Group Project

Investment Recommendations based on Financial Indicators by the use of Machine Learning

Machine Learning in Finance

Authors

Hodel Marius

Ly Tony

Rauhut Christian

Lecturer

Zimmermann Benjamin

Department of Banking and Finance

University of Zurich

Date of Submission: 18.04.2021

Contents

List of Figures

List of Tables

1 Overview

The task consists of creating an investment recommendation model, which is able to predict an annual stock return based on financial indicators from the US Stock market for the period of 2014 to 2018. Our model should be able to indicate the return into a buy/hold/sell classification, which will be compared with the S&P 500, a stock market index which includes the largest companies listed in the United States.

Before delving into our data set, we decided to coordinate our approach so we will be able to cover each of the questions asked step by step. In our initial meeting we quickly realized which milestones we would be facing during this project. First of all, we noticed a lack of inconsistency in our data set that we will further elaborate in our first part of this summary. In our second step we outlined which algorithm will be the most suitable to apply in this project. Here we would measure the performance for each technique used to make a comparison. We are going to discuss which performance metric was applied in the corresponding section. Third, we further refine our most accurate model under the conditions such as varying the number of given features as well as adding new features. Last but not least, we would conclude our project with highlighting our achievements but also our limitations.

2 Data Cleaning

Dealing with corrupt or inaccurate records can be vital when constructing a fully functional algorithm. As the data provider explains, we need to consider how we will manage missing values and extreme values through mistypings to improve data quality. To overcome these problems, we have therefore asked ourselves the following questions: How much data is missing? Based on that, should we rather remove or estimate the missing data? How will we detect outliers? The following paragraph will explore our process of cleaning the data frame.

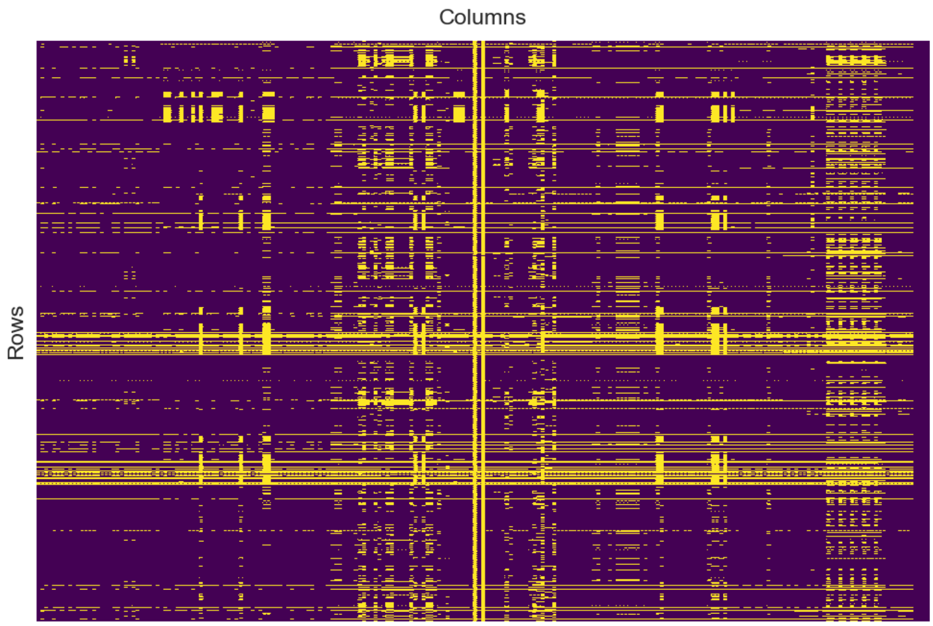


Figure 1 – Overview of our Data Frame (missing values are yellow)

Particularly, we were interested to scan our data frame for missing values to locate trends. By using the Heatmap function from the visualization library Seaborn we were able to observe that some features may be irreparable. Moreover, we felt that some rows lack on validity since there is insufficient amount of information. We set our limit at 50 percent of data which a feature respectively a row should possess. Furthermore, when dealing with the outliers we found that some ratios are just implausible. The author referred the 1% and 99% percentile values which could contain outliers. In order to solve this problem, we have been working with the standardised Z-Score Values. For a normal distribution, to exclude the referred percentile, we will use a standard deviation of 2.33. We simply used the Z-Value which results from a two-tailed test. By using this data cleaning process, 11'579 observations remain from our beginning 22'077 rows. For our remaining undefined values, we will use the median for the corresponding column, in order to attain robustness over the slightly changed and adjusted data set.

3 Machine Learning Algorithm

We divided our approach to find the appropriate algorithm in two steps. First of all, we outlined which algorithm may come into question for this task. The selected models were then examined how they would react to our cleaned data set. We agreed to examine the performance of the following algorithms: Linear and Quadratic Discriminant Analysis, Decision Trees, and a simple Artificial Neural Network model. We measured the performance based on accuracy since this indicator enables us to see if our model can surpass the benchmark where an algorithm just always guesses the most common classification. Secondly, from all the tested models we chosen the model which performed the best. This model should then be analysed under varying the number of given features and newly created features.

3.1 Linear and Quadratic Discriminant Analysis

LDA presents a similar method compared to logistic regression where variables can express a categorical predictor. Further the methods of both classification procedures provide similar results under assumption of normally distributed variables; however, there are some fundamental differences regarding the methods and the characteristics. LDA for example delivers fewer sensitive results, especially when the assigned classes are strongly separated of each other (James et al. (2017)). Compared to logistic regression, where more instability is involved, this is certainly an advantage of LDA. Another important characteristic of the model is the ability to predict when more than two response classes are applied.

For our project, we firstly fed the LDA model with non-standardized and standardized data. Because it is assumed that the features are approximately normal distributed, we thought that centering the mean at 0 with a standard deviation of 1 could give us better predictions. However, the test score difference between non-standardized (54.4%) and standardized (54.8%) was minimal. Trough upsampling we tried to balance the data because there were a lot more sell responses than hold or buy responses. After balancing, all three classes had approximately 6000 entries. However, the test score with standardized balanced data (45%) was not as good as standardized data (54.8%)

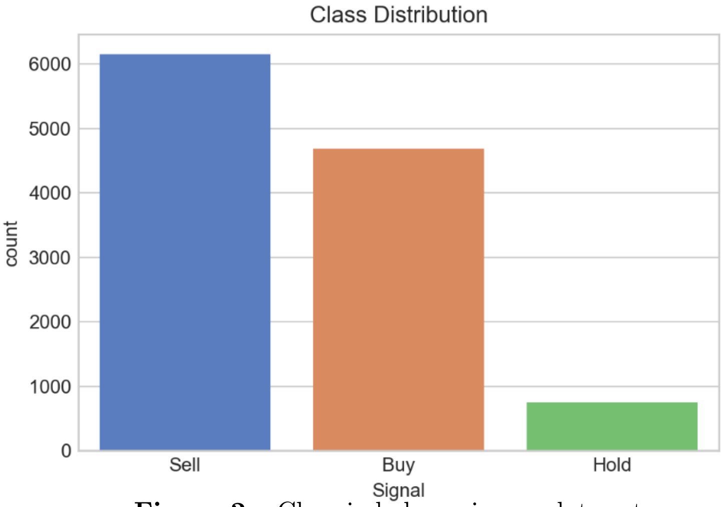


Figure 2 – Class imbalance in our data set

3.2 Decision Tree

Decision trees can be used for both regression and classification problems. This algorithm has several advantages but also disadvantages compared to other algorithms. James et al. (2017) allude that trees in general can be easily explained to people, are able to reflect human behaviour, and can be portrayed. On the other hand, decision trees do not have the same level of predictive accuracy. This, however, are we going to adjust by the use of random forests, a powerful prediction model which builds on decision trees.

We will use a classification tree since we want to predict a qualitative response variable. To measure the classification error rate for our decision tree, we will use the "Gini Index" and the "Cross Entropy", which both work well as a criterion in our prediction. We see that our model has an accuracy of 53%. Without any adjustments, this decision tree would perform as good as a model which would only guess on Sell, which would also result to 53%. Nevertheless, this is without any adjustments.

Random forests are able to handle problems of high variance which could result of having strong predictors in our data set (James et al. (2017)). This strong predictor will persistently influence further decisions in our process. Similar to bagging, random forest will try to improve decision trees by using multiple trees which will result in a majority vote. Essentially the information gain occurs because the trees will be decorrelated by removing the strongest predictors. Additionally, random forests will allow each tree only a subset of the available predictors in order to avoid similar trees.

By the use of random forest, we were able to achieve accuracy of our model of 58%. Therefore, this adjustment enabled our model to outperform our benchmark.

3.3 Artificial Neural Network

For our project, we were also interested to implement an artificial neural network algorithm. Regarding of this model, we made several assumptions for instance in the depth of our model, number of epochs as well as in the percentage of dropouts to prevent overfitting. This resulted to an accuracy of 56.2% which is greater than a default guess of 53% on Sell. We can see that our model is able to make several right decisions when classifying the test score.

3.4 Comparison

Following our strategy, we therefore select the algorithm random forest, which was able to have an accuracy of 58% without any major adjustments.

4 Optimizing our Selected Model

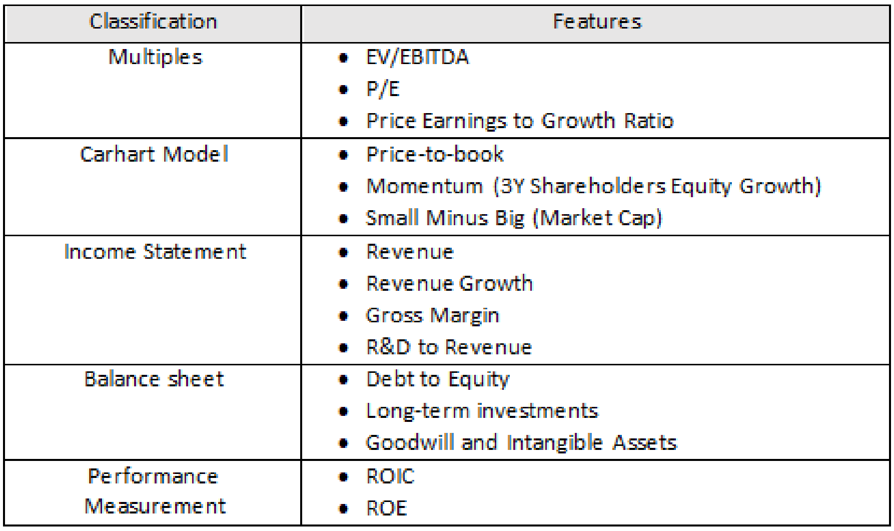
In this section of the summary, we will use various optimization approaches to improve our algorithm. We came up with several possibilities to adjust the number of features but also weaknesses that we have not tackled before. If one adjustment increases our accuracy, we will keep this adjustment for the following adjustments.

4.1 Removing Features based on our Academic Education

In this approach we wanted to examine whether our model can improve when reducing the number of features based on our academic education. We came up with five groups of features. First, valuation profoundly works with market and transaction multiples. Therefore, we included multiples such as EV/EBITDA or P/E. Secondly, since the Carhart Model is able to show significance for each of its inputs, we included features that were close to the ones used by the model. Furthermore, we added predictors that show characteristics about the firm based on the income statement, balance sheet and efficiency indicators.

This resulted to an accuracy of 56% compared to our initial model of 58% of accuracy. Since random forest works quite well with many features, we believe that this is why this approach is not able to show better performance. However, it is interesting that 15 features show nearly as good performance as 230 columns.

Table 1 – Features based on our academic education



Classification	Features
Multiples	• EV/EBITDA • P/E • Price Earnings to Growth Ratio
Carhart Model	• Price-to-book • Momentum (3Y Shareholders Equity Growth) • Small Minus Big (Market Cap)
Income Statement	• Revenue • Revenue Growth • Gross Margin • R&D to Revenue
Balance sheet	• Debt to Equity • Long-term investments • Goodwill and Intangible Assets
Performance Measurement	• ROIC • ROE

4.2 Analysing the Problem of Imbalance

As we already saw in Figure 2, we observe that our data set is heavily skewed towards the Sell classification. To balance out our minor classifications, we use the resample function from Scikit-Learn to increase the number of samples from the classifications buy and hold to the number of measured sell classification. This led to an accuracy of 83%, which is above of the non-optimized accuracy of 58%.

When checking our resulted model, we were really suspicious of the high precision score of 91% for the classification Hold. Prior to the adjustment of balancing the data, our model suffered to classify this class and scored 0% in precision. We were not sure if the approach in the lecture notes on page 143 and 144 could lead to data leakage on our trained model. Especially when up sampling the whole data set and then undertake the split between train and test data, our fitted model would have the advantage to work with replications of a small minority. We also checked if our model changes if we first undertake the train test split and afterwards up sampled our trained data. This would result to a way lower accuracy of 55% compared to 83%. Resulting from that, rebalancing will not increase our accuracy.

4.3 Creating new Features with PCA

With the use of PCA, we wanted to observe if our algorithm performance increases if we create a low dimensional representation of all features. In this process, we decided to take the ten most meaningful features that are created with PCA. This step resulted to an accuracy of 54%. Therefore, this adjustment did not increase our accuracy.

4.4 K-Fold Cross Validation

K-fold Cross Validation enables to create a more stable solution, where our trained model is trained in several iterations. This way of model evaluation led to an accuracy of 56%, which is still below of our non optimized model.

4.5 Macroeconomic Data

Up until now, we did not actually use new data besides the given data frame. Hence, we will add to our data frame following categories of macroeconomic data: money market, product market, labour market, further leading indicators, and lagged data (more information about the features can be found on the excel sheet “Macroeconomic data”). The data was drawn from the Bloomberg Terminal, Federal Reserve, and U.S. Bureau of Economic Analysis. This resulted to an accuracy of 62%, which means that our algorithm can profit from new information.

5 Conclusion

This project enabled us to gain an insight into planning, executing, and verifying a project in machine learning. We were able to learn various concepts in practice and how to effectively work when solving questions that we have never face that way before. We are quite happy that this course enabled us not just to work with the acquired tools, but also to think outside the box when seeking solutions. To conclude, we would like to add some limitations which we had during this project.

We believe there would be more efficient ways of cleaning the data frame. However, this gave us a much better feeling throughout the following steps, since we knew that errors during the process did not refer to our data frame.

Having a more consistent data frame over all the years could probably help us to analyse trends over several periods. This, however, would have brought a lot of adjustments to our current cleaning strategy.

Bibliography

[1] Bloomberg, 2021, World Economic Statistics (ECST) – United States, 08.04.2021

[2] Economic Research - St. Louis Fed, 2021,30-Year Fixed Rate Mortgage Average in the United States, 08.04.2021

[3] Federal Reserve, 2021, Industrial Production and Capacity Utilization - G.17, 07.04.2021

[4] James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani, 2017, An introduction to statistical learning. With applications in R (Springer; Springer Science+Business Media, New York).

[5] U.S. Bureau of Economic Analysis (BEA), 2021, GDP and Personal Income, 07.04.2021

[6] U.S. Bureau of Economic Analysis (BEA), 2021, International Transactions, International Services, and International Investment Position Tables, 07.04.2021

[7] Zimmermann, Benjamin, 2021, Machine Learning in Finance, A Gentle Introduction with a Focus on Applications in Python