Finding fast growing firms

Maeva Braeckevelt and Brúnó Helmeczy

Executive Summary

This analysis aimed at predicting the fast growing of a firm.

Introduction

This analysis serves to predict fast growth of firms between 2012 and 2014. This analysis could help an investor to decide to invest in a certain company or not. The data was originally created by Bisnode, a major European business information company. The original dataset, bisnode-firms, considers companies between 2005 & 2016 in both Manufacturing (Electrical equipment, Motor vehicles, etc) & Services (Accommodation, and Food & Beverage services activities).

As business context, we consider our client to be a silent investor group as of 2012, who is looking for suitable companies to invest in, with expected returns on their investments in 2 years. As such, we considered companies at least 1 year old, since such companies have all accumulated financial data over multiple years, yet many are young, with higher chances of gaining further market share. We only considered businesses generating annual sales between 100 thousand, & 10 million Euros, boasting sufficiently large sales for the business venture to be considered 'serious', yet small enough for it's size not to be considered a disadvantage.

Data

Label engineering

The first step of this analysis was to define fast growth. To do so, we used the compound annual growth rate formula.

CAGR = (Vfinal/Vbegin)^1/t - 1

t being the time in years, we defined it as 2. It seems logic for us that if an investor took interest in a firm, he will be able only to invest the year later. Vbegin is the Sales (€) in the year ongoing Vfinal is the Sales (€) in the year +2

we converted in percentage. we defined any firm that have a % CAGR of bigger or equal at 30% in two years as a fast growing firm, it means that they growth for 70% in two years.

Sample design

Our sample is defined as - Firms still in business in 2012 - Firms that have sales between 10 000€ and 10 millions €

Our fast growing firms are compound annual growth percentage bigger or equal at 30% in two year

After having design our sample, we end up with 15692 companies and 2220 of them (14%) are fast growing.

Feature engineering

#MAEVa Now, We need to select our x variables and maybe transform them but before we need to clean them. - Some of the variable are financial accounts (like inventories, current liabilities, etc), so they can't be negative, thus we decided to replace the negative value to 0. - We created ratios , easier for interpretation and spotting extreme values - we used the a method called winsorization - We elevated some variable to a quadratic function - We took the log of sales.

We decided to classify the variable

- **Firm**, *5 variables* : Age of firm, squared age, a dummy if newly established, industry categories, location regions for its headquarters, and dummy if located in a big city
- **Financial 1**, *16 variables* : Winsorized financial variables : sales, fixed, liquid, current, intangible assets, curret liabilities, inventories, equity shares, subscribed capital, sales revenues, income before tax, extra income, material, personal and extra expenditure, extra profit.
- **Financial 2** : Flags (extrem, low, high, zero - when applicable) and polynomials : quadratic term are created for profit and loss, extra profit and loss, income before tax, and share equity.
- **Financial 3** : % blance sheet, % profite and loss
- **Growth**, *X variable* : Sales growth is captured by a winsorized growth variable, its quadratic term and flags for extreme low and high values.
- **HR**, *5 variable* : 5 variables : For the CEO: Female dummy, winsorized age and flags, flags for missing information; foreign management dummy; labor cost, and flag for missing labor cost information.
- **Data quality**, *3 variables*: Variables related to the data quality of the financial information, flag for a problem, and the length of the year that the balance sheet covers.

We chose to include some interactions as well, that we defined by common knwoledge.

- **Interaction** : Interactions with the sales growth, firm size, and industry

Probability prediction and model selection

We decided to use three different models for the prediction of fast growing firm : Logit, Lasso and Random Forest. Our 5 Logistic regression models gradually incorporate more & more of the variable groups above, where our final model, similarly to LASSO, incorporates all available variables. Finally, our Random Forest model considers all variable groups except interactions.

Logit

```
## + Fold1: mtry= 9, splitrule=gini, min.node.size=10
## - Fold1: mtry= 9, splitrule=gini, min.node.size=10
## + Fold1: mtry=10, splitrule=gini, min.node.size=10
## - Fold1: mtry=10, splitrule=gini, min.node.size=10
## + Fold1: mtry=11, splitrule=gini, min.node.size=10
## - Fold1: mtry=11, splitrule=gini, min.node.size=10
## + Fold1: mtry= 9, splitrule=gini, min.node.size=15
## - Fold1: mtry= 9, splitrule=gini, min.node.size=15
## + Fold1: mtry=10, splitrule=gini, min.node.size=15
```

```
## - Fold1: mtry=10, splitrule=gini, min.node.size=15
## + Fold1: mtry=11, splitrule=gini, min.node.size=15
## - Fold1: mtry=11, splitrule=gini, min.node.size=15
## + Fold2: mtry= 9, splitrule=gini, min.node.size=10
## - Fold2: mtry= 9, splitrule=gini, min.node.size=10
## + Fold2: mtry=10, splitrule=gini, min.node.size=10
## - Fold2: mtry=10, splitrule=gini, min.node.size=10
## + Fold2: mtry=11, splitrule=gini, min.node.size=10
## - Fold2: mtry=11, splitrule=gini, min.node.size=10
## + Fold2: mtry= 9, splitrule=gini, min.node.size=15
## - Fold2: mtry= 9, splitrule=gini, min.node.size=15
## + Fold2: mtry=10, splitrule=gini, min.node.size=15
## - Fold2: mtry=10, splitrule=gini, min.node.size=15
## + Fold2: mtry=11, splitrule=gini, min.node.size=15
## - Fold2: mtry=11, splitrule=gini, min.node.size=15
## + Fold3: mtry= 9, splitrule=gini, min.node.size=10
## - Fold3: mtry= 9, splitrule=gini, min.node.size=10
## + Fold3: mtry=10, splitrule=gini, min.node.size=10
## - Fold3: mtry=10, splitrule=gini, min.node.size=10
## + Fold3: mtry=11, splitrule=gini, min.node.size=10
## - Fold3: mtry=11, splitrule=gini, min.node.size=10
## + Fold3: mtry= 9, splitrule=gini, min.node.size=15
## - Fold3: mtry= 9, splitrule=gini, min.node.size=15
## + Fold3: mtry=10, splitrule=gini, min.node.size=15
## - Fold3: mtry=10, splitrule=gini, min.node.size=15
## + Fold3: mtry=11, splitrule=gini, min.node.size=15
## - Fold3: mtry=11, splitrule=gini, min.node.size=15
## + Fold4: mtry= 9, splitrule=gini, min.node.size=10
## - Fold4: mtry= 9, splitrule=gini, min.node.size=10
## + Fold4: mtry=10, splitrule=gini, min.node.size=10
## - Fold4: mtry=10, splitrule=gini, min.node.size=10
## + Fold4: mtry=11, splitrule=gini, min.node.size=10
## - Fold4: mtry=11, splitrule=gini, min.node.size=10
## + Fold4: mtry= 9, splitrule=gini, min.node.size=15
## - Fold4: mtry= 9, splitrule=gini, min.node.size=15
## + Fold4: mtry=10, splitrule=gini, min.node.size=15
## - Fold4: mtry=10, splitrule=gini, min.node.size=15
## + Fold4: mtry=11, splitrule=gini, min.node.size=15
## - Fold4: mtry=11, splitrule=gini, min.node.size=15
## + Fold5: mtry= 9, splitrule=gini, min.node.size=10
## - Fold5: mtry= 9, splitrule=gini, min.node.size=10
## + Fold5: mtry=10, splitrule=gini, min.node.size=10
## - Fold5: mtry=10, splitrule=gini, min.node.size=10
## + Fold5: mtry=11, splitrule=gini, min.node.size=10
## - Fold5: mtry=11, splitrule=gini, min.node.size=10
## + Fold5: mtry= 9, splitrule=gini, min.node.size=15
## - Fold5: mtry= 9, splitrule=gini, min.node.size=15
## + Fold5: mtry=10, splitrule=gini, min.node.size=15
## - Fold5: mtry=10, splitrule=gini, min.node.size=15
## + Fold5: mtry=11, splitrule=gini, min.node.size=15
## - Fold5: mtry=11, splitrule=gini, min.node.size=15
## Aggregating results
## Selecting tuning parameters
## Fitting mtry = 9, splitrule = gini, min.node.size = 10 on full training set
```

We started by carrying out a probability prediction by logit. Used 5-fold cross-validation to select the best model.

We created five different models. The predictors of the first two models were handpicked and then we gradually add more categories at each models.

- **X1** : Log of sales in Millions, the square of the Log of sales in Millions , Winsorized value of the change of the log of Sales versus last year, profit and loss by total sales, Industries categories
- **X2** : X1, fixed assets divided by total assets, share equity divided by total assets,current liability divided by total assets, hight flags for current liability divided by total assets, flag error for current liability divided by total assets,age, foreign_management
- **X3** : Log of sales in Millions, the square of the Log of sales in Millions, Firm, Financial 1, Growth
- **X4** : Log of sales in Millions, the square of the Log of sales in Millions, Firm, Financial 1, Growth, *Financial 2*, *HR*, *Data quality*
- **X5** : Log of sales in Millions, the square of the Log of sales in Millions, Firm, Financial 1, Growth , Financial 2, HR, Data quality, *Intercactions*

variables - tables

Ratio EBITDA

Show RMSE

Choose the model

AvgCAGR
64.1

AvgCAGR
0.45

	0	1
0	0.81	1
1	0.19	0

Model	Number.of.predictors	CV.RMSE	CV.AUC	CV.threshold	CV.expected.Loss
X1	11	0.366	0.58	0.306	0.805
X2	18	0.364	0.608	0.354	0.802
X3	61	0.361	0.646	0.469	0.795
X4	114	0.362	0.648	0.47	0.793
X5	245	0.366	0.644	0.549	0.798
LASSO	26	0.361	0.625	0.488	0.8
RandForest	114	0.361	0.651	0.45	0.795

all variables

comparing lasso to the best logit

Random forest

variable from our best logit model (take off the interaction)

models

Classification

Ex-Sample Testing & Model Diagnostics

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

	no_Hyp.Growth	Hyp.Growth
no_Hyp.Growth	2471	466
Hyp.Growth	21	21

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

HyperGrowth_f	AvgAnnGrowth
no_Hyp.Growth	-29.3
Hyp.Growth	85.9

HyperGrowth_f	AvgAnnGrowth
no_Hyp.Growth	-29.3
Hyp.Growth	85.9

[1] -104139 [1] 508725

```
## `summarise()` regrouping output by 'HyperGrowth_f' (override with `.groups` argument)
```

HyperGrowth_f	ind2	AvgAnnGrowth	Nr_Investments
no_Hyp.Growth	26	-26.1	3
no_Hyp.Growth	27	-46.3	3
no_Hyp.Growth	28	-56.1	2
no_Hyp.Growth	29	-25.1	3
no_Hyp.Growth	33	-100	1
no_Hyp.Growth	55	-21.6	1
no_Hyp.Growth	56	-11.2	8
Hyp.Growth	26	62.1	1
Hyp.Growth	27	31.5	1
Hyp.Growth	28	34.3	2
Hyp.Growth	29	155	4
Hyp.Growth	33	119	1
Hyp.Growth	55	100	2
Hyp.Growth	56	70.1	10

Conclusion / Summary