“Loan Worthy or Not?”

Harnessing Analytics to Understand What Variables are Predictive of Credit Score Bracket

Brittany Nguyen, Dalal Al Ramadhan, Nikita Singh

15.072 Advanced Analytics Edge

Fall 2022

I. Problem Statement

Credit scores are numbers which assess the likelihood of any individual or business to repay loans in a timely manner. On an individual level, credit scores can span between 300 to 850 (worst to best rating, respectively). Within that range, a credit score between 580 and 660 is good, a score between 670 and 740 is great, and a score of 800 or more is excellent. The most common use of credit scores is to assess whether the subject will repay debts, subsequently driving decisions behind who will receive credit cards, loans, and mortgages. The data used for this project includes the usual suspects for predicting credit score: occupation, annual income, monthly salary, number of bank accounts, number of credit cards, and more.

Throughout the course of this project, we satisfy two main goals using a variety of analytical techniques in the R programming language. The first is seeing whether an individual's credit score bracket (defined in the data by levels "Good," "Standard," and "Poor") can be predicted using different factors regarding economic and loan status. Secondly, we assess what variables are most important to each model in predicting credit score and understand the impacts of including those variables within each model. All analytical techniques employed were classification-based models (as we were predicting credit score bracket, not the score itself). The models selected were Multinomial Logistic Regression, CART Decision Trees, Random Forest, XGBoost, and a Stacked Ensemble Multinomial Logistic Regression Model (containing the Logistic Regression, CART, and Random Forest models).

II. Data

The data obtained for this project comes from Kaggle. Both a training and testing set were provided, but only the training set was used as it had 100,000 rows and contained the dependent variable "Credit_Score." The testing set provided was not used since it did not contain the dependent variable. An initial glance at the data showed that there were 28 individual-level variables to use in the prediction of the credit score bracket. Each row represented one individual and their respective credit information. Some independent variables such as "Month," "Age," and "Annual_Income" were read in as type chr while others such as "Interest_Rate" and "Num_Credit_Card" were read in as type int. Clearly the data needed to be cleaned and type casting was performed as one of the first data cleaning steps to be explained in Section III. The dependent variable, "Credit_Score," was read in as type chr as well with levels "Good," "Standard," and "Poor" for each observation.

III. Methodology

A. Data Cleaning

Once the data was read in, data cleaning commenced. All variables not already in an appropriate class were cast using the as.factor() or as.numeric() features in R. Secondly, the data frame was analyzed for missingness. Each column was analyzed such that if missing values existed, they were dropped using the drop_na() functionality. Additionally, all variables were cleaned so that outliers were removed. A distribution-based approach was used to do this. The quantile() function was utilized in order to obtain the 25^{th} and 97.5^{th} percentiles of that given variable, and all values for that variable were subset to be within that range. This ensured that all

variables had realistic values. Odd characters were removed from certain variables, including "_". Social security number was not deemed a useful predictor, and thus that column was dropped. Finally, the "Payment_Behaviour" column which contained levels "Low_spent_Small_value_payments" and "High_spent_Medium_value_payments" was split into two columns: "Spending behaviour" and "Payment value behaviour."

As some of the predictive variables of interest were categorical, one-hot encoding was necessary before the models were constructed. "Month," "Occupation," "Credit_Mix," "Payment_of_Min_Amount," "Spending behaviour" and "Payment value behaviour" were turned into dummy columns for their different levels and the original columns were dropped. The new dummy set was appended to the training data frame. Not all models required the use of one-hot encoded variables; for instance, Logistic Regression, CART, and Random Forest employed the original values for these categorical predictors while XGBoost used the encoded ones.

B. Train Test Split

The table() function within R provided a way to check for any class imbalance prior to splitting the data into training and test sets. The class difference was amended by over sampling the "Good" and "Poor" classes to match that of the majority class. Ultimately, all three levels contained 12244 data points. Stratified Splitting was then chosen as the technique to separate the training data frame into train and test sets. This technique preserves the distribution of data points across both sets, such that the training set does not keep all the "Good" data points and the test set contains only "Standard" and "Poor" (as one such example).

C. Model Selection

Once the data cleaning and splitting were performed, the modelling process began. The first model run was a baseline multinomial logistic regression model. The next three chosen models were CART, Random Forest, and XGBoost. Finally, we decided to try out an ensemble stacked multinomial regression model for comparison to the baseline logistic regression model. The concept of "stacking" models is to utilize the predictions of several individual models (Logistic Regression, CART, and Random Forest in this project) to increase the robustness of predictions. These five models were compared based on accuracy and AUC. Additionally, the variable importance scores for each model were outputted, so that we could qualitatively assess what factors were deemed most vital towards predicting credit score for each algorithm.

IV. Results: The "Best" Model to Predict Credit Score and Important Factors

The results from running all five models are summarized in the chart below. Both a "Baseline" model and "Improved" model were constructed for CART and Random Forest. In each case, the baseline model used most variables from within the dataset, namely the ones which produced relatively high accuracy and AUC metrics when used in combination with one another. Meanwhile, the improved model used only the top 7 features as predictors (as determined by variable importance).

	Logistic Regression	CART Baseline	CART Improved	Random Forest Baseline	Random Forest Improved	**XGBoost**	Stacked Model
Accuracy	60.51%	62.59%	63.40%	80.24%	79.70%	**90.36%**	79.49%
AUC	0.6823	0.6806	0.6871	0.8180	0.8040	**0.9496**	0.7950

Table 1: Accuracy and AUC Values for all 7 Models

As evident in the results above, the "best" model according to out of sample performance on the testing data was XGBoost as it had the highest accuracy (90.36%) and AUC (0.9496). The variable importance from each model, namely which variables each model deemed most important in predicting credit score, vary. For CART, "Interest_Rate," "Num_Credit_Card," "Delay_from_due_date," "Outstanding_Debt," and "Num_Bank_Accounts" were among the top 7 predictors. For Random Forest, "Outstanding_Debt," "Interest_Rate," "Num_Credit_Card," and "Delay_from_due_date" were similarly shared important predictors.

Both CART trees for the baseline and improved model are also included within Figures 1 and 2. It is important to note that the hierarchy of variables which the trees chose to split on did not completely match the top predictors outputted from variable importance, as this is a normal phenomenon. For instance, the tree for the improved CART model first splits upon the outstanding debt variable, then moves on to use interest rate and delay from due date as defining characteristics (number of credit cards did not show up as one of the first two splits even though it is ranked highly in variable importance). Additionally, although we considered removing the "Interest_Rate" variable from the variable importance analysis, we decided against doing so. Including it only had a small impact on accuracy, and its variable importance was not significantly higher than other variables within the model. Thus, we decided it was not an overbearing factor in the analysis.

Finally, multiclass AUC curves for the baseline CART, Random Forest, and XGBoost models are included for interpretability in Figures 3, 4, and 5. While normally AUC is computed for binary classification problems by measuring how well models can differentiate between two classes, this AUC curve includes 3 plots as there are 3 pairwise relationships in the dependent variable used in this study: "Good," "Standard," and "Poor." Thus, it was necessary to measure the model's prediction of Good versus Standard, Good versus Poor, and Standard versus Poor credit score brackets. Since with AUC curves a more maximal area under the curve indicates a more predictive model, the strongest performance could be seen from the XGBoost curves (highest area under each of the graphs), followed by the Random Forest curves, and finally the CART curves (lowest area under each of the graphs).

V. Conclusions and Recommendations

In conclusion, the models mostly performed as expected. Boosted models including XGBoost tend to perform very strongly, as indicated by high accuracy and AUC values. However, these models are complex and tend to lack interpretability. Meanwhile, models employing a tree structure (CART and Random Forest) are more understandable, especially by non-technical audiences. Thus, while XGBoost is the best performing model, it may not be the best model overall when considering the tradeoff between performance and interpretability. Perhaps the Random Forest model would be more optimal as it has a higher interpretability with a small decrease in accuracy (79.70%) and AUC (0.8040). One extension of using tree-based models was learned in our machine learning class for instance, where Optimal Classification Trees were used to interpret the results of K-Means clustering. This is just one of many examples of harnessing the interpretability of trees to better understand performance.

In comparing the baseline models to the "improved" models, increases in the performance metrics was also observed. For CART, the accuracy increased slightly up to ~63% while the AUC also increased in the thousandth decimal place (very small difference). Small decreases were observed for the improved model for Random Forest. The logic behind the "improved" models was that only including variables ranked highly from the variable importance would result in higher predictive power. However, even though this was the case for CART, it was not for Random Forest. Future analysis would play with the combination of variables included in the improved Random Forest variable to perhaps include upwards of 10 (instead of 7) of the most important predictors.

Another conclusion that can be made relates to the output of the stacking ensemble multinomial model. Using predictor columns from the individual logistic regression model, CART, and Random Forest, the model is constructed to make more robust predictions than any model would alone. However, this was the opposite result of what occurred in our analysis. The stacked ensemble model had a relatively low accuracy of 79.49% and an AUC of 0.7950. These performance metrics were slightly worse than those of the improved Random Forest model. One potential source for this surprising result is collinearity between the three predictor columns used in the stacked model (the predictions from Logistic Regression, CART, and Random Forest). Further analysis would involve altering these 3 prediction columns to reduce collinearity between them, so that improvements in accuracy and AUC could be seen when stacking.

Thus, both questions we had set out to answer at the start of the study were effectively answered. The first involved finding which model had the highest performance in predicting credit score bracket. Companies extending loans and mortgages should use XGBoost for the highest performance but use Random Forest if looking for a balance between good performance and interpretability. Secondly, we wondered which variables were most important when predicting credit score bracket. For instance, if a consumer is looking for a loan, which aspects of their portfolio should they first optimize? CART and Random Forest illustrated that interest rate, number of credit cards, payment delays from due date, and outstanding debt are important in predicting whether a borrower will repay their debts on time. Thus, those looking to receive loans, mortgages, or credit cards should check their interest, credit mix, debt, and whether they have significant delays in payments prior to meeting with lenders. These analytically driven insights are valuable in helping lenders to reduce their risk, as well as in helping borrowers to increase their likelihood of receiving funds.

VI. References

The Investopedia Team. “Credit Score: Definition, Factors, and Improving It.” *Investopedia*, Investopedia, 18 Sept. 2022, https://www.investopedia.com/terms/c/credit_score.asp.

Paris, Rohan. “Credit Score Classification.” *Kaggle*, https://www.kaggle.com/datasets/parisrohan/credit-score-classification?select=train.csv.

VII. Appendix

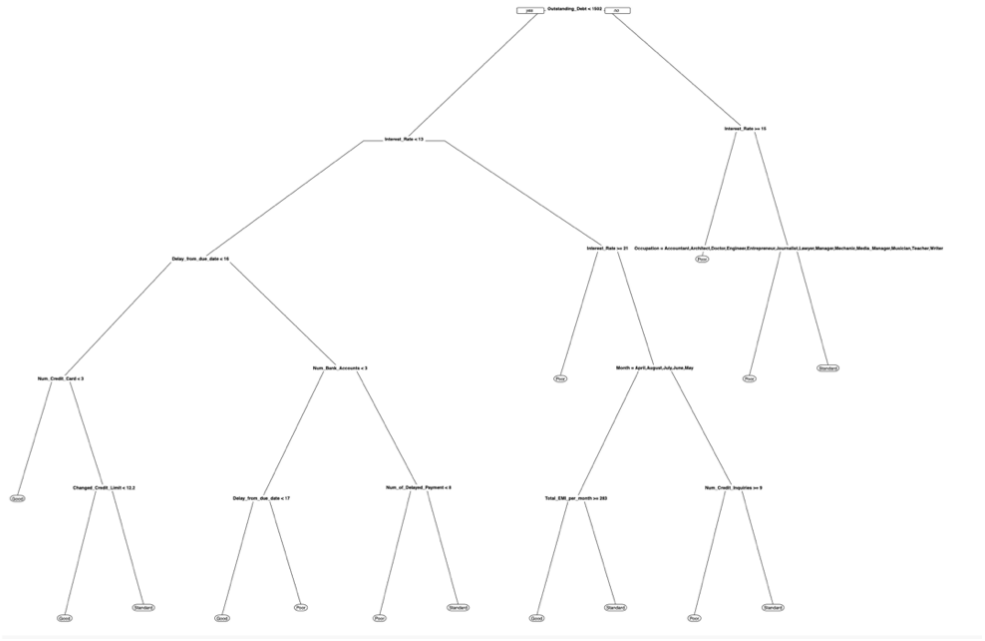

Figure 1: Baseline CART Decision Tree

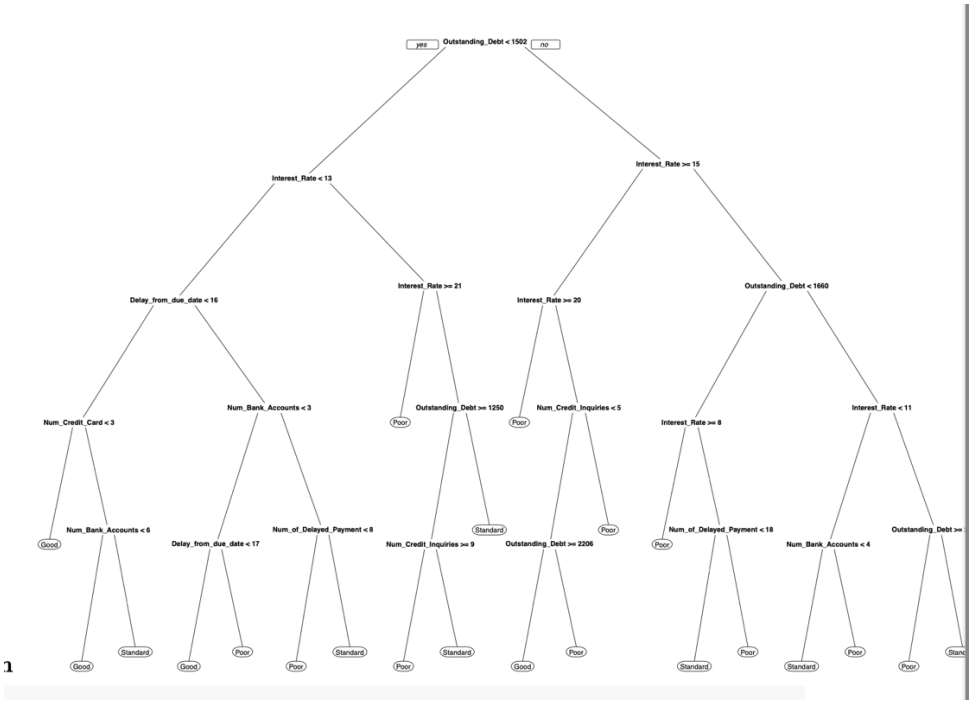

Figure 2: Improved CART Decision Tree

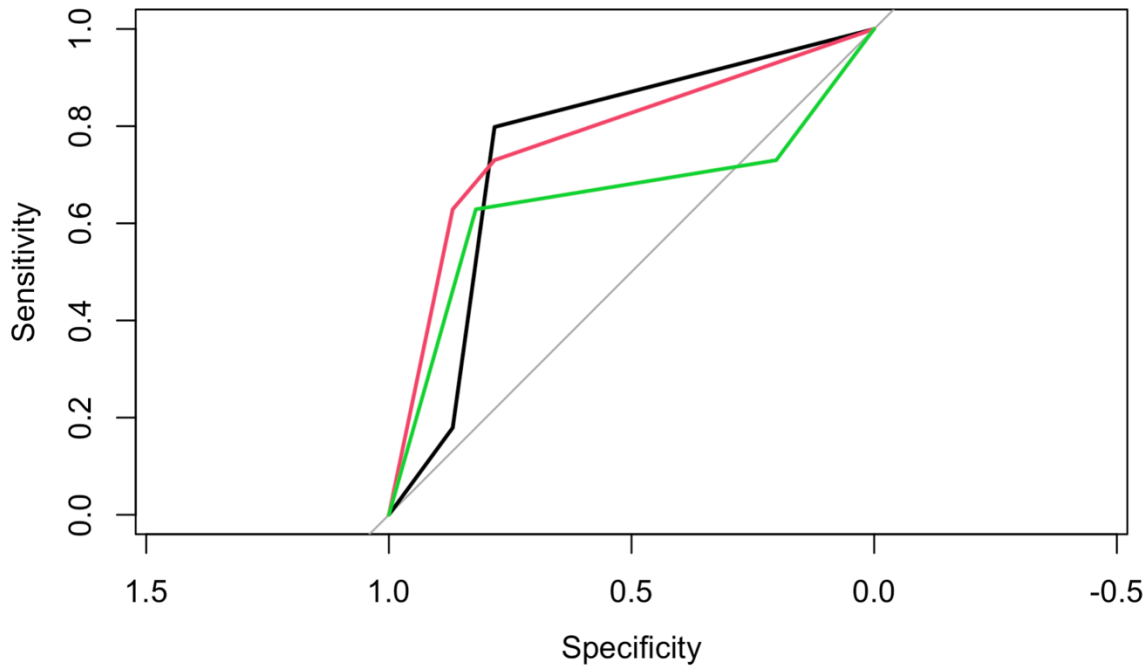

Figure 3: Baseline CART AUC Curve

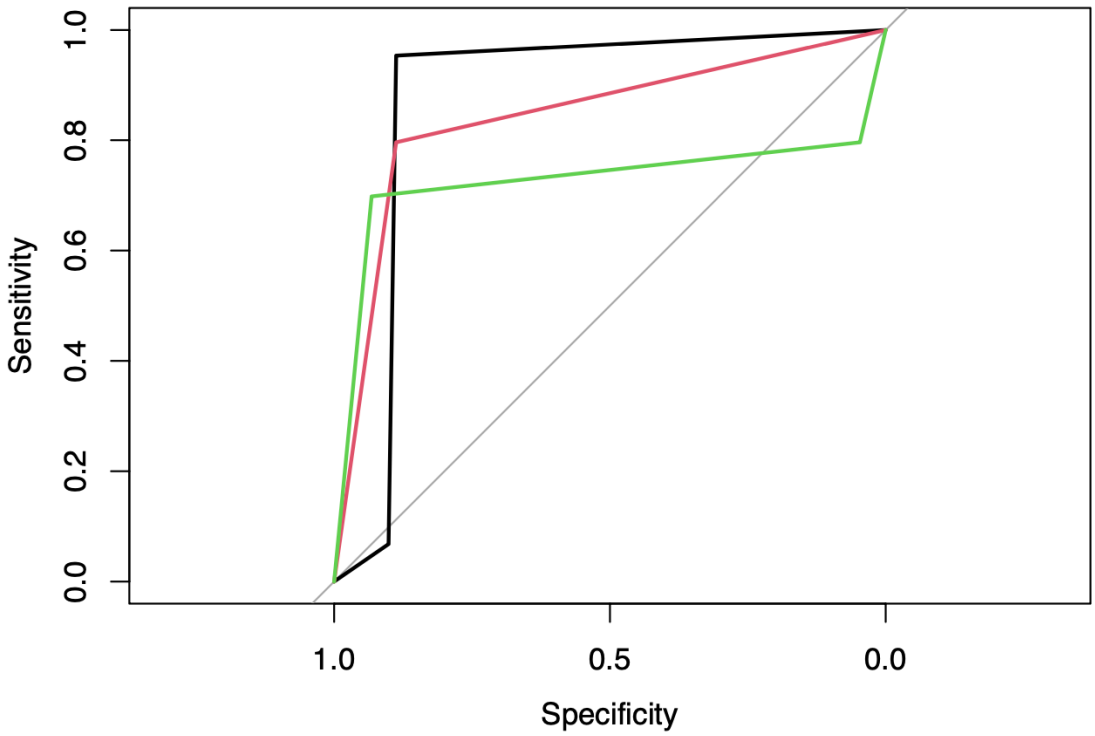

Figure 4: Baseline Random Forest AUC Curve

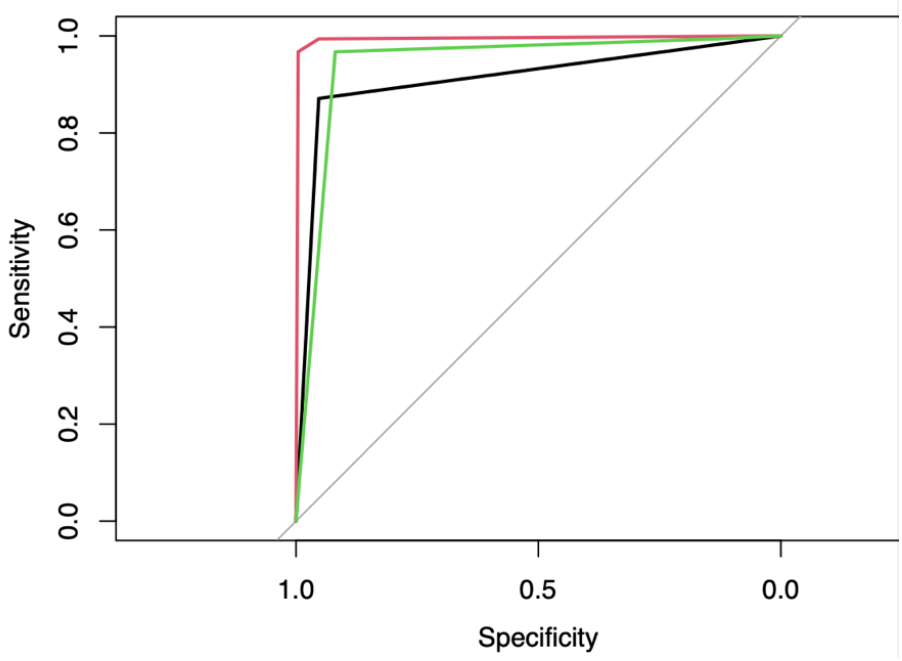

Figure 5: Baseline XGBoost AUC Curve