Churn Modelling Analysis

Isha Tyagi(1810110085)
Sanskriti Agrawal(1810110216)

Churn Modelling Analysis

Goal of the project

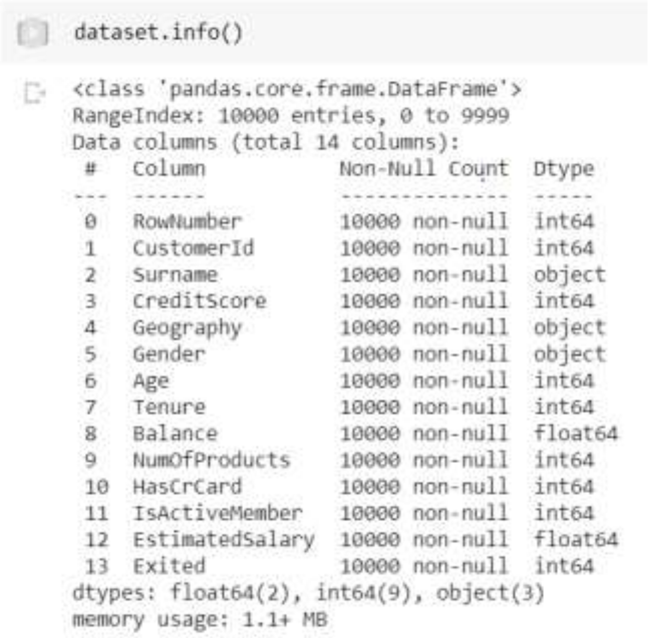
Customer churn metrics can help businesses improve customer retention.
The goal of this project is to predict customer churn, to predict that we use the target variable which is a binary variable reflecting the fact whether the customer left the bank (closed his account) or he continues to be a customer.
The best way to avoid customer churn is to know your customers, and the best way to know your customer is through historical and new customer data.

We aim to accomplish the following for this study:

- Identify and visualize which factors contribute to customer churn
- Build a prediction model that will perform the following:
    - Classify if a customer is going to churn or not
    - Preferably and based on model performance, choose a model that will make it easier for customer service to target low hanging fruits in their efforts to prevent churn by building a classifier that analyzes user data to identify when a customer is leaving the service or product.

The Dataset

The Input Variables are:



```
dataset.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
```

RowNumber	each row consist of one client information (numeric)
CustomerId	unique identifier for customers (numeric)
Surname	last name of the client (categorical)
CreditScore	Credit score of the client(numeric)
Geography	the territory of the customers (categorical)
Gender	male or female (categorical)
Age	age of the client (numeric)
Tenure	the time with the bank as a client (numeric)
Balance	balance (numeric)
NumOfProducts	How many accounts, bank account affiliated products the person has (numeric)
HasCrCard	the person has a credit card or not (categorical)
IsActiveMember	active product user with transaction vs no activity or transaction (categorical)
EstimatedSalary	estimated salary income or each client (numeric)
Exited	attrition, Did they leave the bank after all? Yes (1), No (0)

The variable to be predicted :

- Exited Yes (1)— has the client churned? (binary: “1”, means “Yes”, “0” means “No”)

Mathematics and Description of the Method

Machine Learning Methods:

- **Logistic Regression:**
  Logistic regression (LR) is a statistical method similar to linear regression since LR finds an equation that predicts an outcome for a binary variable, Y, from one or more response

variables, X. To predict class it uses the log odds ratio rather than probabilities and an iterative maximum likelihood method rather than a least squares to fit the final model.

- **K Nearest Neighbors:**
  K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions).
- **Support Vector Machine:**
  The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

Artificial Neural Network:

Artificial Neural Networks or ANNs have three layers that are interconnected. The first layer consists of input neurons. Those neurons send data on to the second layer, which in turn sends the output neurons to the third layer. ANNs are considered non-linear statistical data modeling tools where the complex relationships between inputs and outputs are modeled or patterns are found.

$$Y = f(x)$$

$$\hat{Y} = \hat{f}(x)$$

Here x = (x1, x2,...,xd) which are the columns or features of our dataset explored above.

So the job of the ANN is to estimate f(x) as closely as possible.

If we partition the input space I_p into 2 sets or classes, P_1 and P_2, then all observations tagged 1 are in P_1 (exited) and the rest are in P_2.

$$\bigcup_{j=1}^{k} p_j = I^p$$

Thus we have the following labeling rule:
f(x) = j if x belongs to P_j(equation)

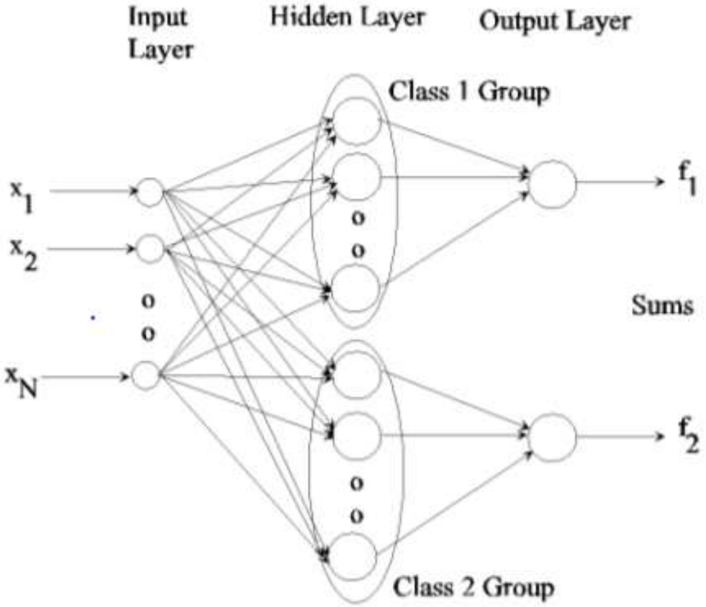
$$G_n(x) = \sum_{j=1}^{n} \alpha_j \sigma(\omega_1 x_1 + \cdots + \omega_p x_p + \theta_j)$$

$G_n(x)$ converges to the actual F belongs to C $[0,1]^p$ in the max norm if sigma be a real valued continuous sigmoidal function.
So f(x) can be partitioned into D(which is approximated accurately) and D^c (which couldn't be approximated well) and all observations fall in one of these 2 categories.

Input Layer	Hidden Layer 1	Hidden Layer 2	Hidden Layer 3	Output Layer 2
• Input data	• 6 neurons	• 6 neurons	• **4 neurons**	• 1 neurons

Probabilistic Neural Network:

A **probabilistic neural network (PNN)** is a feed-forward neural network, which is widely used in classification and pattern recognition problems. It was derived from Bayesian networks.
A probabilistic neural network (PNN) has 4 layers of nodes. The figure below displays the architecture for the PNN for our dataset that recognizes K = 2 classes(excited and not excited).



$X_1, X_2, X_3, \ldots X_d$ are independent variables and let Y = 1,2,3,...,k. (k are the number of classes)
Probability that x comes from bucket 'j' in Y,

$$\begin{aligned} p(j) &:= P(Y=j \mid X = x) \\ &= P(Y=j , X =x) / P(X=x) \\ &= f_{XY}(x,j) / f_X(x) \\ &= 1/c\,(f_{XY}(x,j)) \end{aligned}$$

The final predicted label for X= x will be,

$$\arg_{1 \leqslant j \leqslant k}^{\max} p(j)$$

Input Layer	Pattern Layer	Summation Layer	Output Layer
● Input all variables (10+1) = 11	● Number of neurons = (n*k) = 20,000 ● x_i are restricted to class j , $p - i = exp^{-Di^2/2\sigma^2}$	● Input : 2 neurons. ● Output: SD(j) $\sum_{i=1}^{n} e^{-D_j^2/2\sigma^2}$	● Input:1 neuron ● Output: argmax p(j)

RESULTS

Results from exploratory data analysis:

We note the following from categorical variables distribution plots:

- Majority of the data is from persons from France. However, the proportion of churned customers is 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.
- The proportion of female customers churning is also greater than that of male customers
- Interestingly, the majority of the customers that churned are those with credit cards. Given that the majority of the customers have credit cards could prove this to be just a coincidence.
- Unsurprisingly the inactive members have a greater churn. Worryingly is that the overall proportion of inactive members is quite high suggesting that the bank

may need a program implemented to turn this group to active customers as this will definitely have a positive impact on the customer churn.

We note the following from the box-plot continuous variables:

- There is no significant difference in the credit score distribution between retained and churned customers.
- 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
- With regard to the 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.
- Worryingly, the bank is losing customers with significant bank balances which are likely to hit their available capital for lending.
- Neither the product nor the salary has a significant effect on the likelihood to churn.

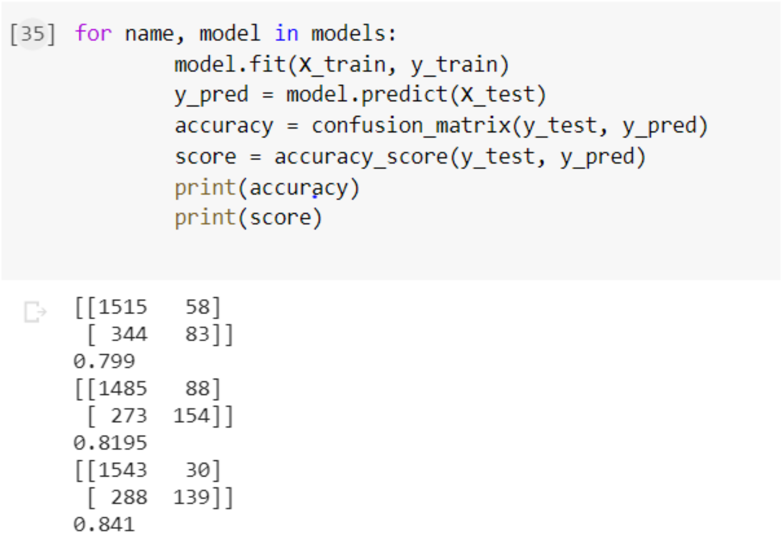
Results from building the models:

We used ***confusion matrix*** as the metric to measure accuracy in both cases.

We get similar accuracies for ANN and PNN. So, both perform similarly for our data.

Our results are as follows:

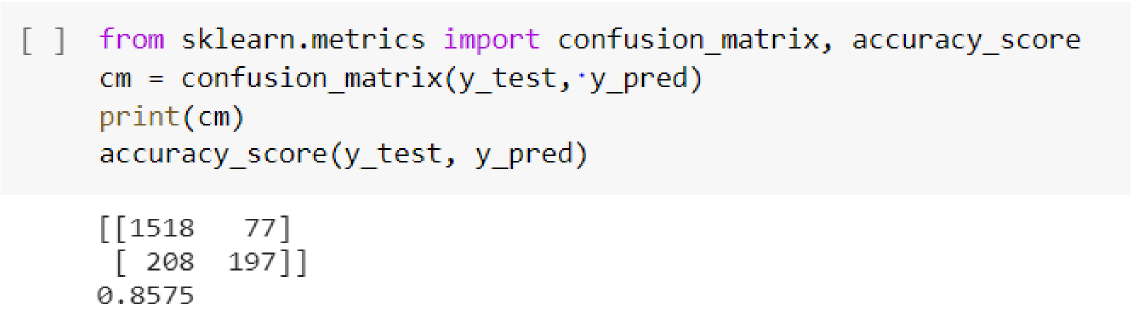
- ML models gave the following accuracies:
  The accuracies are of logistic regression **(79.9%)**, k nearest neighbors **(82.5%)** and support vector machine(**84.25)** respectively:



```
[35] for name, model in models:
             model.fit(X_train, y_train)
             y_pred = model.predict(X_test)
             accuracy = confusion_matrix(y_test, y_pred)
             score = accuracy_score(y_test, y_pred)
             print(accuracy)
             print(score)
```

```
[[1515   58]
 [ 344   83]]
0.799
[[1485   88]
 [ 273  154]]
0.8195
[[1543   30]
 [ 288  139]]
0.841
```

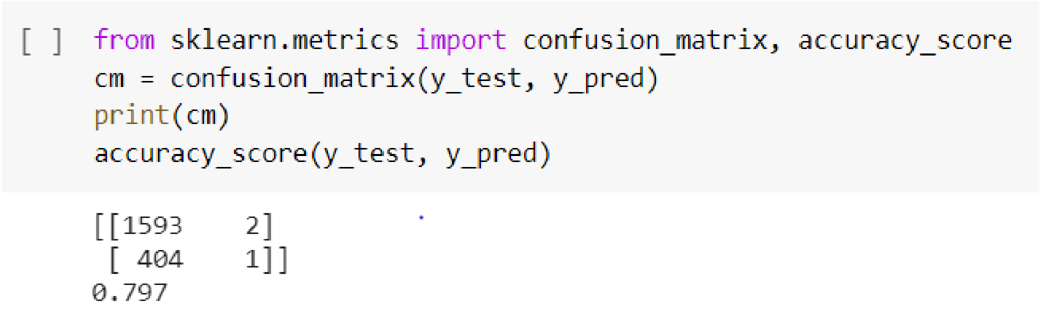
- ANN gives **86.19%** train accuracy and **85.75%** test accuracy without cross-validation and regularization.



```
[ ]  from sklearn.metrics import confusion_matrix, accuracy_score
     cm = confusion_matrix(y_test, y_pred)
     print(cm)
     accuracy_score(y_test, y_pred)
```

```
[[1518   77]
 [ 208  197]]
0.8575
```

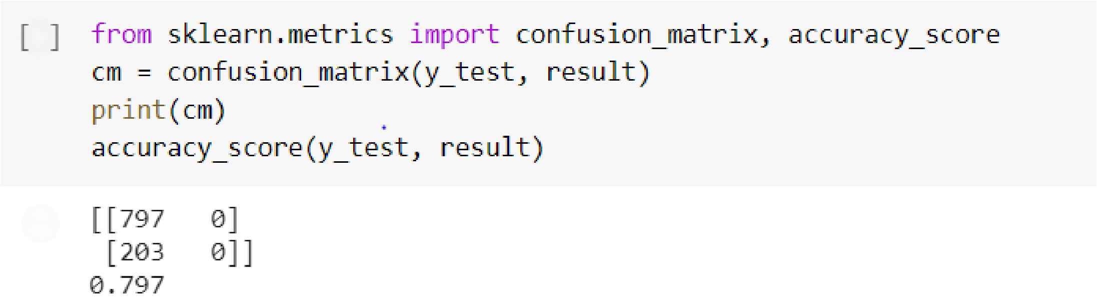
- ANN gives **80.3%** train accuracy and **79.7%** test accuracy with cross-validation and regularization.



```
[ ]  from sklearn.metrics import confusion_matrix, accuracy_score
     cm = confusion_matrix(y_test, y_pred)
     print(cm)
     accuracy_score(y_test, y_pred)
```

```
[[1593    2]
 [ 404    1]]
0.797
```

- PNN gives **79.7%** for std 1.



```
from sklearn.metrics import confusion_matrix, accuracy_score
cm = confusion_matrix(y_test, result)
print(cm)
accuracy_score(y_test, result)
```

```
[[797   0]
 [203   0]]
0.797
```