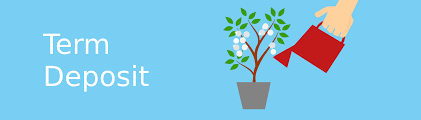
**BANK TERM DEPOSIT SCHEME PREDICTIVE MODEL**

****

**PROBLEM STATEMENT**

The banks are moving with the pace of technology and incorporating different techniques to get the clients on-board. There are multiple marketing techniques in the market different banks are resorting to get people involved into different banking schemes. One such technique is phone calling the clients, getting their details and letting them know about the different schemes. It might require multiple calls to the same client to figure out if the client will be on-board or not. The purpose of the hackathon is to build a model which would predict if the client will say ‘yes’ or ‘no’ for the scheme. The scheme in question is term deposit and is the same for all the clients. If the client gets on-board, it is denoted with ‘yes’ and if he does not, it is denoted with ‘no’.

A Term deposit is a deposit that a bank or a financial institution offers with a fixed rate (often better than just opening deposit account) in which your money will be returned back at a specific maturity time.

**Tools/Technology Used:-**

1. **Scikit-learn**
2. **Pandas**
3. **Numpy**
4. **Matplotlib**
5. **Jupyter Notebook**
6. **Seaborn**
7. **Sklearn**

**Technical Architecture:-**

1. **Data Exploration**
2. **Data Preprocessing**
3. **Feature Engineering**
4. **Splitting the dataframe**
5. **Model building**
6. **Training**
7. **Evaluation**
8. **Prediction**

**Market need to solve the business problem**

Marketing to potential clients has always been a crucial challenge in attaining success for banking institutions. It’s not a surprise that banks usually deploy mediums such as social media, customer service, digital media and strategic partnerships to reach out to customers. But how can banks market to a specific location, demographic, and society with increased accuracy? With the inception of machine learning - reaching out to specific groups of people have been revolutionized by using data and analytics to provide detailed strategies to inform banks which customers are more likely to subscribe to a financial product.

Data Description:-

The data set consists of the 21 attributes along with their values. The term deposit is denoted with variable y. The data can be understood in the 4 parts:

1. Bank client data attributes

2. Related with the last contact of the current campaign attributes

3. Other Attributes

4. Social and Economic Context Attributes

**DATA PREPROCESSING**

The preprocessing done in the notebook consists of the following steps:

1. Checking for the null values.
2. Dropping any duplicates in the data
3. Find Missing Values
4. Find Features with one value
5. *Converting the target variables into 0s and 1s*

FEATURE ENGINEERING

Feature engineering refers to a process of selecting and transforming variables/features in your dataset when creating a predictive modelusing machine learning. After splitting the dataset, convert the categorical variable into numeric because it works with only numeric features.

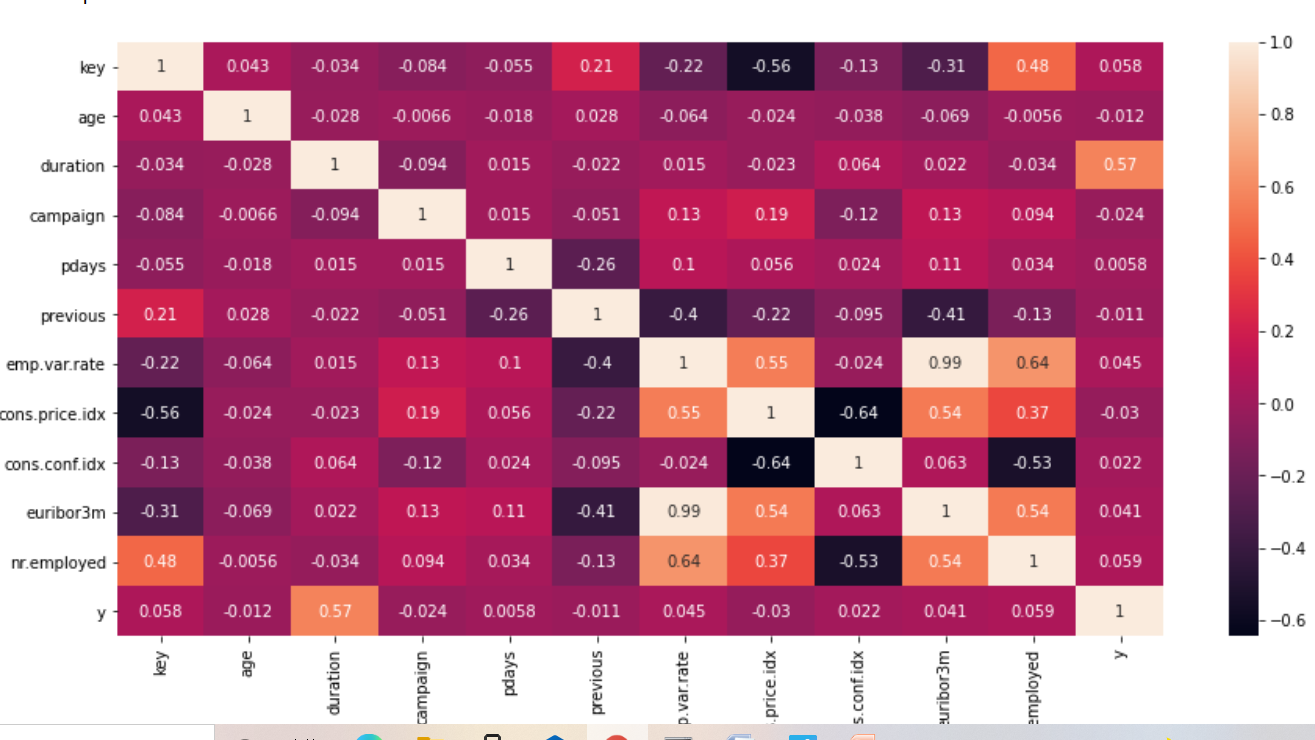
Label Encoding:  converting the labels into a numeric form so as to convert them into the machine-readable form. Machine learning algorithms can then decide in a better way how those labels must be operated.

Feature engineering involves the following steps:-

1. Drop unwanted Features
2. Handle Categorical Features
3. Handle Feature Scaling
4. Remove Outlier

FEATURE SELECTION:

It is the process of reducing the number of input variables when developing a predictive model. It is desirable to reduce the number of input variables to both reduce the computational cost of modeling and, in some cases, to improve the performance of the model. In this case few columns such as pdays, previous etc has low correlation with target y hence we did not selected those columns for model building.



SPLITTING THE DATAFRAME

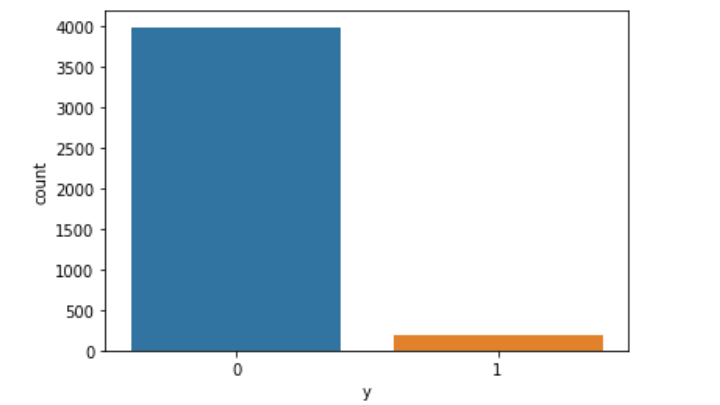
Before making the model, split the data set into the train data set with 70% and the validation data set with 30%. Train the model on 70 % of the training dataset, test the model on 30% of the validation dataset, and then predict the target variable using the trained model.

Converting the train into X and y so that we can pass it onto train\_test\_split function.

X --> contains the data frame without the target i.e. y.

y --> contains only the target value y.

**Handling the imbalanced dataset**

****

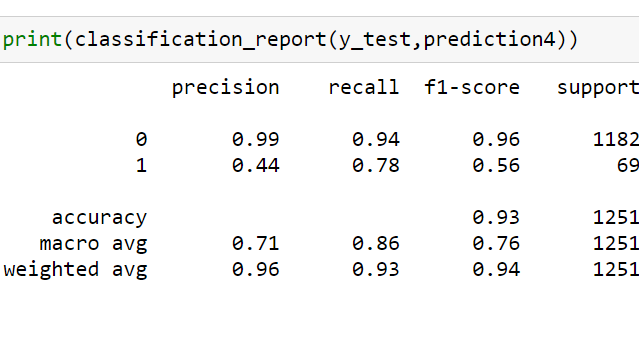
A balanced dataset is a dataset where each output class (or target class) is represented by the same number of input samples. Oversampling is a technique which increases the number of samples of the smallest class up to the size of the biggest class. This is done by generating synthetic samples. Different techniques can be applied to oversample a class. I will use SMOTE to balance the dataset .

Model Building:

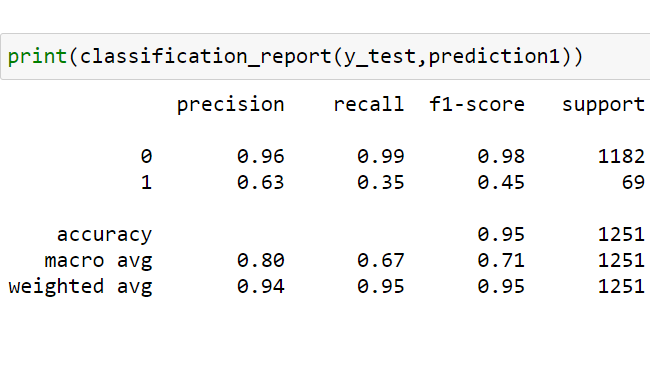
We will build different models and then on the basis of the evaluation metrics we will decide the best model. The different evaluation metrics for the models are accuracy, recall, F1 score and precision.

Classification Algorithms :

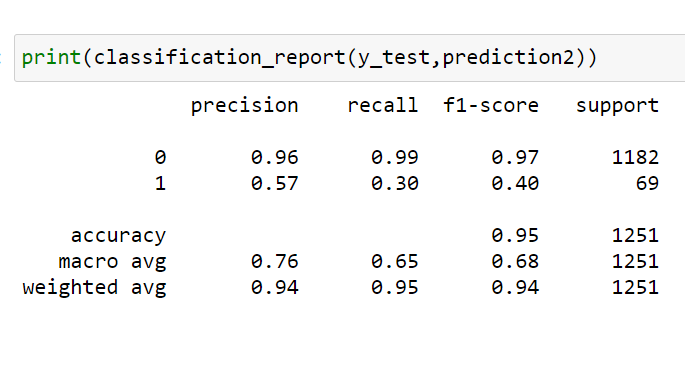
Logistic Regression - Logistic regression is one of the classification algorithms. Logistic regression is the statistical model for classifying the classes. Logistic regression classifies binary labels like pass\fail. Logistic regression is used to predict the categories based on the threshold value.



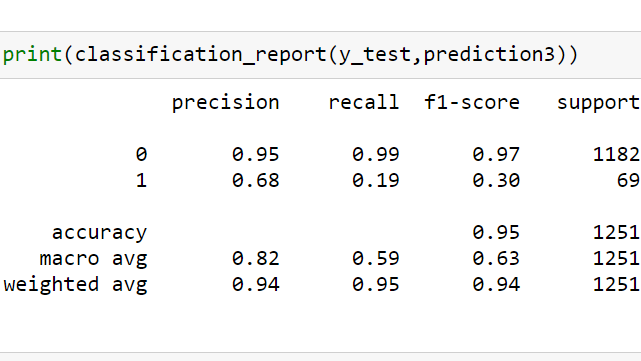
Support Vector Machine - A support vector machine is applied for both classification and regression problems. The main goal of SVM is to divide the datasets into classes to find a maximum marginal hyper plane.



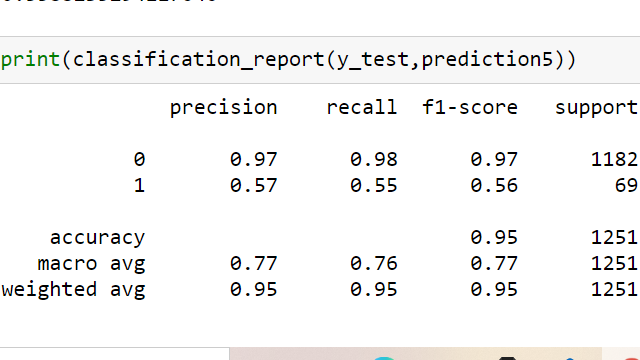
K-Nearest Neighbors - K Nearest Neighbors looks at the k closest data points and probability sample that has positive labels. It is easy to implement, and you don't need an assumption for the data structure. KNN is also good for multivariate analysis.



Random Forest Classifier - Random forest works like a decision tree algorithm but it performs various decision tree analysis on the dataset as a whole. That is, it is the bigger version of the decision tree; a forest is bigger than a tree, you can think of it that way. Random forest takes random samples of trees and it works faster than the decision tree algorithm.



Decision Tree Classifier - Decision trees works through the data to decide if one action occurs, what will then be the result of a "yes" and a "no". It works each data making the decision of which path to take based on the answer. Because of this decision making process, this algorithm has no assumptions about the structure of the data, but instead decides on the path to take through each decision the algorithm performs.



Evaluation Metrics of Classification Models :-

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.NO | Models | Accuracy | Precision | Recall | F1-Score |
| 1. | Logistic Reg | 93.28 | 43.90 | 78.26 | 56.25 |
| 2. | SVM | 95.28 | 63.15 | 34.78 | 44.85 |
| 3. | KNN | 94.88 | 30.43 | 30.43 | 39.62 |
| 4. | Decision Tree | 95.20 | 56.71 | 55.07 | 55.88 |
| 5. | Random Forest | 94.80 | 60 | 17.39 | 26.96 |

Conclusion:-

In this study, five algorithms were applied viz., Logistic Regression, Decision Tree, Random Forest, KNN and Support Vector Machine. Among these five algorithms, the Logistic Regression Classifier has accuracy, i.e., 93.28%, and Recall scores are 78.26%.