Data Analysis 3: Assignment 3

Nicolas Fernandez

Arbash Malik

GitHub (Code)

1. Introduction:

The purpose of this project was to predict the number of defaulted firms for the 'Manufacture of computer, electronic and optical products' industry for 2015. The goal was achieved through building multiple predictive classification models based on the data provided to us here. There were 5 models that were created, namely OLS Logistic, Logistic Lasso, Random Forest, and GBM, and their performance was evaluated based on their expected loss value, and the final prediction for default made was based on the model with the lowest loss value. These models consider variables like sales, accounting & financial variables, and human capital variables. Some variables were also engineered on the existing variables to help make a better model. The variables that were fed into the machine learning algorithms (Random Forest and GBM) were shortlisted from the model through LASSO.

2. Data Cleaning and Data Preparation:

The dataset that we downloaded initially had 287,829 observations, with 48 columns. The dataset contains data from 2005 to 2016. However, it had to be cleaned and filtered down to only the observations that were relevant and like the industry we had to predict for, making sure that the data we used in our models was a result of conscious decisions made, ensuring that we were not feeding any irrelevant data into the models that made us compromise on the quality and accuracy of the models. Hence, a lot of time and detailed thought process was spent on the data cleaning and data preparation.

The first decision we made was to unstack the dataset and stack the dataset again while creating for indexing for each year and company to make observations for each company in every year. We train our models exclusively on our preferred industry data. The reason for this is that it leads models that performs well specifically for that industry but may not generalize well to other industries, but since we only had to predict for 'Manufacture of computer, electronic and optical products' industry (*ind2 = 26*). This was also done in consideration of our computing power to make sure the models run fast, since the dataset was large. Next, we defined two very important variables, namely 'default' and 'status_alive'. The 'default' is a binary variable and indicates if the company has defaulted. The 'status_alive' is also a binary variable and indicates if the firm is still operating.

Some feature engineering was done to create variables for our models.

- **sales_vars** - This includes the sales and transformed sales columns as it is a great measure of a company's performance.
- **quality_vars** - Here we include the balance sheet columns for accounting for transparency in their financial reporting.
- **profit_loss_vars** - We account for the profit/loss of a company since this is also a measure of a company's performance.
- **balance_sheet_vars** - Including the balance sheet variables since it can indicate whether a company is likley to default.
- **human_capital_vars** - This represents characteristics of senior management for ecah company, for instance foreign majority CEOs or if the share of female CEOs is the majority.
- **firm_charac_vars** - These represent characteristics of each company, for example if majority gender of personnel is female, or if the company is in the central region of Europe.
- **d1_vars** - These are the first difference variables and account for changes in year-to-year across companies to check for trends, either positive or negative
- **engineered_vars** - These are to account for any trend in the data when examining the ratios that have been engineered from the data itself.
- **flag_vars** - These are the accumulate all of the flag variables that were created and assign them to a variable.
- **Interactions** are also made between sales and certain categorical variables that may affect the sales column data, for example the majority gender of personnel or the region location of the company. These have been broken up into two different variables, one to include in an OLS model and then the other to include in the LASSO.

The next decision we took was to winsorize the data in columns of our choosing. The penalties chosen given our data were the 1st percentile and the 90th percentile. The data has a long right tail due to some extreme values in the data and therefore a stricter 90 percent was used instead of the more classical 95 percent. We choose just the 1st percent because we want to account for the zombie firms that may exist that don't have any sales values or profit but also not discount the really small and/or struggling firms.

3. Sample Designing:

Next, we decided to split the training and holdout sets according to the definition of as given (see in code). Our training dataset had 9,689 observations while the holdout set had 1,037 observations. For the null values in both the training and holdout set, we chose to impute the values of numerical variables with the mean of their respective columns. We decided to use the mean values because given the low amount of nulls overall in each column, the mean and mode values can be considered to preserve their respective values and minimize bias in the estimation of other variables. While for the binary variable, we imputed with the mode. The rationale behind this approach is that it preserves the distribution of the binary feature and ensures that the imputed values are representative of the existing data.

4. Modelling:

There were 5 models that were created: 2 OLS Logistic, Logistic Lasso, Random Forest, and GBM, and their performance was evaluated based on their expected loss value, and the final prediction for default made was based on the model with the lowest loss value:

4.1 OLS logistic models:

We created two theoretically profound OLS models:

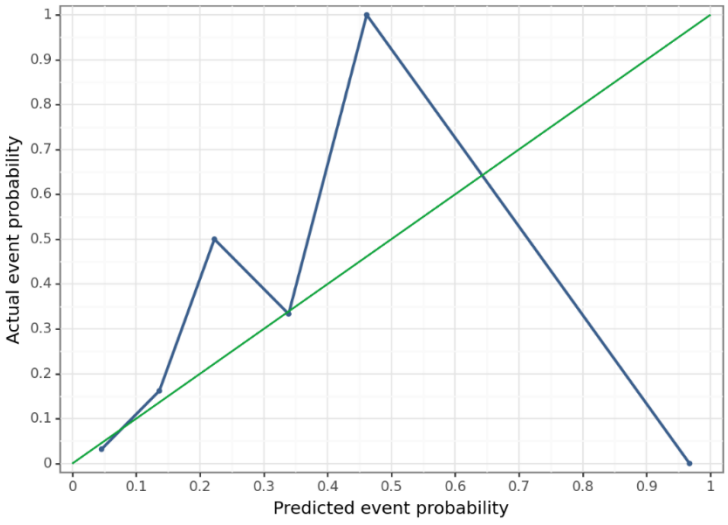
- **M1:** this model contains all the sales variables, firm quality variables, and profit/loss variables to account for basic predictors of whether it survived.
- **M2:** this model contains everything from M1 but also uses balance sheet variables, and basic interaction terms with sales.

Cross validation was done and RMSE & ROC_AUC scoring was also calculated for each model. Find the scores below:

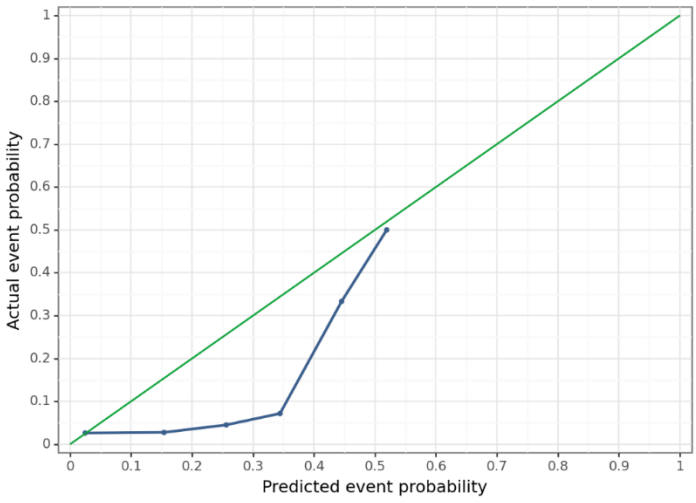
	Variables	CV RMSE	CV AUC
M1	16	0.237	0.704
M2	37	0.343	0.663

Calibration plots were also done which made it visible to us that the models were not performing well.

M1 Calibration Plot



M2 Calibration Plot



4.1 Lasso Logistic model:

We created a Lasso logistic model as well, where we provided all the variables for the lasso to shortlist. We also multiplied the lambda by 0.5 to make the penalty stricter to shortlist the best features in the model. The following values were obtained:

Variables	Lambda	CValue	CVRMSE
38	0.0005	0.258	0.232

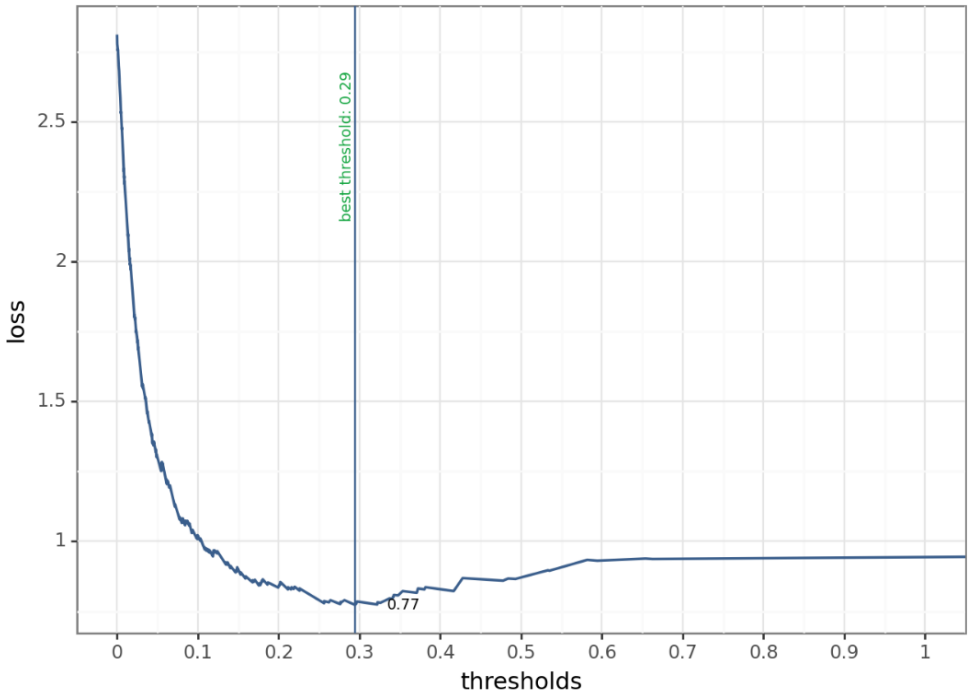
4.2 Random Forest Model:

We created a random forest model with the variables that lasso model shortlisted. This decision was done to ensure that the model performance is high, since using the same variables, a machine learning model will perform better than logistic model. The following results were obtained:

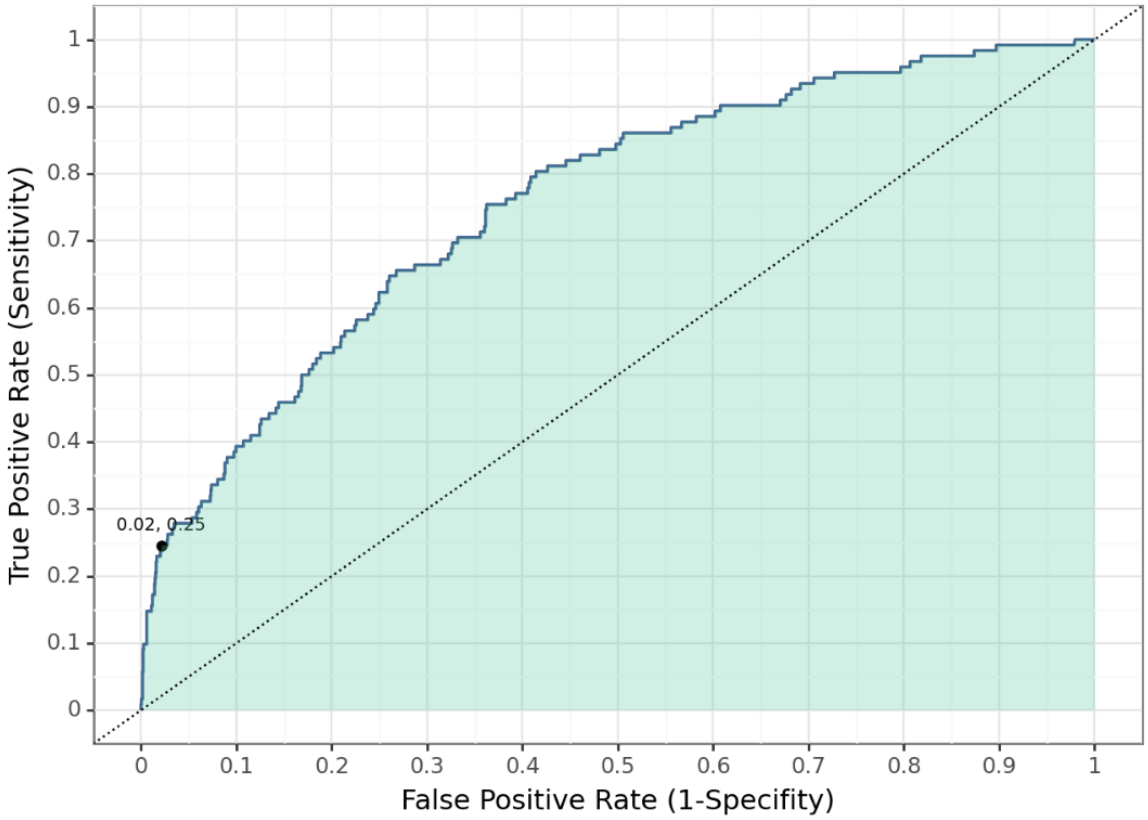
Max Features	Min Sample Split	CV Accuracy	CV AUC	CV RMSE
7	9	0.938	0.777	0.230

Some graphs were also created for the random forest model to visualize its performance:

Loss Plot



ROC Plot



4.4 GBM Model:

A Gradient Boosting Machine (GBM) model was constructed that also used the variables shortlisted from the LASSO model for the same reason as before.

Max Depth	Max Features	Min Sample Split	CV Accuracy	CV AUC	CV RMSE
6	40	10	0.937	0.744	0.239

5. Model Comparisons:

The models were run using Classification regressions given that the predicted variable was not continuous but binary with the target goal of the model to be the model with the lowest expected loss values. Amongst the logit models (M1, M2, and LASSO) the LASSO model performed the best regarding the training set when evaluating their average expected loss. The Random Forest model, however, performed even better than the logit models in this regard, as shown in the table below:

	Number of Coefficients	CV RMSE	CV AUC	CV Threshold	CV Expected Loss
M1	16	0.237	0.704	0.112	0.853
M2	37	0.343	0.663	0.413	0.884
LASSO	38	0.232	0.758	0.165	0.794
RF	38	0.230	0.777	0.216	0.761
GBM	38	0.238	0.744	0.157	0.796

The best three models were then run with the holdout set to compare their expected losses on the holdout set in order to test whether or not they are good predictive models. The results can be seen below:

	Expected Loss on Holdout
LASSO	2.008
RF	0.564
GBM	0.677

The lower the expected loss, the better the predictive value. In this regard we can see that the Random Forest model vastly outperforms the LASSO model and also does a much better job of predicting defaults correctly compared to the GBM model given our loss function parameters of penalizing a false negative by 15 and a false positive by 3. Below is a table showing the confusion matrix of the Random Forest model on the holdout set:

	Predicted (Not Default)	Predicted (Default)
Actual (Not Default)	946	35
Actual (Default)	32	24

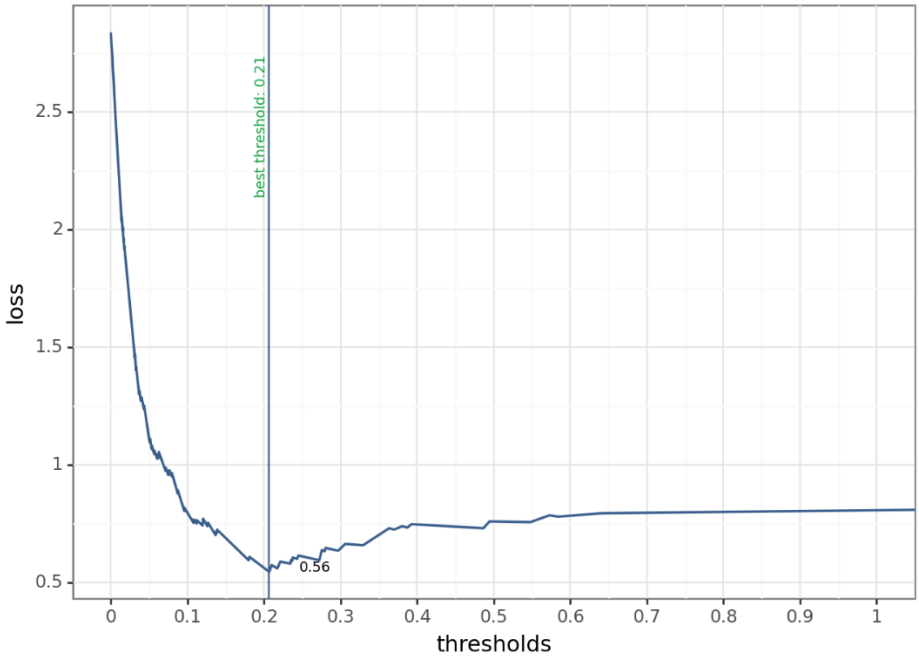
As we can see, the Random Forest model does a very good job of correctly predicting that a firm will not default while minimizing the most out of any model the number of firms that are predicted to not default that actually do default, which is what we are penalizing the most with our loss function. In conclusion, the Random Forest model does the best job of predicting firms to default based on past data out of the models that were created.

For random forest we created a calculated the required scores for the holdout set:

Holdout Brier	Holdout AUC	Holdout Accuracy	Optimal Threshold Sensitivity	Optimal Threshold Specificity
0.044	0.823	0.948	0.429	0.964

We also made the Loss plot and the ROC plot for the holdout set as well:

LOSS Plot for Holdout



ROC Plot for Holdout