DA3 Assignment 3 Hugo TING, Chun Yin

Introduction

Data Used

The focus of this study is on firms that experienced high growth rates between 2012 and 2014, within the period of 2010 to 2015. To distinguish between fast growing and non-fast-growing firms, we utilized a loss function that minimizes the average expected loss and applies a suitable threshold to aid investment decisions. The primary objective of this case study is to construct a predictive model that can aid individuals in their investment decisions by distinguishing between fast and non-fast-growing firms. The loss function used in this study assesses the impact of decisions driven by the prediction, with regards to false negatives and false positives. To classify firms, seven different models were developed, including OLS, LASSO, Random Forest, and OLS Logit, based on selected features of the companies. The data used for this study was obtained from Bisnode, a company that provides digital business, marketing, and credit information. The dataset contains detailed information about companies in the manufacturing and services industries from 2005 to 2016. The study focuses on the cross section of companies in 2012 and aims to determine whether they were fast growing or not in subsequent years.

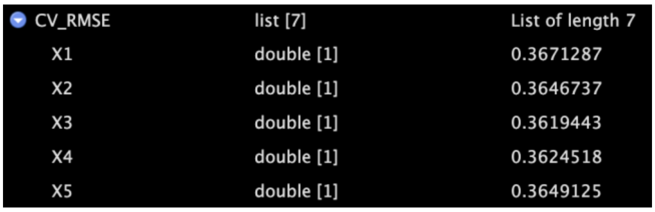
Data Modelling

We have set a threshold of 28% or higher for CAGR to determine whether a company is fast growing or non-fast growing. This threshold was selected because small or mid-sized firms typically experience higher annual growth rates than larger companies in their early years. Therefore, to classify a small or mid-sized firm as fast growing, we require it to have a CAGR of 28% or higher over a two-year period. Accordingly, we define fast growing firms as those whose CAGR sales value is 28% or greater. We are specifically focusing on mid and small size firms with sales between 10 million and 1000 euros. To begin label engineering, we have defined our y variable as whether a company is fast growing or non-fast growing.

Models Comparison

Logit Simple

A simple logit model prediction is performed, and the best logit model is selected after cross-validating and evaluating the Models by the holdout dataset. 5 different logit models are proposed and tested from simple to the most complex one which has the most variables. The simplest logit M1 model is built based on basic domain knowledge and more variables are added into the models 2-5. We are using the RMSE and the AUC score to choose which is the best model for prediction.

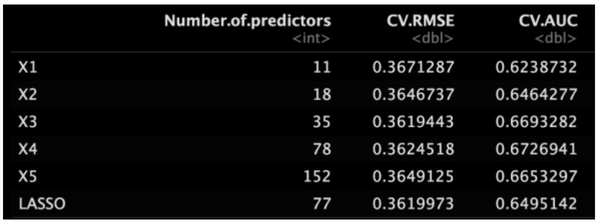


CV_RMSE	list [7]	List of length 7
X1	double [1]	0.3671287
X2	double [1]	0.3646737
X3	double [1]	0.3619443
X4	double [1]	0.3624518
X5	double [1]	0.3649125

The table above shows the RMSE of the 5 logit models. We can see that after doing a 5-fold cross-validation, logit model 3 has the lowest average RMSE out of the 5-folds. All of them perform at around 0.36-0.37 RMSE, so model 3 is prevailing by a small margin in terms of RMSE performance after cross-validation, so we will be using model 3 logit as the base model to compare with LASSO and random forest RMSE and AUC.

Logit LASSO

A LASSO model is also built based on the logit model 5 as logit model 5 has all the variables including the insignificant ones, and LASSO's penalty points should get rid of the useless variables.

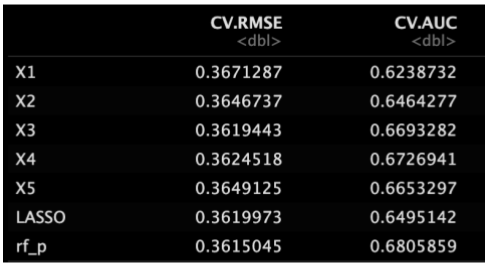


	Number.of.predictors <int>	CV.RMSE <dbl>	CV.AUC <dbl>
X1	11	0.3671287	0.6238732
X2	18	0.3646737	0.6464277
X3	35	0.3619443	0.6693282
X4	78	0.3624518	0.6726941
X5	152	0.3649125	0.6653297
LASSO	77	0.3619973	0.6495142

From the above table, we can see that by using logit model 5 with 152 variables, LASSO drops 75 of the variables so the LASSO logit model only has 77 variables. We can see the RMSE is improved from logit model 5 after shrinking the coefficients using LASSO, but the AUC decreases. In addition, the RMSE of LASSO is not performing as well as simple logit model 3 in both RMSE and AUC, so we can say that the LASSO logit is not the model we would like to have for prediction. Logit model 3 is still the prevailing prediction model before having random forest.

Random Forest

Random forest is the last prediction model in this exercise, and the functional forms and variables interactions are defined by the random forest itself. Tuning parameters of 5, 6 and 7 is set in each split.



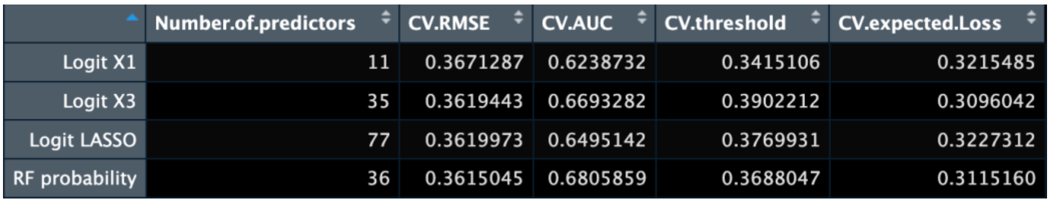
	CV.RMSE <dbl>	CV.AUC <dbl>
X1	0.3671287	0.6238732
X2	0.3646737	0.6464277
X3	0.3619443	0.6693282
X4	0.3624518	0.6726941
X5	0.3649125	0.6653297
LASSO	0.3619973	0.6495142
rf_p	0.3615045	0.6805859

From the table above, we can see that the in terms of the RMSE, random forest has the lowest among all the logit and LASSO models. It performs even better than the RMSE of model 3. At the same time, the random forest prediction model performs the best in terms of AUC after cross validation. So, the random forest will be the preferred model for prediction.

Diagnosis

Best Model for prediction

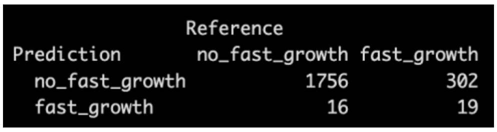
Apart from minimizing the RMSE and maximizing the AUC, we would also like to look at the expected loss and choose the lowest one with the loss function. Before looking into the expected loss numbers, random forest was our best model.



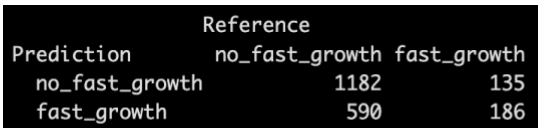
	Number.of.predictors	CV.RMSE	CV.AUC	CV.threshold	CV.expected.Loss
Logit X1	11	0.3671287	0.6238732	0.3415106	0.3215485
Logit X3	35	0.3619443	0.6693282	0.3902212	0.3096042
Logit LASSO	77	0.3619973	0.6495142	0.3769931	0.3227312
RF probability	36	0.3615045	0.6805859	0.3688047	0.3115160

From the table above, we can see that the expected loss of the random forest model is only the second lowest. In this case the logit model 3 performs better than the random forest model. Although the RMSE of simple logit model 3 is slightly higher than the random forest, this minor difference should not have a great impact on the prediction results from the logit model 3.

Confusion Matrix



```
                  Reference
Prediction        no_fast_growth fast_growth
  no_fast_growth            1756         302
  fast_growth                 16          19
```



```
                  Reference
Prediction        no_fast_growth fast_growth
  no_fast_growth            1182         135
  fast_growth                590         186
```

First, we used the 0.5 as a threshold for the loss function although we know it is not ideal, we would still like to see what the results is. The table on the left show the confusion table of the threshold = 0.5. From that, we can see that the accuracy is (1756+19)/2093 = 85%, sensitivity is 19/(19+302) = 6% and specificity is 1756/(1756+16) = 99%.

While the confusion matrix on the right used the mean of predicted probabilities to be the threshold, which is a more sensible way of doing so and the loss is minimized. From that, we can see that the accuracy is (1182+186)/2093 = 65%, sensitivity is 186/(186+135) = 58% and specificity is 1182/(1182+590) = 67%. We can see that after changing the threshold, the sensitivity increased significantly as it took the mean of the predicted probabilities into consideration. Although the accuracy and specificity decreased significantly, according to the loss function, using the mean of the predicted probabilities to be the threshold can minimize the loss in a confidence of 65% accuracy.