DA3 - Assignment 3

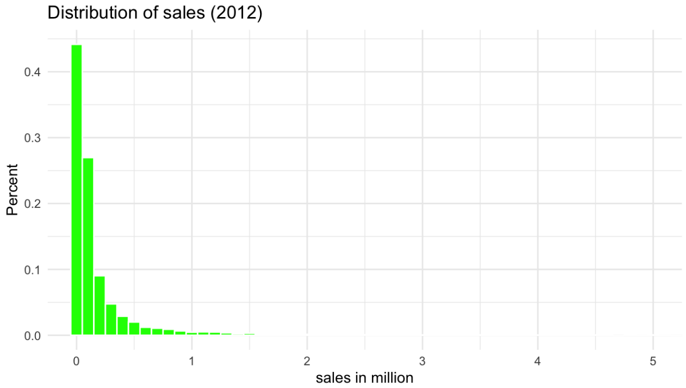
Summary Report

Oszkar Egervari

2022-02-18

Introduction

The goal of this project is to help investors finding companies with a potential for fast growth. To achieve this I built various prediction models on a dataset consisting of companies in the European Union. The data includes all registered companies from 2005-2016 in three selected industries (auto manufacturing, equipment manufacturing, hotels and restaurants). Several features were used during the prediction, such as P/L statements, balance sheets, management information etc. I considered a company fast growing when its compound annual growth rate (CAGR) exceeded 35%. This is an arbitrary number, however after some research I found out that between 15% and 25% CAGR, investors consider a company to be a ‘good investment’ in general, so I wanted to overshoot that value by a bit.



To create a proper analysis we filtered the data for companies with full balance sheet for these 3 years and with a sales number exceeding 0. On top of that, I excluded companies with a sales value (annual) below 1000 and above 10 million Euros. At this point we are left with 59 538 observations. As a last step, the year is set to 2012 and - after creating the cagr_sales variable - companies with cagr_sales values exceeding 5000 or NAs are filtered out, which leaves us with 14 628 observations. The distribution of CAGR of sales can be seen below.

Feature engineering

During this task I take a look at financial variables and their validity. Variables were flagged with mistakes in their balance sheet (e.g. negative values in assets), also category variables and factors were created. Finally I dropped observations with too many missing values in key variables and we finished with 116 variables and 13 099 observations.

Modelling

fast_growth_f	Number of companies	Percentage
no_fast_growth	10703	82%
fast_growth	2396	18%

I. Probability logit models

First I build 5 different logit models. Each model contains a different set of variables increasing in complexity, meaning that model 1 is the least complex, while model 5 is the most.

	Variables
X1 model	Log sales + Log sales^2 + Change in Sales + Profit and loss + Industry
X2 model	X1 + Fixed assets + Equity + Current liabilities (and flags) + Age + Foreign management
X3 model	Log sales + Log sales^2 + Firm + Engine variables 1 + D1
X4 model	X3 + Engine variables 2 + Engine variables 3 + HR
X5 model	X4 + Interactions 1 and 2

Comparing the models, the two most important measures were the Root Mean Squared Error (RMSE) and the Area Under Curve (AUC), both averaged on the 5 different folds used during cross validation. We can see from the table below that the RMSE and AUC values are really similar for all of the models. The best model that had the lowest RMSE was the third one (X3), but the best model that has the highest AUC was the fourth one (X4). These two models outperformed all the others based on these criterion. I will consider the X3 model as a benchmark as it is simpler with less than half of the predictors the 4th model possesses.

	Number.of.predictors	CV.RMSE	CV.AUC
X1	11	0.3788	0.6394
X2	18	0.3767	0.6574
X3	35	0.3747	0.6771
X4	75	0.3755	0.6781
X5	149	0.3770	0.6720

II. LASSO model

After the basic logit models, by applying LASSO the variables are selected based on calculations, rather than being handpicked. The LASSO model contains all the variables of the 5th logit model.

	Number.of.predictors	CV.RMSE	CV.AUC
X3	35	0.3747	0.6771
LASSO	24	0.3748	0.6586

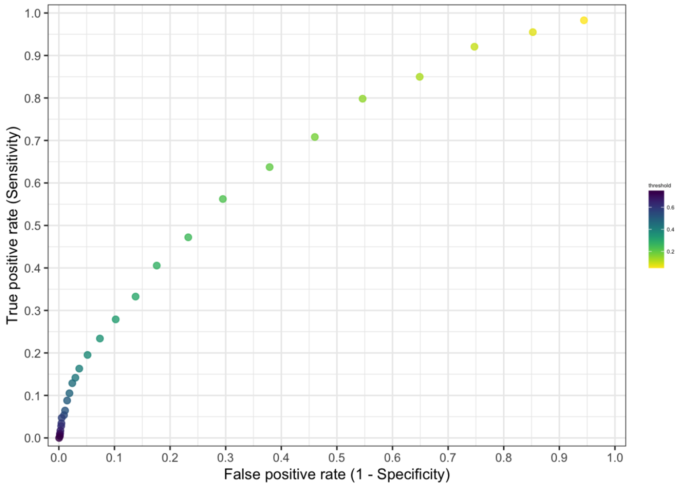
III. Random forest

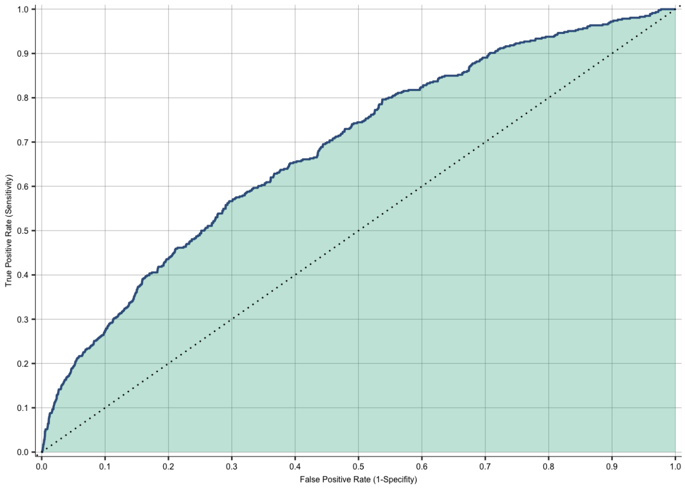
After the logit and LASSO models, I built a random forest model, since it’s good at finding non-linear patterns and interactions. The variables of the fourth model were used without adding polynomials, flag variables etc. The random forest outperformed our best model, the X3 logit model, both in terms of lower RMSE with 0.372 and higher AUC with 0.691 values.

	CV.RMSE	CV.AUC
X3	0.3747	0.6771
Random_forest	0.3722	0.6912

ROC curve

Before finding the best thresholds for our prediction, I drew up the ROC plot for the random forest model. The curve shows us how the increasing threshold leads to lower True positive and False positive rates. The second plot shows the Area Under the Curve. There is a trade off between accurate true positive and true negative predictions. Setting up the loss function helps us find the optimal threshold.





Finding the optimal threshold

To find the optimal threshold, we have to weigh the consequences of a false negative and a false positive prediction. In case of a false positive prediction, one would make a bad investment. It can mean acquiring less return compared to an alternative investment, but also potentially losing money. On the other hand, a false negative prediction would would result in loosing out on a good investment. Considering all this, my personal opinion is that a false positive prediction has less downside, than a false negative, so I set the loss function to a 1/2 FP/FN ratio, meaning that a false negative prediction costs twice as much as a false positive.

	Number.of.predictors	CV.RMSE	CV.AUC	CV.threshold	CV.expected.Loss
Logit X3	35	0.3747	0.6771	0.3507	0.3392
Logit LASSO	24	0.3748	0.6586	0.3413	0.3517
RF probability	33	0.3722	0.6912	0.3514	0.3347

Model choice

The choice for the final model was mainly based on the expected losses. Here the lowest expected loss is for the random forest, then the X3, finally the LASSO model. Ordering the models based on AUC results in the same list. RMSE results are the same with these mdoels.

Even though the random forest was the best for all the comparison measures, I ended up going with the less complex X3 logit model. The numbers are close, but X3 is a much simpler model, having less variables than the LASSO model and it is easily to interpret compared to the black box random forest model. On top of that, the random forest runs significantly longer than the other models.

Model evaluation

Now that we have the optimal threshold vales we can turn to the holdout set and evaluate our chosen best model the logit X3. The expected loss we get when calculating it on the holdout set is **0.325**. This is even smaller than what we got for the train data. Another way of checking the performance of our model is taking a look at the confusion matrix. Based on that, the model accuracy is 81.3% and its sensitivity is 93.7%.

	no_fast_growth	fast_growth
no_fast_growth	2047	373
fast_growth	106	93

model	Test.RMSE	Test.AUC	Avg..Exp..Loss
X3 Logit Model	0.3674	0.6834	0.3249