

MICRO-CREDIT DEFAULTER MODEL

Submitted by:

SMITA DHADHAL

**ACKNOWLEDGMENT**

The data has been provided by FlipRobo Technologies.

**INTRODUCTION**

This project is being performed for a fixed wireless telecommunications network provider. The company is collaborating with an MFI to provide micro-credit on mobile balances to be paid back in 5 days. The Consumer is believed to be defaulter if he deviates from the path of paying back the loaned amount within the time duration of 5 days. For the loan amount of 5 (in Indonesian Rupiah), payback amount should be 6 (in Indonesian Rupiah), while, for the loan amount of 10 (in Indonesian Rupiah), the payback amount should be 12 (in Indonesian Rupiah).

In order to improve the selection of customers for the credit, the client wants a predictive machine learning model that could help them in further investment and improvement in selection of customers.

**Analytical Problem Framing**

**Key Inputs:**

Below are the key inputs :

* Input dataset for classification analysis
* Test data proportion: Between 0 to 1, default 0.3 (or 30%)
* Normalization: 0 - No Normalization, 1 - Min-max scaling, 2 - Z-score scaling
* List of model objects to test
* Number of folds for grid-search (hyper-parameter tuning)
* Scoring criteria to determine the best model
* Flag to see the level of detail on the terminal during model fit
* Hyper-parameter library

### General Execution Steps:

After taking these inputs, the following actions are performed for each model:

* Forward feature selection
* Normalization
* Grid search for hyper-parameter tuning
* Metric calculation for the best model

**Model/s Development and Evaluation**

**Models:**

* Logistic Regression
* Ridge Classifier
* Decision Tree Classifier
* Random Forest
* Bagging (Using decision tree by default)
* Gradient boosting
* XGBoost

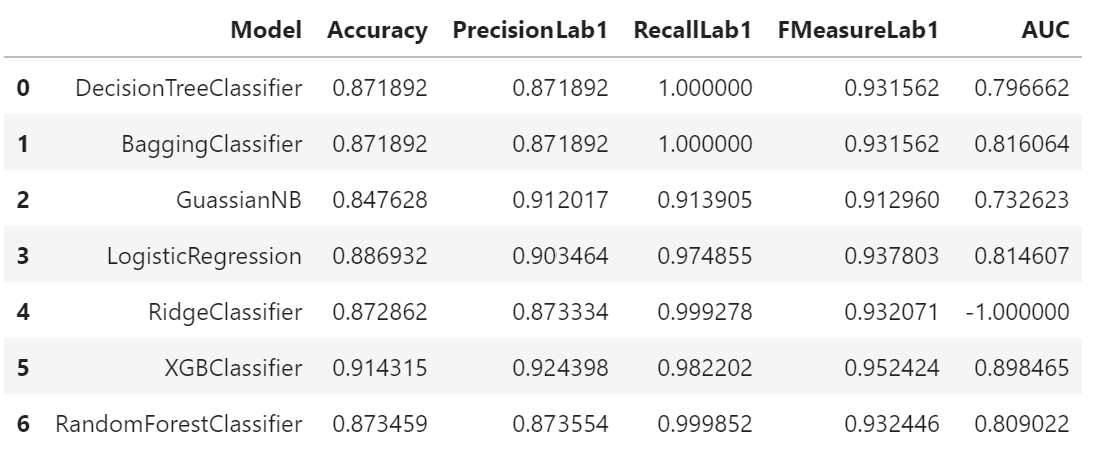
**Output:**

A pandas data frame 'results' is created which provides the following metrics for each model:

* Model name
* Model accuracy
* Precision and Recall
* F-measure
* The area under the receiver operating characteristics curve (AUC)

**CONCLUSION**

**Results:**

****

By going through the results we can compare all the metrics and we can come on a conclusion that **XGBClassifier** can be the best model in this case. Though **LogisticRegression** can also perform well.