Predicting the Loan Defaulter

Abstract

The main idea of this project was to develop a framework by using classification models to predict the loan defaulter in banking to minimise the risk of losing money while lending to customers. I worked with the data provided by Kaggle, and although the data was huge, I tried to achieve results for more than one model. To improve the models, I did an intensive EDA to focus more on the features most closely related to the target features of predicting loan defaulters.

Design

The project aims to identify patterns that predict if a client has difficulty paying their instalments which may be utilised for taking actions such as denying the loan, reducing the amount of loan, etc. Moreover, the goal is to understand the driving features behind loan default, in other words, the features which are strong indicators of default.

The Target feature is about whether a client has payment difficulties or NOT:

1: Client with payment difficulties: he/she had late payment more than X days. "Defaulter"

0: All other cases when the payment is paid on time. "Non-Defaulter"

Data

An available dataset was chosen from Kaggle [1] in order to perform this project. The dataset contains 307,511 rows with 122 features.

Algorithms

Feature Engineering:

- Converting the categorical features to relevant binary-numbers using dummy.
- Checking for columns that are contains more than 40% of missing values and delete them. (Total of 52 columns are there which have more than 40% null values and have been deleted).
- Selecting some valuable features which have strong correlation with the Target feature.

Models:

Logistic regression, k-nearest neighbors, decision tree and random forest classifiers were used, and the best classifier is the Random Forest with quite good F1 score.

Model Evaluation and Selection:

The entire training dataset of 307,500 records was split into 80/20 train vs. test, and all results were quantified over several evaluation performance metrics: accuracy, precision, recall and F1 score. the primarily focus on F1 score within the metrics, as it encompasses aspects of both precision and recall. Nonetheless, class weights were included to improve performance against F1 score and provide a better classification.

The best random forest classifier scores:

Accuracy 0.92, F1 0.88, precision 0.85, recall 0.92

Tools

Mainly, the python programing language is used to perform this project. In addition, various libraries are used as mentioned below. Nonetheless, the project result is presented using Tableau software for interactive visualizations.

- NumPy and Pandas for data manipulation.
- Scikit-learn for modelling.
- Matplotlib and Seaborn for plotting.

Results for 4 Models

Model	Function	Precision	Recall	Accuracy	F1
Logistic regression (LR)	**LogisticRegression**	**0.87**	**0.54**	**0.91**	**0.64**
K-nearest neighbours (K-NN)	**KNeighborsClassifier**	**0.86**	**0.91**	**0.92**	**0.84**
Decision Tree (DT)	**DecisionTreeClassifier**	**0.88**	**0.61**	**0.84**	**0.70**
Random Forest (RF)	**RandomForestClassifier**	**0.85**	**0.92**	**0.92**	**0.88**

References:

[1] Kaggle. Loan Defaulter Dataset. Retrieved from https://www.kaggle.com/gauravduttakiit/loan-defaulter