CUSTOMER CHURN MODELLING

WITH MACHINE LEARNING

By & Ameur kais

SOMMAIRE

Customer Churn definition :
Also known as customer attrition, or customer turnover, is the loss of customers and it is an important and challenging problem for e-comerce and online businesses.

Importance of Churn Analysis ?

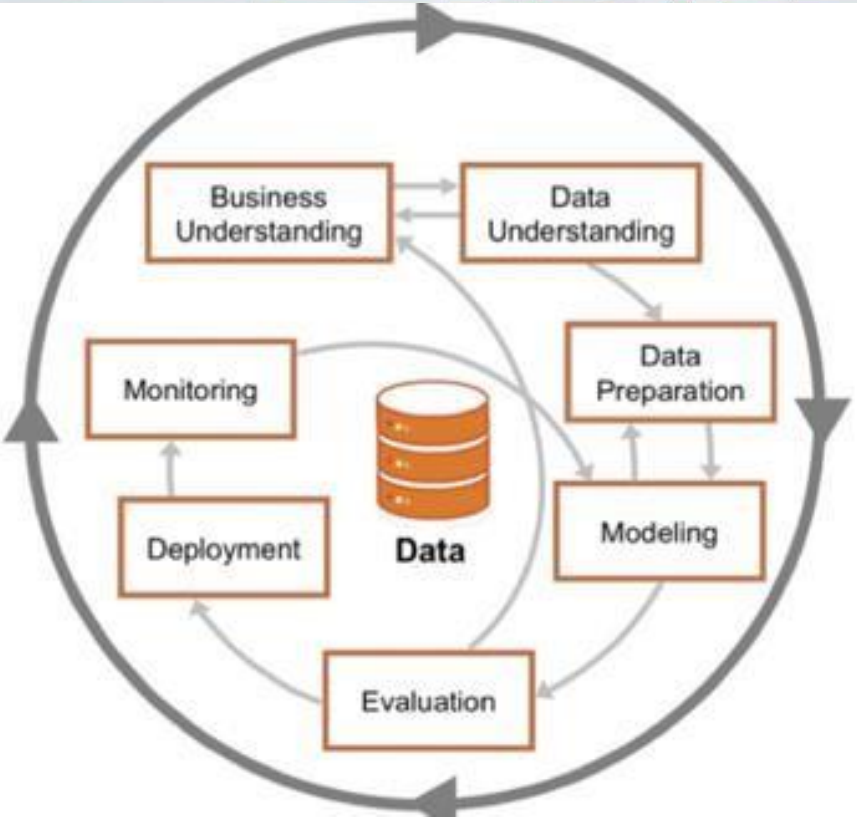
It cost more to acquire new customers than it is to retain existing ones

Churn Prediction; a technique that helps to find out which customer is more likely to churn in the given period of time.

Aims of the project :
1.Analyze how the different features affect retention or more specifically, customer churn
2.Build a classification model to predict which customers will churn
3.Models Evaluation

METHODOLOGY
CRISP-DM

We used 4-phase technique was used:
1. Data collection
2. Data understanding
3. Data preprocessing
4. Modelling and Evaluation



SOMMAIRE

A) Importation de bibliothèques et de jeux de données

Import required libraries
//Pandas to handle the Data and numpy to perform some numerical calculations.

```
import numpy as np
import pandas as pd
```

Now, we'll import the libraries for making plots.

```
import matplotlib.pyplot as plt import seaborn as sns import matplotlib.cm as cm %matplotlib inline
```

In order to preprocess and perform feature engineering, we need the LabelEncoder from scikit-learn module.

```
from sklearn.preprocessing import LabelEncoder
```

And last, we import the libraries for splitting the set, finding the best model and evaluating the final models.

```
from sklearn.model_selection import train_test_split

from sklearn.linear_model import LogisticRegression from sklearn.svm import SVC from sklearn.svm import LinearS
from sklearn.ensemble import RandomForestClassifier

from sklearn.model_selection import cross_val_score from sklearn.model_selection import GridSearchCV

from sklearn.metrics import accuracy_score, classification_report, roc_auc_score, roc_curve, precision_recall_curve
confusion_matrix

from sklearn.feature_selection import SelectFromModel
```

A) Importation de bibliothèques et de jeux de données

About the Data
The dataset used for this project contains data about customers who are withdrawing their account from a bank.

Load the data

```
df = pd.read_csv('Churn_Modelling.csv')
```

Load, Visualize and understand the dataset

There are several things we need to analyze:
- Determine the target feature and the labels in target
- Analyze the correlation between target and discrete/continuous features
- Analyze the correlation among different features
- Determine the number of missing values
- Analyze and determine the possible outlier data
- Plan to use what strategies to handle missing values and outliers

SOMMAIRE

B) Statistiques descriptives

```
1  data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   RowNumber        10000 non-null  int64
 1   CustomerId       10000 non-null  int64
 2   Surname          10000 non-null  object
 3   CreditScore      10000 non-null  int64
 4   Geography        10000 non-null  object
 5   Gender           10000 non-null  object
 6   Age              10000 non-null  int64
 7   Tenure           10000 non-null  int64
 8   Balance          10000 non-null  float64
 9   NumOfProducts    10000 non-null  int64
 10  HasCrCard        10000 non-null  int64
 11  IsActiveMember   10000 non-null  int64
 12  EstimatedSalary  10000 non-null  float64
 13  Exited           10000 non-null  int64
dtypes: float64(2), int64(9), object(3)
memory usage: 1.1+ MB
```

From the above, there are 10000 observations and 14 variables in the data set and there were no missing values. Since there are no missing values let's perform basic visualization to understand how the data is distributed.

```
1  # Checking if our dataset contains any NULL values
2  data.isnull().sum()
```

```
RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          0
Gender             0
Age                0
Tenure             0
Balance            0
NumOfProducts      0
HasCrCard          0
IsActiveMember     0
EstimatedSalary    0
Exited             0
dtype: int64
```

B) Statistiques descriptives

```
[14]  1  data.describe()
```

	RowNumber	CustomerId	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
count	10000.00000	1.000000e+04	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	10000.00000	10000.000000	10000.000000	10000.000000
mean	5000.50000	1.569094e+07	650.528800	38.921800	5.012800	76485.889288	1.530200	0.70550	0.515100	100090.239881	0.203700
std	2886.89568	7.193619e+04	96.653299	10.487806	2.892174	62397.405202	0.581654	0.45584	0.499797	57510.492818	0.402769
min	1.00000	1.556570e+07	350.000000	18.000000	0.000000	0.000000	1.000000	0.00000	0.000000	11.580000	0.000000
25%	2500.75000	1.562853e+07	584.000000	32.000000	3.000000	0.000000	1.000000	0.00000	0.000000	51002.110000	0.000000
50%	5000.50000	1.569074e+07	652.000000	37.000000	5.000000	97198.540000	1.000000	1.00000	1.000000	100193.915000	0.000000
75%	7500.25000	1.575323e+07	718.000000	44.000000	7.000000	127644.240000	2.000000	1.00000	1.000000	149388.247500	0.000000
max	10000.00000	1.581569e+07	850.000000	92.000000	10.000000	250898.090000	4.000000	1.00000	1.000000	199992.480000	1.000000

From the above, a couple of question linger:

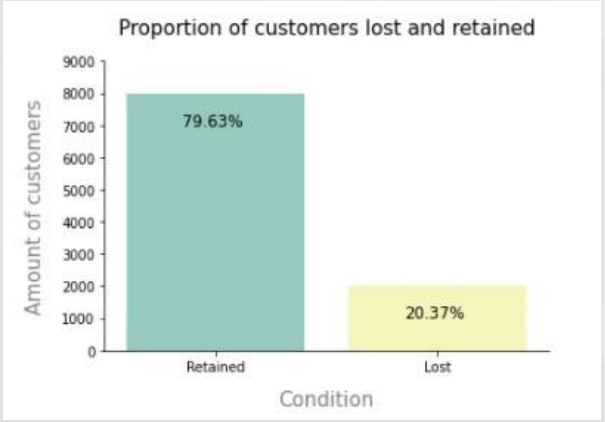
1. The data appears to be a snapshot as some point in time e.g. the balance is for a given date which leaves a lot of questions:
    1. What date is it and of what relevance is this date
    2. Would it be possible to obtain balances over a period of time as opposed to a single date.
2. There are customers who have exited but still have a balance in their account! What would this mean? Could they have exited from a product and not the bank?
3. What does being an active member mean and are there difference degrees to it? Could it be better to provide transaction count both in terms of credits and debits to the account instead?
4. A break down to the products bought into by a customer could provide more information topping listing of product count

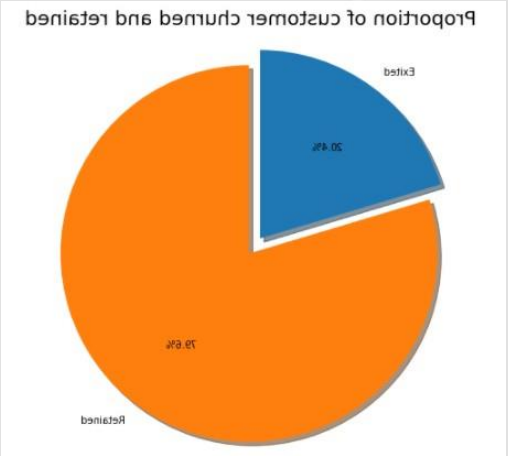
For this exercise, we proceed to model without context even though typically having context and better understanding of the data extraction process would give better insight and possibly lead to better and contextual results of the modelling process

B) Statistiques descriptives

Because our target variable is exited, we should analyse the data focusing on how the different features are related to this variable. Let's start by getting how many customers have been lost.

We can observe that **20.37%** of the customers have churned.
This information is valuable because for classification models we need to confirm that our dataset does not suffer from data imbalance, which usually reflects an unequal distribution of classes within a dataset. Even though the class is not equally distributed, we can say that it does not suffer from high-class imbalance.





B) Statistiques descriptives

We note the following:

- **Geography :** Majority of the data is from persons from France. However, the proportion of churned customers is with inversely related to the population of customers alluding to the bank possibly having a problem (maybe not enough customer service resources allocated) in the areas where it has fewer clients.

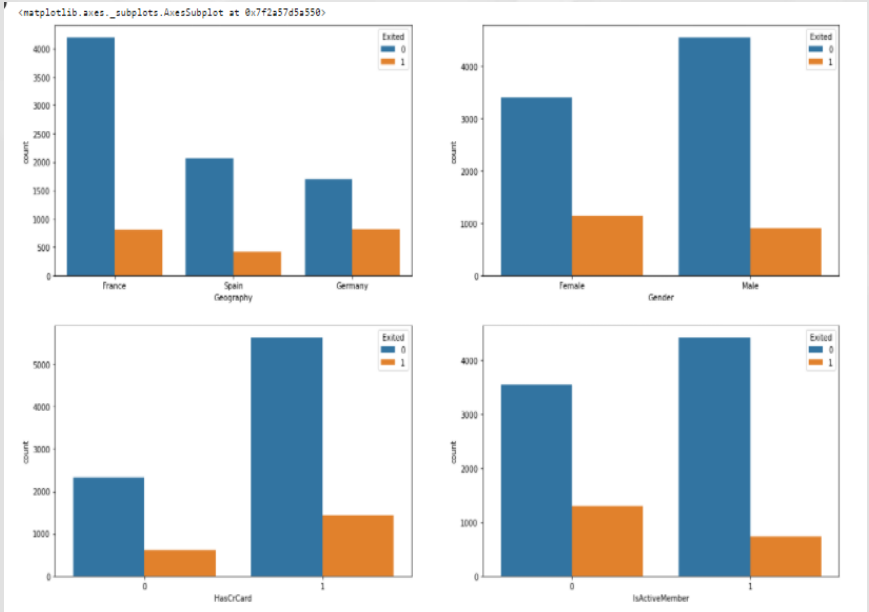
-Gender: We can clearly see the **Female customers** had more exits than the male customers.

-**Credit cards:** It is generally expected that people who have more interactions and products of the bank, would likely be retained for a longer time. However, we can see that people who have credit cards have more exits than those who do not own credit cards.

-Active Member: Unsurprisingly the inactive members have a greater churn. Worryingly is that the overall proportion of inactive mebers is quite high suggesting that the bank may need a program implemented to turn this group to active customers as this will definately have a positive impact on the customer churn.

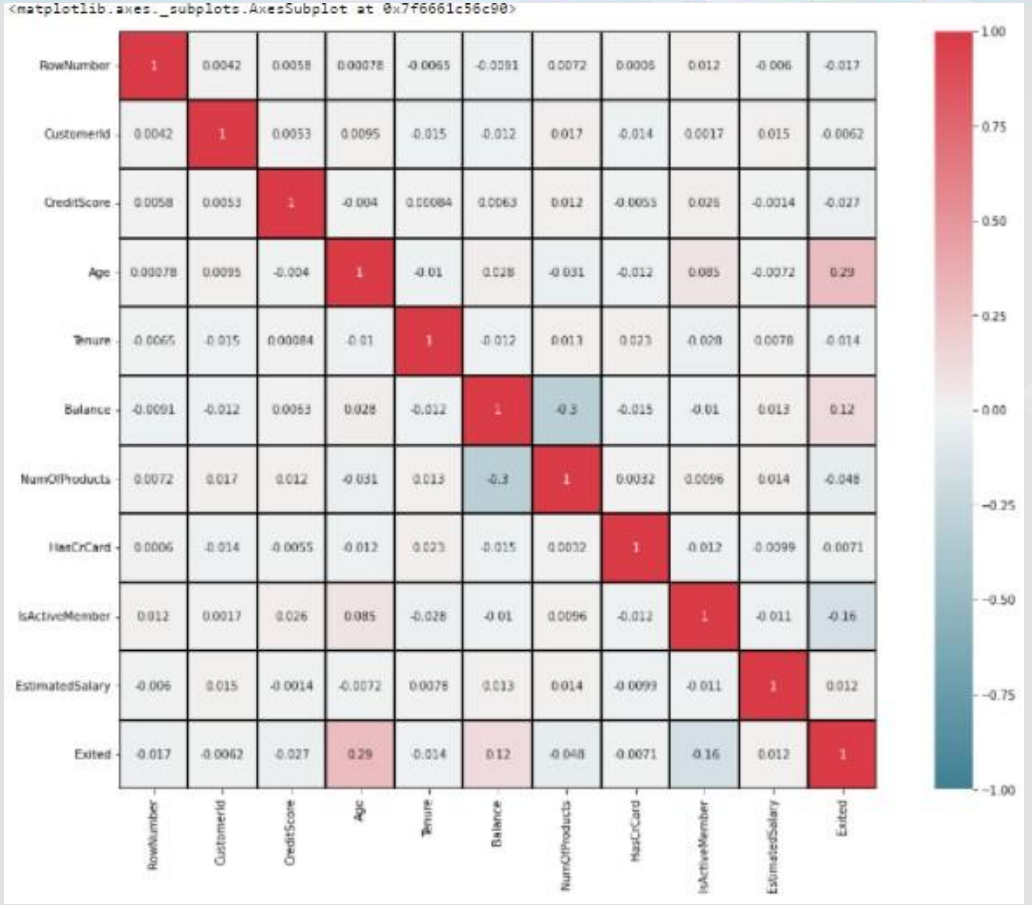
-Number of Products: This is also an expected observation, where we see that customers who own more products from the bank are likely to be retained for a longer time than those who own less products.

-Tenure: We see that the tenure of a customer does not really tell us much if that customer is likely to be churned or not. Initially, it looks like new joinees and older people (10 years) have been churned less. However, on a closer analysis we can see that the overall number of retained customer are significantly less in both these cases. As a result, we can probably conclude that new joinees and older customers may be more likely to be churned as their churn rate (percentage) is likely to be higher than other tenure rates.



B) Statistiques descriptives

-From the above, we observed age has the strongest relation with Exited (0.29).
-Here we can assume that as the age of the customer increases, the rate of losing the customer increases. (Positive strong relationship).
-Also, exited and balance variable have a relatively strong relationship (0.12).
-And Lastly, exited and the variable IsActiveMember have a moderately strong relationship (-0.16). They have a strong negative relationship.



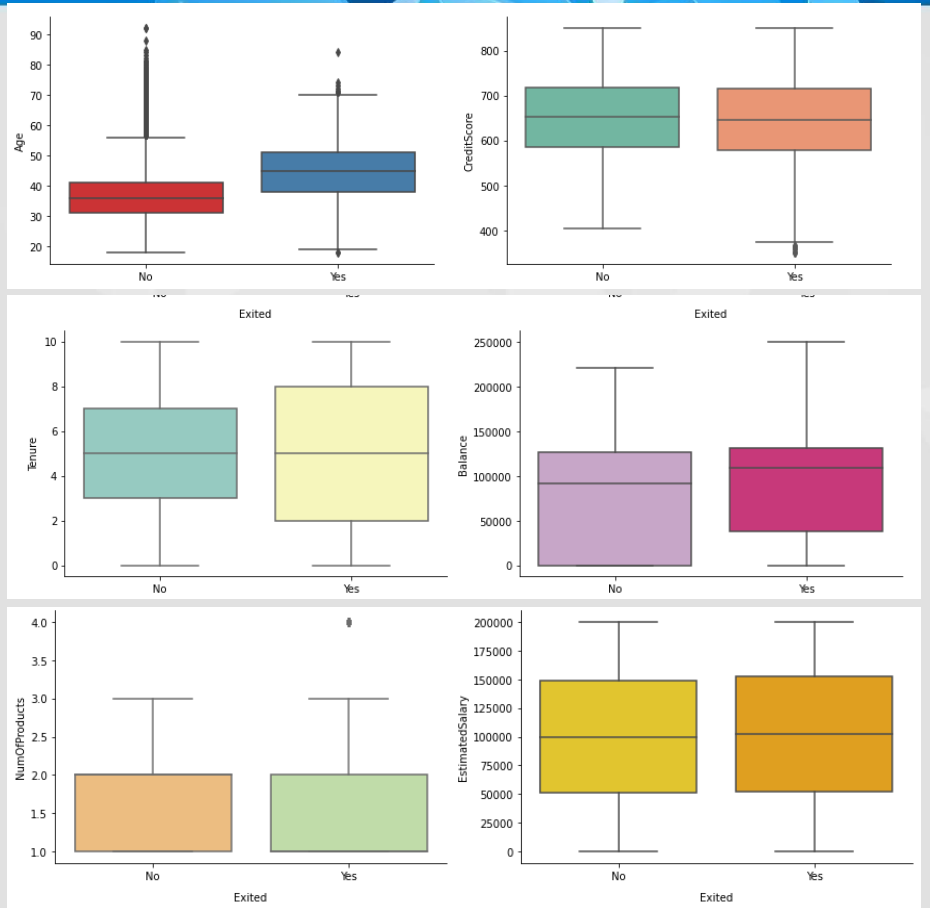
SOMMAIRE

C) Boite à moustaches

From the above visualization, there is the presence of outliers in columns such as “CreditScore”, “Age”, “NumOfProducts”.

Those outliers can be removed and cleaned

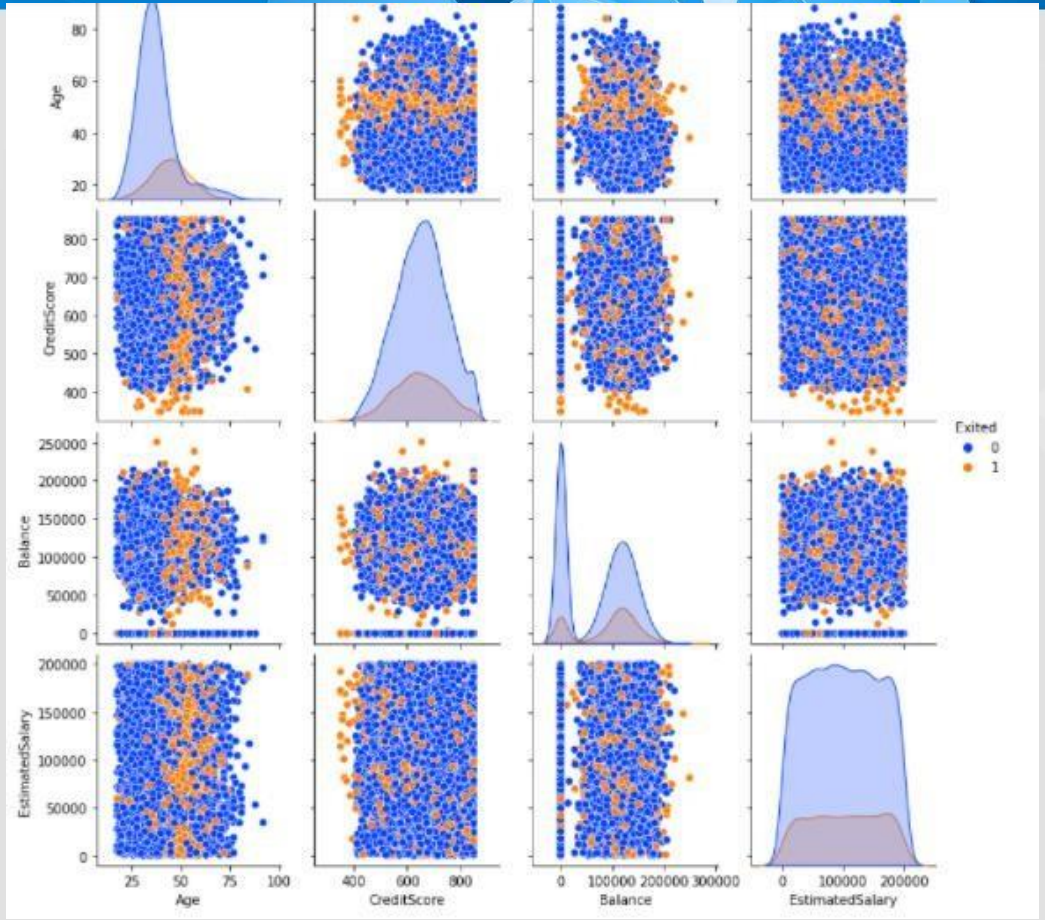
From the plots, we learn several things:
- **Credit score :**There is no significant difference in the **credit score** distribution between retained and churned customers.
-**Age** : The older customers are churning at more than the younger ones alluding to a difference in service preference in the age categories. The bank may need to review their target market or review the strategy for retention between the different age groups
-**Tenure** : the clients on either extreme end (spent little time with the bank or a lot of time with the bank) are more likely to churn compared to those that are of average tenure.
-**Balance** : the bank is losing customers with significant bank balances which is likely to hit their available capital for lending.
-Neither **estimated salary** nor the **number of products** seems to have an effect on customer churn.



SOMMAIRE

B) Nuage de points

No relationship can be inferred between continuous variables from the plots.



SOMMAIRE

F) Encodage des données catégorielles

Since some Classifier algorithms do not accept categorical fields, they need to be transformed
There are two methods for transforming categorical fields:

1. Label encoding - each unique category value is assigned an integer value
2. One Hot encoding - the integer encoded variable is removed and a new binary variable is added for each unique integer value

The first one is not preferred since the classification algorithms tend to assume a natural ordering between categories, which may result in poor performance or unexpected results

F) Encodage des données catégorielles

Encoding Categorical variables into numerical variables One Hot Encoding

```
[185] 1
      2 x = pd.get_dummies(x)
      3 x.head()
```

	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Geography_France	Geography_Germany	Geography_Spain	Gender_Female	Gender_Male
0	619	42	2	0.00	1	1	1	101348.88	1	0	0	1	0
1	608	41	1	83807.86	1	0	1	112542.58	0	0	1	1	0
2	502	42	8	159660.80	3	1	0	113931.57	1	0	0	1	0
3	699	39	1	0.00	2	0	0	93826.63	1	0	0	1	0
4	850	43	2	125510.82	1	1	1	79084.10	0	0	1	1	0

```
[186] 1 x.shape
```

```
(10000, 13)
```

SOMMAIRE

G) Fractionnement de l'ensemble de données en ensembles de train et de test

G) Fractionnement de l'ensemble de données en ensembles de train et de test

In order to train and test our model, we need to split our dataset into to subdatasets, the training and the test dataset.

It is common to use the rule of 80%-20% to split the original dataset. It is important to use a reliable method to split the dataset to avoid data leakage; this is the presence in the test set of examples that were also in the training set and can cause overfitting.

```
# splitting the data into training and testing set

from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2,shuffle=True, stratify=y, random_state = 0)

print(x_train.shape)
print(y_train.shape)
print(x_test.shape)
print(y_test.shape)
```

```
(8000, 13)
(8000,)
(2000, 13)
(2000,)
```

SOMMAIRE

H) Features Scaling

Feature scaling (standardization) using sklearn library (StandardScalar) to scale down features into properties of Standard Normal Distribution where mean = 0 and standard deviation = 1.

```
from sklearn.preprocessing import StandardScaler

sc = StandardScaler()
x_train = sc.fit_transform(x_train)
x_test = sc.fit_transform(x_test)

x_train = pd.DataFrame(x_train)
x_train.head()
```

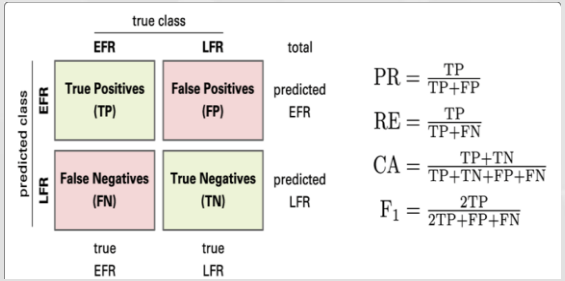
	0	1	2	3	4	5	6	7	8	9	10	11	12
0	-1.240217	0.779861	0.353903	-1.234514	-0.902981	-1.549632	-1.038490	1.640990	0.998002	-0.578120	-0.575041	1.096651	-1.096651
1	0.759749	-0.273827	0.353903	0.285421	0.813713	0.645314	0.962936	-1.555875	-1.002002	1.729744	-0.575041	-0.911867	0.911867
2	-1.727256	-0.944356	-0.339090	0.855696	-0.902981	0.645314	-1.038490	1.103811	0.998002	-0.578120	-0.575041	1.096651	-1.096651
3	0.044735	-0.178037	0.353903	0.518006	0.813713	0.645314	-1.038490	-1.709357	0.998002	-0.578120	-0.575041	-0.911867	0.911867
4	-1.924143	-0.561197	0.007406	-1.234514	0.813713	-1.549632	0.962936	-0.375574	0.998002	-0.578120	-0.575041	-0.911867	0.911867

SOMMAIRE

2. Modelling (KNN, D_Tree, R_Forest, SVM, Logistic_Regression)

Random Forest with the best accurarcy & best tradeoff between precision & recall

	Model	Accuracy (%)	precision (%)	Recall (%)
0	model_KNeighborsClassifier	0.820	0.617	0.3480
1	model_SVM	0.850	0.828	0.3430
2	model_RandomForest	0.860	0.768	0.4570
3	model_DecisionTree	0.780	0.478	0.5085
4	model_LogisticRegression	0.809	0.590	0.2100

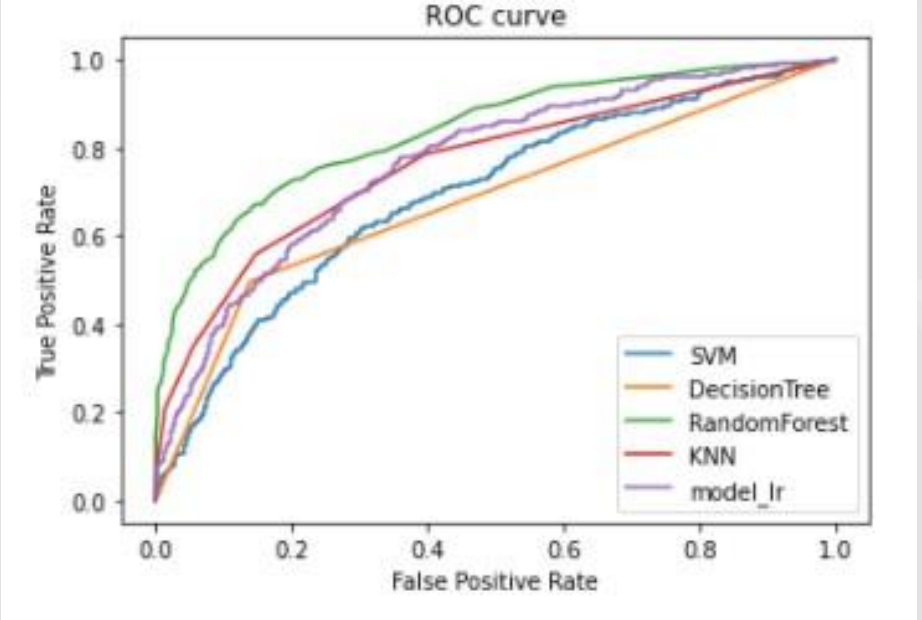


From the review of the fitted models above, the best model that gives a decent balance of the recall and precision, with a precision score on 1's of 0.76 and an accurarcy of 86% is RandomForest.

SOMMAIRE

3. Model evaluation with ROC

RandomForest is the best model with ROC metrics,



…

Future Work :

- Hyper-parameters Optimization
- features engineering
- Modelling with ANN