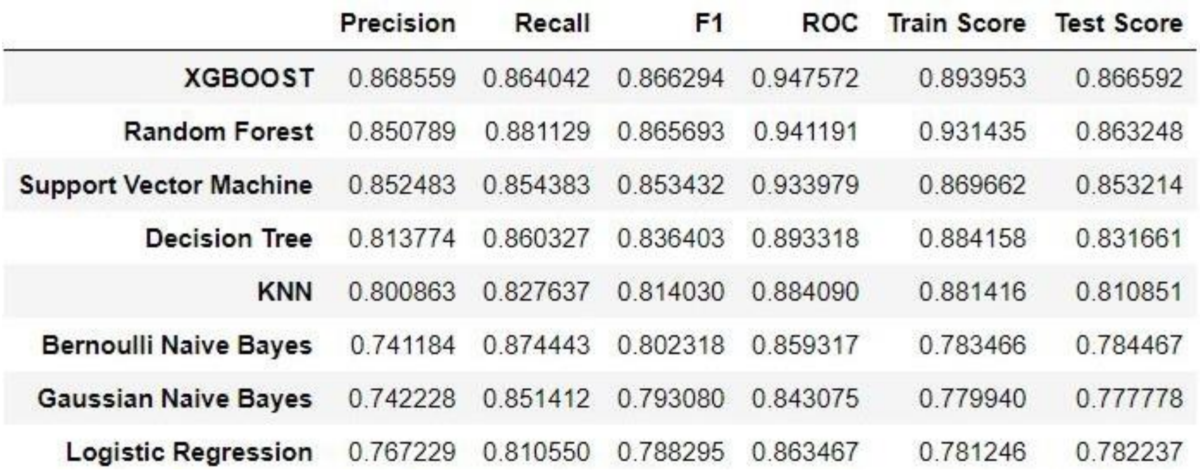
The Receiver Operating Characteristic (ROC) curve was also plotted for the best supervised model XGBOOST in predicting the customer churn.



The ROC curve is a plot between true positive rate and false positive rate.

The area under the ROC curve (AUC) is a measure of how well parameters can distinguish between churned and not churned groups. The AUC is better if it is close to 1.

This table summarises the results we obtained from the models we opted on our dataset ranked from the best model to worst based on precision , F1 score and Roc(Auc) score.



	Precision	Recall	F1	ROC	Train Score	Test Score
XGBOOST	0.868559	0.864042	0.866294	0.947572	0.893953	0.866592
Random Forest	0.850789	0.881129	0.865693	0.941191	0.931435	0.863248
Support Vector Machine	0.852483	0.854383	0.853432	0.933979	0.869662	0.853214
Decision Tree	0.813774	0.860327	0.836403	0.893318	0.884158	0.831661
KNN	0.800863	0.827637	0.814030	0.884090	0.881416	0.810851
Bernoulli Naive Bayes	0.741184	0.874443	0.802318	0.859317	0.783466	0.784467
Gaussian Naive Bayes	0.742228	0.851412	0.793080	0.843075	0.779940	0.777778
Logistic Regression	0.767229	0.810550	0.788295	0.863467	0.781246	0.782237

It could be seen that AUC was equal to **0.95** for the **XGBOOST** model which implies that this model is quite good in predicting the customer churn.

2. Process review

Supervised machine learning models have performed well with an balanced dataset as compared to the imbalanced dataset.(SMOTE)

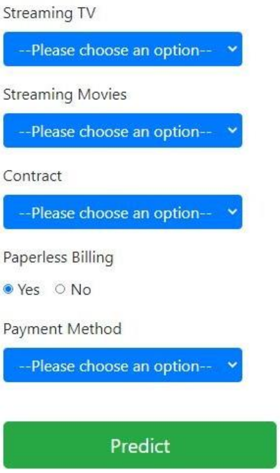
3. Determining next step

The XGBOOST model has the highest F1-score of 86.62%, recall as 86.4%, precision as 86.85% so this model was chosen as the best model for predicting customer churn in this dataset. So we will use it in the deployment .

VII. Deployment:

The model chosen for deployment is the one with the best scores which is the XGBoost.

The deployment is in a form of a survey for a seamless user experience and it is developed with the help of the web framework Django.



Streaming TV

--Please choose an option--

Streaming Movies

--Please choose an option--

Contract

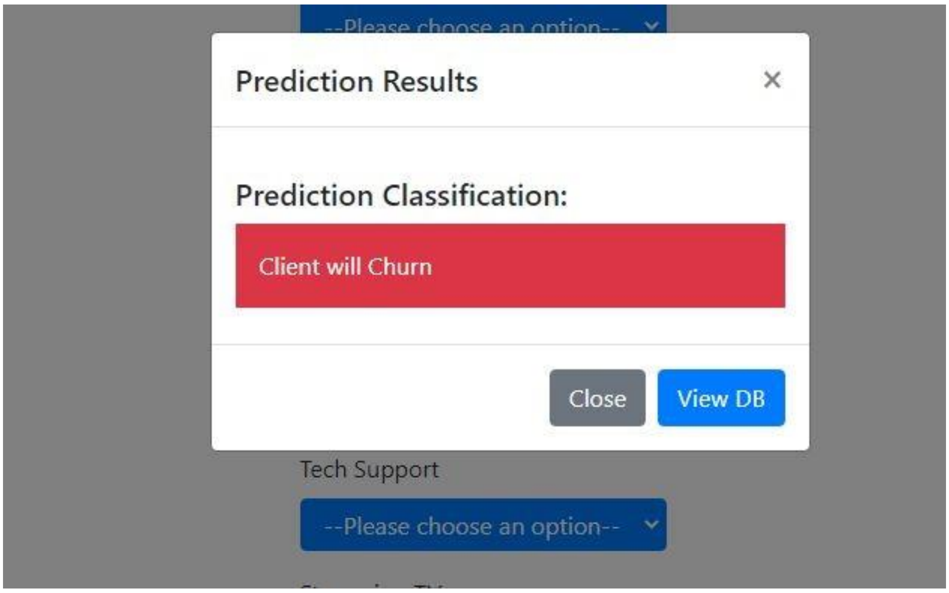
--Please choose an option--

Paperless Billing

● Yes ○ No

Payment Method

--Please choose an option--



--Please choose an option--

Prediction Results

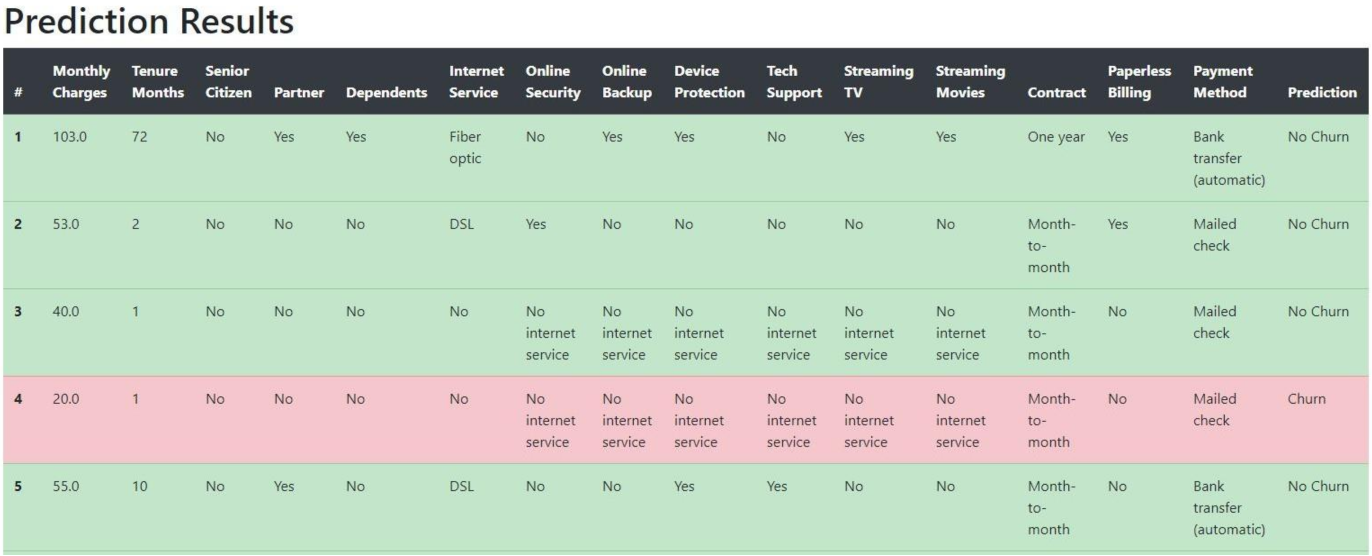
Prediction Classification:

Client will Churn

Tech Support

--Please choose an option--

The results of predictions from the survey are saved in a Dataset for further analysing .



Prediction Results

#	Monthly Charges	Tenure Months	Senior Citizen	Partner	Dependents	Internet Service	Online Security	Online Backup	Device Protection	Tech Support	Streaming TV	Streaming Movies	Contract	Paperless Billing	Payment Method	Prediction
1	103.0	72	No	Yes	Yes	Fiber optic	No	Yes	Yes	No	Yes	Yes	One year	Yes	Bank transfer (automatic)	No Churn
2	53.0	2	No	No	No	DSL	Yes	No	No	No	No	No	Month-to-month	Yes	Mailed check	No Churn
3	40.0	1	No	No	No	No	No internet service	No internet service	No internet service	No internet service	No internet service	No internet service	Month-to-month	No	Mailed check	No Churn
4	20.0	1	No	No	No	No	No internet service	No internet service	No internet service	No internet service	No internet service	No internet service	Month-to-month	No	Mailed check	Churn
5	55.0	10	No	Yes	No	DSL	No	No	Yes	Yes	No	No	Month-to-month	No	Bank transfer (automatic)	No Churn

VIII. Conclusion :

To sum up, with mobile telecommunication being the most relevant way of communication across the globe, Telecommunication companies find themselves facing a huge problem of market saturation which makes finding a new customer a much harder and costly task than keeping returning customers. To solve this issue we have been successful in predicting accurate models with high prediction and accuracy levels of customer churning behavior .As it is a costly task that affects the revenue numbers our deployment of the models will add the value of near perfect predictions to the company's resources .