SEC Data as Predictor of Corporate Success

Francesco De Faveri

August 2020

Table of Contents

1. Introduction

Business success is an essential factor for the well-being of national economies in the world, it follows that it is essential to be able to recognize the determinants responsible in view of changing conditions in the economic system.

This project wants to demonstrate how it is possible to predict the success of a company using the data available on the website of the Securities and Exchange Commission.

In this regard, we define a successful company as one that is growing from an income point of view (Penrose 1959) and that distinguishes itself by a positive differential between revenues and costs.

2. Methodology

The data used refer to the balance sheets of all companies in the SEC database from 2009 to 2019 and can be found via the following hyperlink: https://www.sec.gov/dera/data/financial-statement-data-sets.html

The data are broken down by year and include the information extracted from the EX-101 annexes submitted to the Commission.
They are divided into four data sets containing information about submissions, numbers, taxonomy tags, and presentation.

Before I could proceed with the data cleaning I had to re-group the information in a single database, selecting those relevant to the project.
I filtered the data by retaining only those that fell under the generally accepted accounting principles (GAAP), annual reports (Form 10-K), income statement, balance sheet and cash flow statement.

Once I re-aggregated the data I was able to proceed with data cleaning. I removed duplicates, handled categorical data and converted strings to datetime.

I selected the features with the lowest number of missing values and replaced the missing values with a constant value.

Finally I defined the dependent variable through a Boolean logic by putting equal to 1 all the companies that have seen their net income grow in the years 2016 2017 and 2018.

In this way I obtained a dataframe of 53357 rows for 98 columns with a different balance sheet item for each column containing the financials from 2009 to 2015.
Descriptive statistics have been applied on the dataset to analyze the degrees of correlation between the variables.

Subsequently eight different predictive models were tested with the aim of obtaining the best possible result

The models were:

1. k-nearest neighbors
2. Linear logistic regression
3. Decision tree
4. Random forest
5. Support vector machine
6. Multi-layer perceptron
7. XGBoost
8. Keras neural network on GPU

The dataset was divided into training set, validation set and test set and the models were evaluated using MSE, F1 Score, Precision and Recall.

3. Results

Due to the large amount of features it was preferred to analyze the dataframe through parametric and non-parametric tests.

Once created the heatmap using the Pearson correlation as a metric, we could see that there are several positive and negative linear correlations of moderate intensity between the independent variables.
This indicates the presence of multicollinearity.
On the other hand, there is no significant linear relationship between each of the independent variables and the dependent variable.
This suggests that the relationships between the independent variables and the dependent variable are non-linear.

Using Kendall's correlation, we notice that some of the independent variables have a slight positive or negative relationship with the dependent variable.
In this case, variables that were previously not linearly correlated now appear to be correlated through a monotonous relationship.
An additional clue that makes us think of a non-linear relationship between the independent variables and the dependent variable.

Once we finished the initial training of the models we could analyze the results, here is a table that summarizes the first iteration of the algorithms according to the F1 Score calculated on the validation set.

Model	F1 Score
k-nearest neighbors	0.19
Linear logistic regression	0.02
Decision tree	0.38
Random forest	0.30
Support vector machine	0.02
Multi-layer perceptron	0.37
XGBoost	0.01
Keras neural network on GPU	0.33

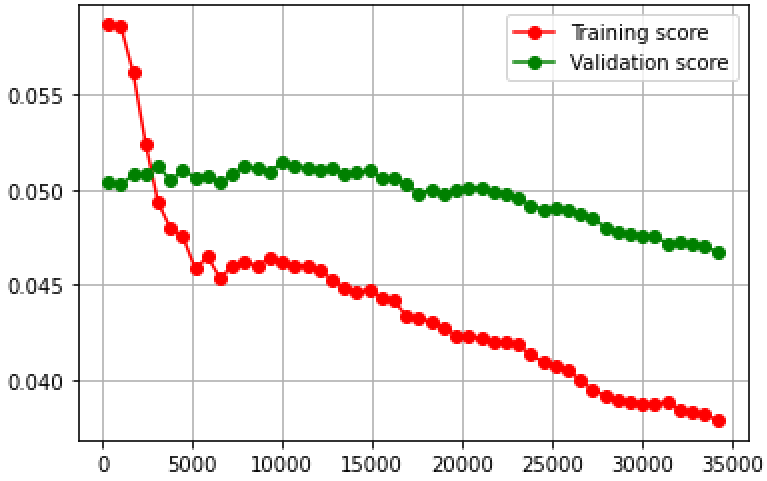
As you can see the models with the best initial performance are Decision tree, Random forest, Multi-layer perceptron and Keras neural network on GPU.

The fact that the Linear logistic regression has performed so badly brings us back to the hypothesis of non-linearity between independent and dependent variables.

The superior performance of algorithms that exploit decision trees and neural networks are probably attributable to their ability to deal with nonlinear relationships, variable interactions, and highly skewed data.

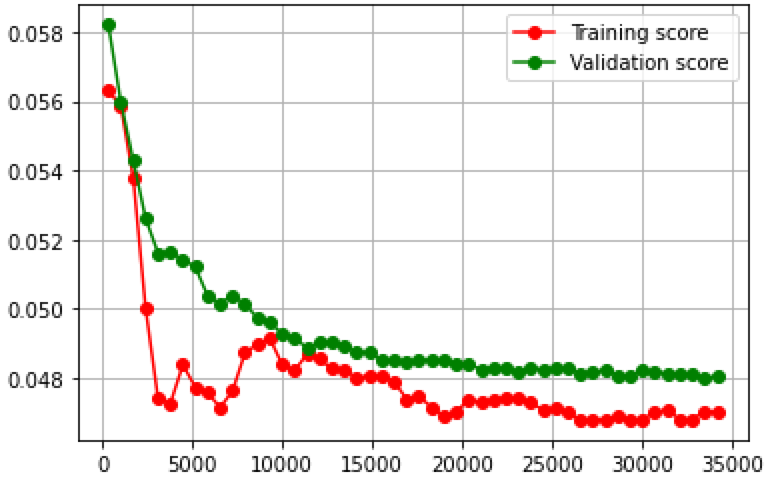
We have traced the learning curves of each model with MSE as error to analyze the learning process.

As far as the k-nearest neighbors model is concerned, we can see that as the training samples increase, the error in training and validation sets decreases; this indicates that the model is able to learn more if more data is collected.



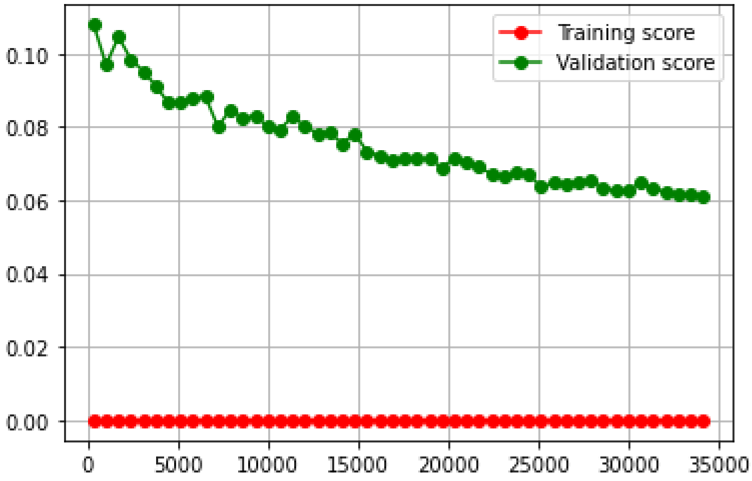
k-nearest neighbors

In the case of the model Linear logistic regression the error reaches the minimum with relatively few training samples and remains stationary indicating the inability of this last one to learn further given its simplicity.

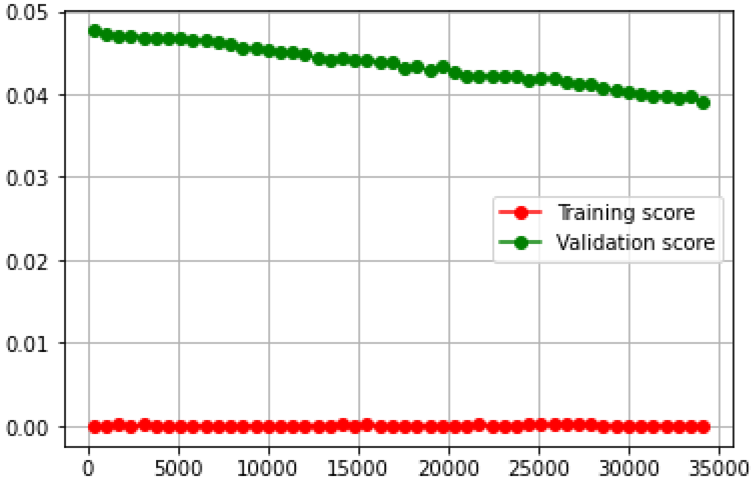


Linear logistic regression

The error for the Decision tree is clearly improving as the training samples increase and the same is true for Random forest.

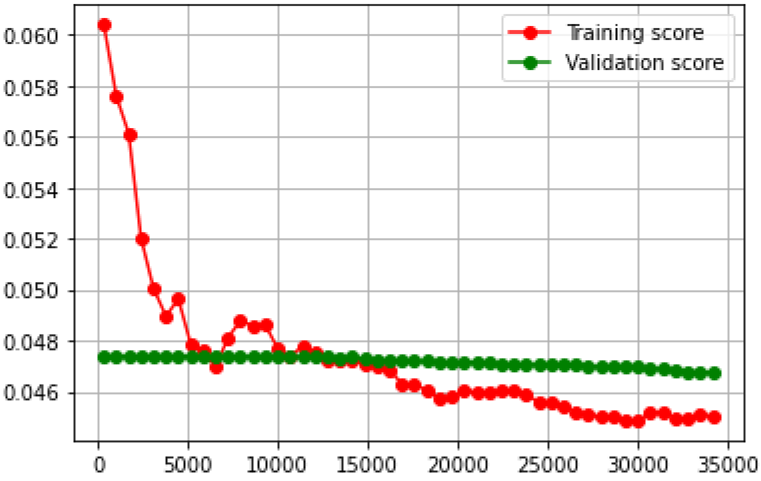


Decision tree



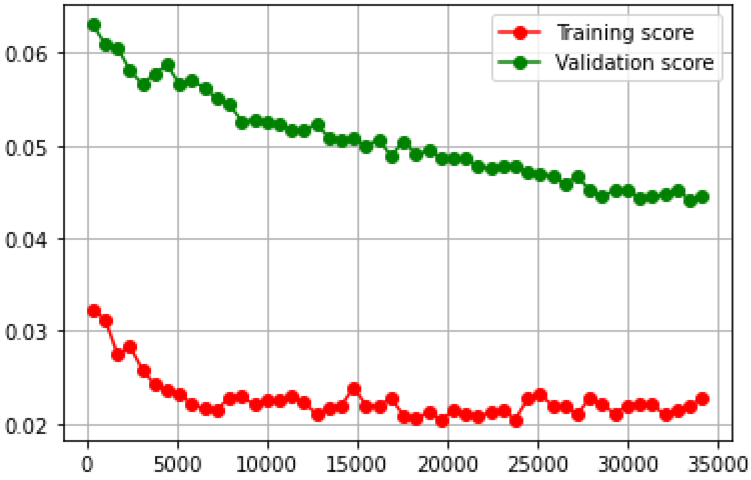
Random forest

The error is stationary also for Support vector machine.

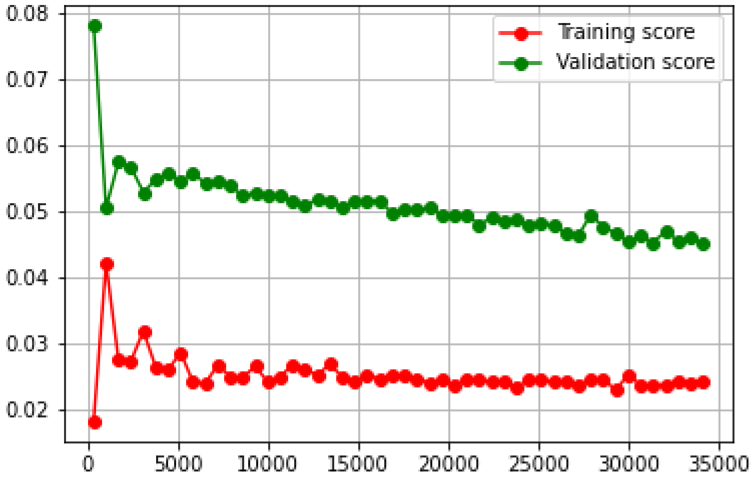


Support vector machine

For the Multi-layer perceptron the error decreases with increasing samples and the same is true for Keras neural network on GPU.

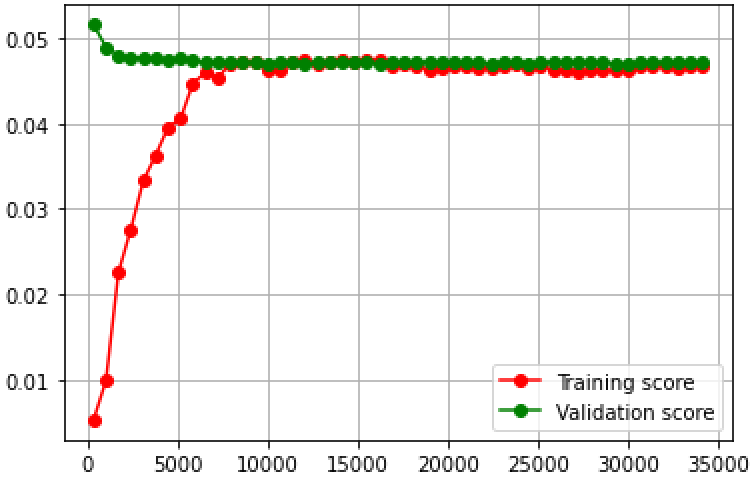


Multi-layer perceptron



Keras neural network on GPU

The error for XGBoost seems to suggest a model too simple to represent the complexity of the relationships between variables.



XGBoost

At this point we decided to use randomized search on hyper parameters of the models with the best performance to see if significant improvements in performance can be achieved.

Here are the results.

Model	F1 Score (before tuning)	F1 Score (after tuning)
Decision tree	0.38	0.38
Random forest	0.30	0.32
Multi-layer perceptron	0.37	0.46
XGBoost	0.01	0.62
Keras neural network on GPU	0.33	0.48

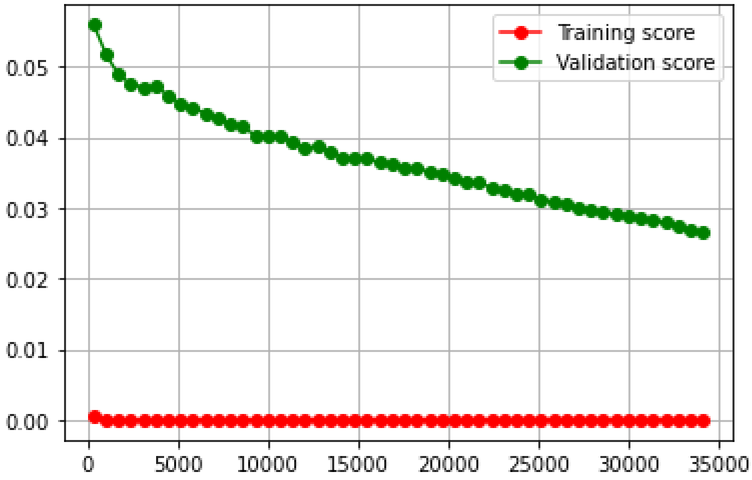
5. Conclusion

Of the eight models presented, five proved best able to capture the relationships between independent variables and independent variable.

Thanks to parameters tuning, three of them showed a clear improvement in performance.

Of all the predictive models used, XGBoost seems to be the most robust to predict a company's profitability success in the medium term.

Nevertheless, a significant variance could not be explained by the models and their performance, as far as the F1 Score is concerned, is still rather unsatisfactory for a field implementation.



XGBoost after parameters tuning

More data, especially data about successful companies, can help improve model performance.

This is clearly visible through the learning curves of models with better performance that indicate high variance despite the use of tuned regularization parameters.

One solution may be to use quarterly data instead of yearly data to increase the number of training samples and thus reduce overfitting.

6. References

https://www.coursera.org/professional-certificates/ibm-data-science

https://www.coursera.org/learn/machine-learning/home/welcome

Penrose E., (1959), The Theory of the Growth of the Firm

Porter M., (1991), Towards a Dynamic Theory of Strategy

Porter M., (1985), Competitive advantage: creating and sustaining superior performance

Grant R., (1999) L'analisi strategica per le decisioni aziendali