**AZUBI TEST**

**PROJECT**  
The project is aimed at predicting a bank Term Deposit subscription(y=yes/no) by building a machine learning model to predict.

**DATA LOADING & DATA EXPLORATION**

We focused on 2 major files

* bank\_train.csv (train set)
* bank\_test.csv (test set)

We used .info(), .describe() to understand the structure of the data we were working with. We identified numerical and categorical features.

**CATEGORICAL SUMMARY**

* From the data, it is seen that most clients did not sign up for the term deposit. About 39,922 representing 88.3% of the customers did not sign up for the term deposit.
* 44,396 customers had no credit default.
* There were also some imbalances that were identified

**NUMERICAL SUMMARY**

* Some columns had very high values and showed the possibility of outliers.
* Under the previous column, it is seen that most clients have previously not been contacted.
* Duration details indicates that some of the calls lasted for quite a long time.

We created some visual to understand distributions (e.g., age, balance, campaign). Used boxplots to detect outliers.

**FEATURE ENGINEERING**

This involves the creation of 3 new columns ie. was\_previously\_contacted, Age\_group, Balance\_Group. We also dropped a few columns like the age and pdays because they were not going to be needed as they were used to create new columns.

**ENCODING CATEGORICAL VARIABLES**

We proceeded to encode categorical variables to make the data types integers. As we can only train with integers as the datatype.

Went through the same cleaning and exploratory analysis for the test dataset. We also dropped same columns we dropped for train dataset.

**Feature Selection**

We dropped the column y for the X\_train and assigned Y to the Y\_train.

**Class Weight**Adjusted the weights of the variables for column Y. This was to increase the weight of yes\_subscriptions.

**Evaluating models**

The models were trained with logistic regression, random forest and XGBoost models.

From this, it is seen that the best performing model is the XGBoost model. it has a good balance between precision and recall. Meaning it is capable of predicting both yes and nos and allows you to get most people who are likely to subscribe.

Weighted class: The data has too many people not subscribing which is not ideal since it will teach that machine to predict more nos than yes. Hence, we made one class have more weight or importance so that the model pays attention to it. We made the yes 7 times more important to match up the nos. and trained the model with those weights.