Classifying Bankruptcy Using XGBoost Algorithm

Richard Kim
Southern Methodist University

Joaquin Dominguez
Southern Methodist University

October 18, 2022

I. Introduction

Quantitative methods play an essential role in the process of evaluating companies within the finance industry. Although recent technological advancements have streamlined the process of financial analysis, the use of past information to predict future outcomes with the objective of benefiting from such knowledge has been used for thousands of years. Merchants traveling along the silk routes had used simple charts and price records of commodities to examine price trends. [1]. Today, we have access to a plethora of information that facilitates a more complex and precise financial analysis than those of the silk route merchants.

The focus of this paper will be on such analysis used for Emerging Markets. Generally, in this sector, it is simply necessary to be able to make an assessment on whether investment in a given company is worthwhile in order to succeed. To this end, due to regulations and internal status reports, plenty of relevant and valuable data can be used for this type of analysis. Given the financial interest at stake in the efficacy of this objective, several data sets have been made publicly available that represent the relevant factors that may aid in the prediction of bankruptcy for a given company. For this paper we used the 'Polish Companies Bankruptcy Data' data set collected from Emerging Markets Information Service (available through UCI Machine Learning Repository). The target variable, in this case, serves to classify whether a company ended up bankrupt given a 5 year time frame from when the data was collected. With respect to methodology, since this is ultimately a classification task, we explored the relative performance of Random Forest and XGBoost algorithms. Ultimately, as expected, the XGBoost model displayed superior performance. It is suggested that further research be done with alternative methods to those presented herein in the interest of increasing performance and scalability.

[1] Peter Scholtz, 2020

II. Methods

i. Data Preprocessing

The raw data consisted of 43,405 observations across 65 features. The target variable was a binary outcome identifying whether or not a given company went bankrupt. It is important to note that over 95% of cases in this data set (41,314) did not go bankrupt, introducing a large imbalance between classes in the outcome variable. There were 401 duplicate entries. After removing the duplicates, the data set comprised 43,004 observations and 65 potential explanatory features. We did include an additional feature for 'year,' outlining whether a particular observation was from year 1-5 of the data set.

All features, except the target variable, contained missing values ranging from 7 to 18836. In order to systematically address these missing values, we used the SimpleImputer func-

tion from SKLearn to impute via the median, since distributions were almost universally non-normal.

i.1 Assumptions of Random Forest

Data Type
Since Random Forest is a non-parametric algorithm, no formal distribution assumptions must be met. It does, however, require that the input data be continuous and the target variable discrete, which this data set does satisfy. Additionally, RF does not allow missing values, in contrast to XGBoost.

i.2 Assumptions of XGBoost

Variable Relationship
XGBoost may assume that integer values for each input variable have an ordinal relationship, which this data set does satisfy.

Missing Values
XGBoost assumes that data may have missing values. Since missing values had to be imputed in order to run Random Forest, we tested XGBoost with and without imputation and kept the model with best performance (imputated model).

ii. Modeling

In an effort to classify the companies that go bankrupt, we developed and applied an XGBoost algorithm to the provided data from Python's *xgboost* package. Given that this is a classification problem, the evaluation metric by which the hyperparameters were tuned was the log loss of the training data. The model was adjusted based on various combinations of learning rates, subsample ratios, and minimum child weight, and was determined using randomized search with cross-validation (CV) at 5 folds. Randomized search CV was conducted with the number of boosting rounds set to 500 and early stopping rounds set to 2.

The final model was trained with a learning rate of 0.1, subsample ratio at 0.85, and minimum child weight at 3. All other hyperparameters were kept at their default values. A random forest classifier was also modeled using *sklearn*'s *RandomForestClassifier* package to provide a baseline comparison for the XGBoost performance. Although both models were optimized for accuracy, performance on precision and recall are also presented in the results.

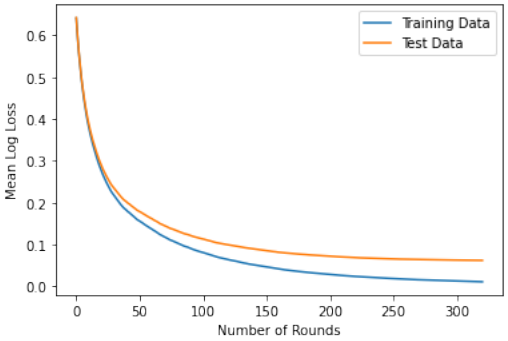

Figure 1: *Learning Rate Curve*

We can see from Figure 1 that the number of boosting rounds was sufficient for modeling the data, given that learning rate flattens out after about 200 rounds. We also note that the performance between the training and test data were relatively close, both log loss values dropping below 0.1.

III. Results

i. Accuracy, Precision, and Recall

For both the random forest classifier and XGBoost model, the precision and recall scores were very high for cases that did not go bankrupt (> 96%). Given their exceptionally high scores, the focus of our results on precision and recall are on the cases that did go bankrupt.

The random forest classifier achieved an overall accuracy of 96.16%. While the random forest model was almost always able to identify the companies that went bankrupt with 97% recall, it had a high Type I error rate, classifying

most outcomes as bankrupt. This is evident in its extremely poor precision score at 26%.

Random Forest Metrics		
	Precision	Recall
Not Bankrupt	100%	96%
Bankrupt	26%	97%

Table 1: *Table of random forest model's precision and recall scores.*

The XGBoost model achieved an overall accuracy of 98.13%. The model was able to identify most companies that truly went bankrupt with a recall at 73%. It also achieved a high precision score, given that 89% of the companies it classified as bankrupt were truly bankrupt.

XGBoost Metrics		
	Precision	Recall
Not Bankrupt	99%	99%
Bankrupt	89%	73%

Table 2: *Table of XGBoost model's precision and recall scores.*

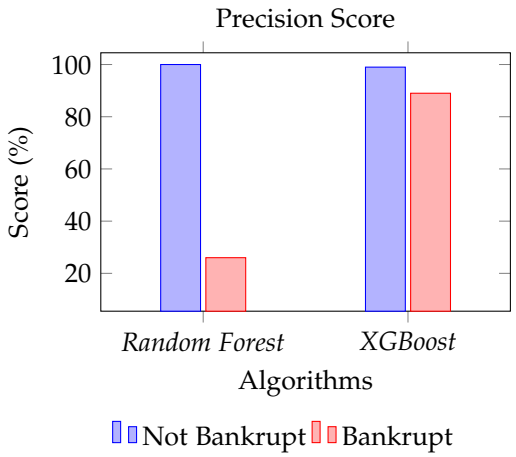

Figure 2: *Precision Metrics (Random Forest and XGBoost*

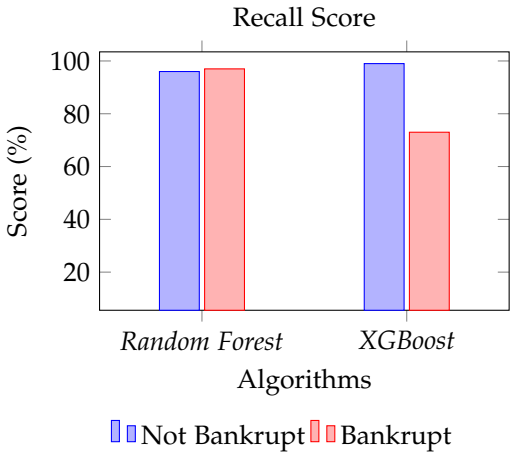

Figure 3: *Recall Metrics (Random Forest and XGBoost)*

IV. Conclusion

In this study, we aimed to develop a model that can classify companies that file for bankruptcy. The XGBoost algorithm we developed was able to achieve an accuracy slightly higher than that of a random forest classifier. Although the difference in overall accuracy was not significant, the precision and recall scores highlight the advantage and practical utility of implementing the XGBoost model.

The random forest classifier had a very low precision score at 26%, while the XGBoost model had a significantly higher score at 89%. This implies that, of all the positive cases identified by the random forest model, 26% of those cases were truly bankrupt, and 73% were incorrectly classified as bankrupt. This is highly problematic, given that predictions will help determine divestment of assets. The recall score at 97% means we are almost always identifying the companies that go bankrupt, but the low precision also means we would be divesting assets from a lot of companies that do not go bankrupt.

The Type I Error is much lower with the XGBoost model, boasting a precision score of 89%. This indicates that if the model predicts a company to go bankrupt, the prediction is reliably accurate. In exchange, the XGBoost model has a reduced recall score of 73%, which

means some companies that do go bankrupt may not get identified. Although this is much lower compared to the random forest model's 97% recall, it is important to note the balance between precision and recall.

The trade-off between precision and recall is crucial here, given that the goal is to make the decision to divest in companies going bankrupt. With the random forest model, its near perfect recall score would mean we would be divesting a lot of assets. However, paired with low precision, we would also be divesting a lot of assets that we may want to continue investing in. The XGBoost model's much higher precision at the expense of some recall will likely produce a more favorable outcome in practice.

The current study results indicate that the XGBoost model demonstrated better overall performance and has more utility in classifying companies that go bankrupt.

i. Code

Please refer to the attached Python notebook.

References

[Peter Scholtz, 2020] Scholtz, P. (2020). *The History of Quantitative Analysis*. Scholtz & Company. https://www.scholtzandco.com/insights/articles/quantitative-analysis/