Work Report Technocolabs Software

Name: **Mudit Vyas** Date of Internship: **1st Oct 2021 - 15th November 2021**

Work and Position: ML Developer Internship, **Team Leader – (Team A)**

- **Aim:** The goal of the project is to identify the best classification model in terms of accuracy and performance for predicting the bankruptcy of corporations using various statistical forecasting techniques.
- **Blueprint for project:**
  - **Link for Blueprint:** https://drive.google.com/file/d/1srXW6xDM1xGaQO2gWtwOoXjQKSzyyPIF/view?usp=sharing
- **Steps involved in project:**
  - EDA
  - Model Development
  - Model Deployment

EDA

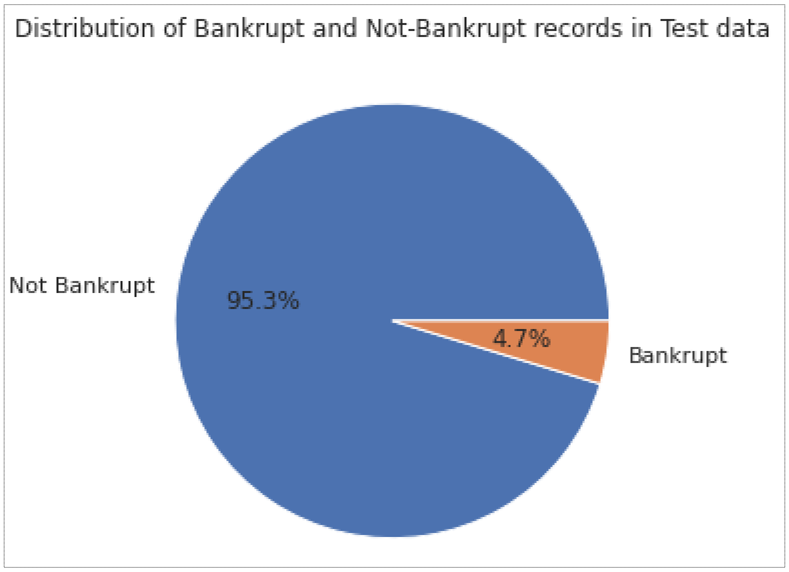
- *Step 1:* Importing and organizing the data
  - Convert the types of the columns for the features to float
  - Convert the class label types to integer
- *Step 2:* Data Analysis and Preprocessing
  - Missing Data Analysis
  - Data Imputation
    - EDA 3rd Year: Mean Imputation
    - EDA 2nd Year: K-Nearest Neighbor
    - EDA 4th Year: Expectation Maximization
    - EDA 1st Year: MICE (Multiple Imputation by Chained Equation)
    - EDA 5th Year: Forward Interpolation
- *Step 3:* Dealing with Imbalanced Data
  - Imbalanced classification is the problem of classification when there is an unequal distribution of classes in the training dataset.

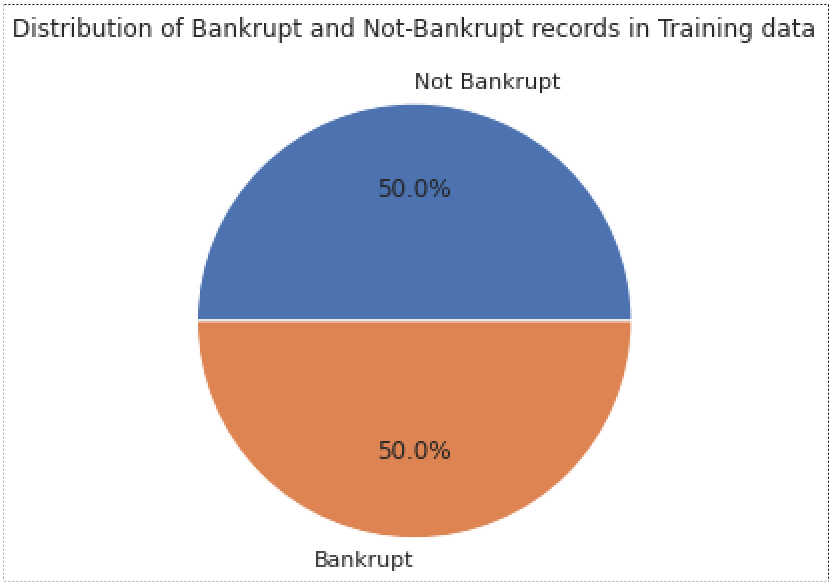
  - The distribution can vary from a slight bias to a severe imbalance where there is one example in the minority class for hundreds, thousands, or millions of examples in the majority class or classes.

- **Challenge of Imbalanced Classification**
  - **Slight Imbalance**. An imbalanced classification problem where the distribution of examples is uneven by a small amount in the training dataset (e.g. 4:6).
  - **Severe Imbalance**. An imbalanced classification problem where the distribution of examples is uneven by a large amount in the training dataset (e.g. 1:100 or more).

- Severe Imbalanced Training Data which implies directly training of model would impact Classification accuracy: precision, recall. F1 score even though achieving good test accuracy score.

- One approach to addressing imbalanced datasets is to oversample the minority class. The simplest approach involves duplicating examples in the minority class.
- Instead, new examples can be synthesized from the existing examples. This is a type of data augmentation for the minority class and is referred to as the **Synthetic Minority Oversampling Technique**, or **SMOTE** for short.



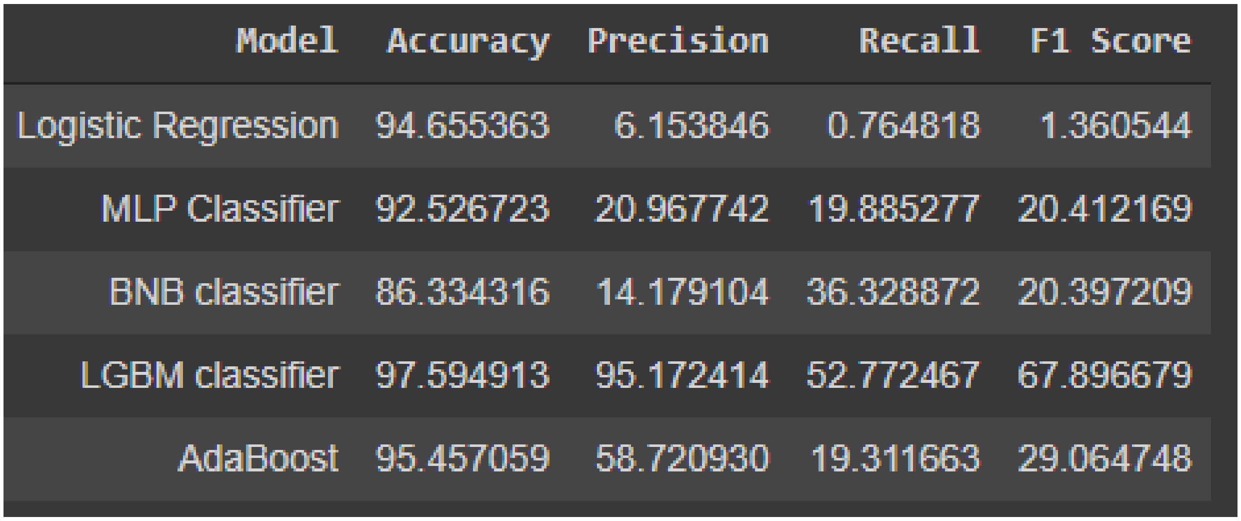


- Model Development

  - Main Model Worked and tried for development process:
    - Logistic Regression
    - MLP Classifier
    - Light Gradient Boosting Machine
    - Random Forest Classifier

  - Model Development Process starts by following order:
    - Obtaining Imputed Data
    - Feature Selection out of 64 attributes: selected top 20 attributes.
    - Feature selection refers to techniques that select a subset of the most relevant features (columns) for a dataset.
    - RFE works by searching for a subset of features by starting with all features in the training dataset and successfully removing features until the desired number remains.
      - Training and Testing Dataset: Using Stratify Train-Test Split

➢ Summary of Model Training with Imbalanced Data

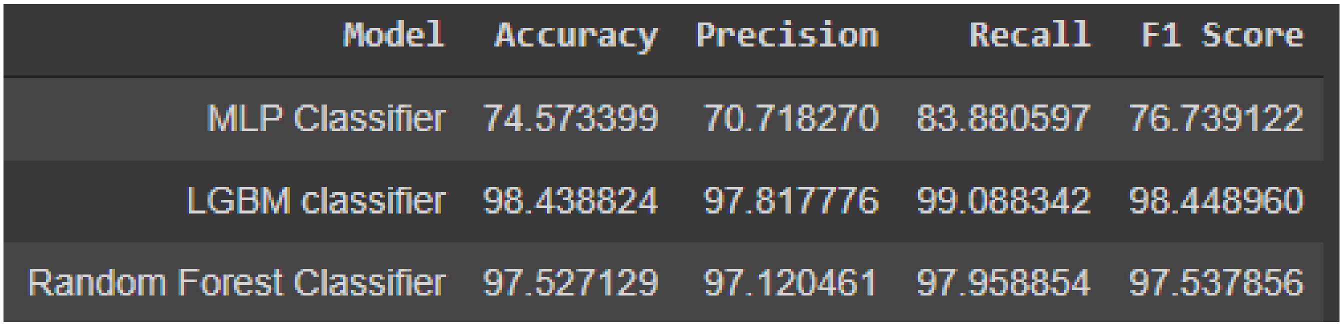


Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	94.655363	6.153846	0.764818	1.360544
MLP Classifier	92.526723	20.967742	19.885277	20.412169
BNB classifier	86.334316	14.179104	36.328872	20.397209
LGBM classifier	97.594913	95.172414	52.772467	67.896679
AdaBoost	95.457059	58.720930	19.311663	29.064748

- ➢ Good Classification Model has F1score, Precision and Recall closer to 100.
  - o Due to few samples of Minority Class (“Bankrupt”) classification report of Imbalanced Data is poor. So, it is essential to Oversample imbalanced data.

➢ Summary of Model Training with SMOTE Oversampled Data

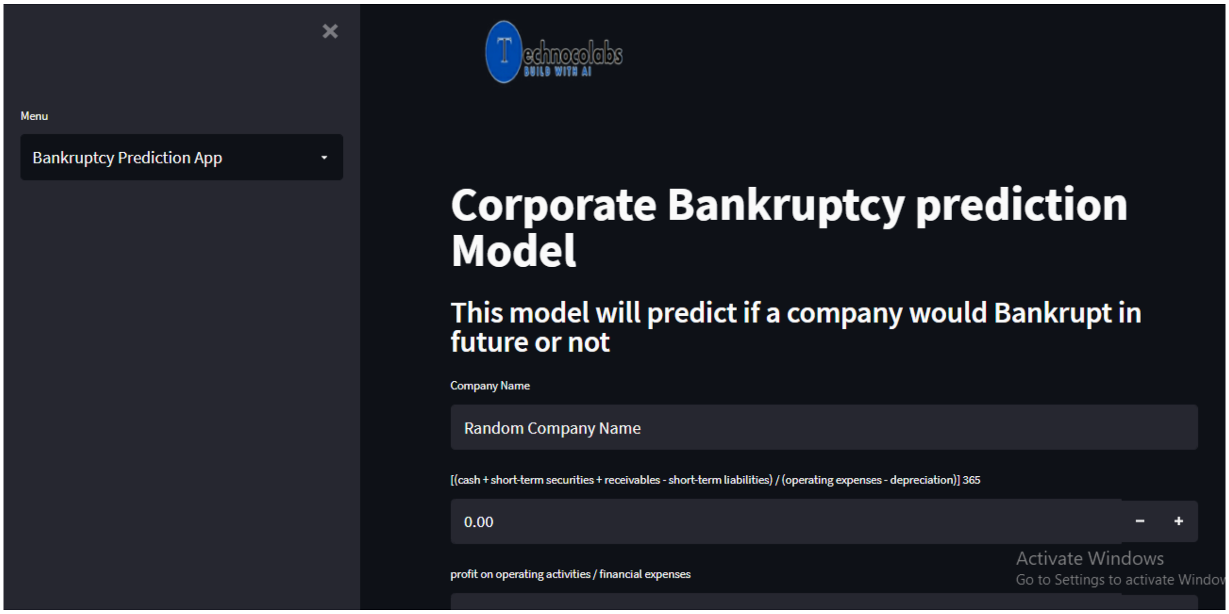
- o LGBM and Random Forest Classifier models show good classification report and can be suitable for real world application.
- o LR model due to oversampling fails and accuracy of model drops to 70%. MLP Classifier may be optimized using Keras / Tensor flow for better model optimization.



Model	Accuracy	Precision	Recall	F1 Score
MLP Classifier	74.573399	70.718270	83.880597	76.739122
LGBM classifier	98.438824	97.817776	99.088342	98.448960
Random Forest Classifier	97.527129	97.120461	97.958854	97.537856

➢ DEPLOYMENT OF ML MODEL

- **Model:** Random Forest Classifier
  - LGBM model cannot be able to deploy on heroku because of size issue.
- **Webapp built using:** Streamlit
- **Language:** Python
- **Application Deployed on:** Heroku
- **Website/App URL:** https://technocolab-app.herokuapp.com/
- For Deployment Top 12 Attributes are used for Streamlit App



Corporate Bankruptcy prediction Model

This model will predict if a company would Bankrupt in future or not

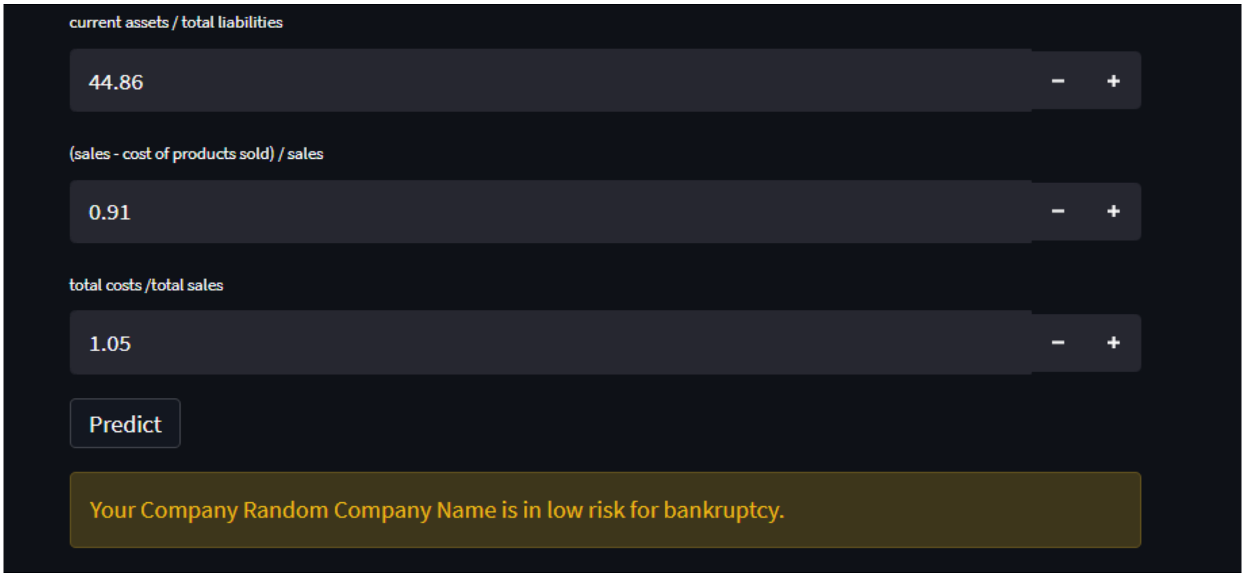
Company Name

Random Company Name

[(cash + short-term securities + receivables - short-term liabilities) / (operating expenses - depreciation)] 365

0.00

profit on operating activities / financial expenses



current assets / total liabilities

44.86

(sales - cost of products sold) / sales

0.91

total costs /total sales

1.05

Your Company Random Company Name is in low risk for bankruptcy.