Presented by: Eshaan Vora

05/13/22

Executive Summary of Case Study:

Case Study 1:

This dataset includes various information about loans that have already been made including the characteristics of the loan receiver. Specifically, there is significant information about individual's delinquencies, debt profile, and creditworthiness. A Random Forest model and Lasso Regression model were implemented to predict the interest rate of the loan and to understand what variables are most important in determining a loan's interest rate.

The 5 most important variables were the sub grading of the loan, disbursement method, debt-to-income, total credit limit, and the loan amount.

Data Cleaning:

My first step in cleaning the data was to understand the scope of the missing data and which features were most affected and so I included a function to print number of "N/A" values per feature. I found that a majority of missing values were due to the fact that most individuals were individual filers and so I separated the dataset into single filers and joint filers. This allowed me to address the bulk of the missing values without compromising predictive quality. I then checked each string variable's factor level and dropped variables that only had 1 unique value as that would have no effect on model performance. Based on each model's variable importance ranking, I dropped the least important variables from the corresponding model.

Predictive Analysis:

The most predictive variable in modeling the interest_rate was the "sub-grade" feature (and this makes sense because the "sub-grade" is a measure of investment quality and therefore, the lower the investment grade, the more risk must be compensated for by a higher interest rate).

However, because loans are graded based on the same borrower characteristics (such as credit risks) as how interest rates are calculated and because loans are graded after they are already issued, the interest rate that is issued is not derived by the sub-grading of the loan but rather the underlying common loan characteristics which correlate the two variables so heavily. For this reason, we will drop our most predictive variable, "sub-grade", to explore the effects of other predicting variables. (Removing this single variable reduces our R-Squared value from 88% to 67%!)

I also dropped the "paid_interest", "paid_principal", and "balance" variables as these variables represent the size of the overall loan which follows the well-known principle that borrowing more money reduces creditworthiness and therefore increases the interest rate offered. (Removing these variables reduces our R-Squared value from 67% to 26%!)

Aside from the variables mentioned above: From the results of my Lasso and Random Forest regressors, the most significant predictors for loan interest rates were: "debt_to_income", "installment", "verified_income", and "disbursement_method."

Modeling Reasoning:

I built a Lasso Regression models to further understand which variables were most important in model performance and whether the interest rate could be predicted accurately using a linear regressor. If the relationship between interest rate and other variables is linear and because we have removed most highly collinear variables, a Lasso Regression model could be an effective and simple estimator of interest rates.

I found out that the Lasso model's error started plateauing at 36 variables, meaning there were some variables, out of the 43 variables in the model, which had little to no effect on model performance. I isolated and removed these variables including "loan_status", "term", or "total_debt_limit"

On another note, I also built a Random Forest Model to provide an ensemble algorithm to help prevent overfitting (as opposed to if I had pursued a Gradient Boosting algorithm) The model's hyperparameter "ntree", or the number of individual trees in the model, was tuned based on how many trees it takes before the error begins to plateau whereas the "mtry" hyperparameter was tuned based on the rule-of-thumb guideline of using the square-root of the number of explanatory variables and then determining which "mtry" value has produced the maximum area under curve close to that rule-of-thumb.

If I had more time, I would focus on validating and further tuning the model, as there are useful scores (such as RMSE) and parameters (such as "max_depth" or "bootstrap") that could be applied to improve model performance.