**Classification Write-up**

**Ed Lee**

DSML, July 2022

**Abstract**

Small business loans administered by SBA is one of the primary sources of funding for kickstarting small firms. I would like to analyze the records of SBA loan approvals and investigate which elements of a small business are likely to have a successful payoff (therefore ‘safe’ to approve), and which characteristics of a business hints at a risk of charge-off.

I have utilized several different classification libraries in order to build a prediction model to see if a certain business is likely not to pay off the loan. Among them, random forest with tuned hyperparameter displayed the best result, with model score of 0.938, ROC area of 0.961, and recall of 0.681.

**Design**

I’ve written a code that performs classification fitting with given classifier. As a starting point, simple logistic regression model was used as the ‘baseline’ model. The classification was performed on train dataset first, then validated with validation set. By comparing model metrics of each result, the best model was chosen. Finally, test model was created for the best model, and then its metrics were analyzed in order to compare how much did the model improve compared to the baseline model.

Each results was visualized for easier comparison.

**Data**

Dataset that contains records of SBA loan approvals was obtain via Kaggle(<https://www.kaggle.com/datasets/mirbektoktogaraev/should-this-loan-be-approved-or-denied>).

The original the dataset contains 899,164 entries with 27 different features. Not only did the data contains null values, among the entries were many rows with invalid values for their features (for example, a value of 'C’ was recorded in one of the features of which its value is supposed to be either 'Y’ or 'N')

Once data cleanup was completed, 610,172 entries remained, and 9 features were used for building models for this project.

Target feature is ‘MIS\_Status’, which is status of the loan, either paid in full or charged-off.

The 'true’ values (confusingly ‘charged-off’ in this case) were observed in the whole dataset at about 1:5 ratio. Therefore, while there was some degree of imbalance, it wasn't too severe.

**Algorithms**

1. Classifiers were the main workforce of the model.
2. For data cleaning and feature engineering class imbalance and checking unique values were used

**Tools**

* **Pandas -** EDA, data cleaning, data manipulation, feature engineering
* **Tableau –** for visualization
* **Scikit-learn** - Building classification models (LogReg, random forest, Naïve Bayes, AdaBoost), preprocessing data, evaluating model
* **XGBoost** - Building classification models (Xgboost classifier)
* **Imblearn** – testing for class imbalance

**Communication**

The result was summarized via Powerpoint. Rest of the projects (codes) will be uploaded to GitHub.