Clustering Approach to Stock Price Expectation

Devvrat Raghav, Dhruv Sinha and Daksh Baheti

1st December, 2019

Abstract

Creating reasonable and reliable expectations of the future, when faced with micronumerosity, is considered to be the pinnacle of financial forecasting . To obtain reliable forecasts, we employ an unsupervised machine learning algorithm to cluster companies listed in the Standard & Poor's (S &P) index. We use 18 different financial attributes to perform the clustering and then use these clusters to predict individual stock prices of each company using the stock price of other companies within its cluster as independent variables in a regression framework. This approach to stock price prediction not only reduces the amount of information required to perform the task but also reduces the sample required by current techniques to predict stock prices. This paper, therefore, in addition to extending the sparse literature that exists on the intersection of machine learning and stock price prediction, also provides a new approach to thinking about the uses of the former in the financial domain.

1 Introduction

The formation of expectations plays a significant role in financial markets all over the world today. The movement of stock prices, trading volume, time of trading, and other vital aspects are heavily dependent not only on the analysis of historical data of stocks but also on how this data - and surrounding information - is perceived vis-a-vis the stock. A positive outlook towards one stock may lead to a significant increase in its share value, whereas a negative outlook might be enough to tank the price of another stock. Therefore, in order to catch up with Peter Bernstein's famous saying - "The fundamental law of investing is the uncertainty of the future" - it becomes imperative to study the expectations aspect of the financial markets in order to make an informed decision.

One of the most fundamental human traits is that of comparison. To compare and sort things continues to be the basis of all classification based on which decisions are based. For example, an apple that has a black patch is clustered with other apples that display some defects and is then declared unfit for consumption. The same analogy applies to a majority of physical and non-physical phenomena, and it is only logical to conclude that classification into groups forms a very regular basis of decision making and expectation formation.

Putting these two ideas - of expectation formation in financial domain and classification - together yields a very obvious, yet potent, idea of grouping and clustering like stocks and based on the group traits, make decisions about individual stocks. This paper explores the use of unsupervised learning classification technique such as K-mean clustering on a subset of stocks and further proposes to use this clustering to predict the movement of a stock based on its peers. The next section provides a review of related work in this field and is followed by the methodology and data section. We then report the results obtained and provide a brief discussion of these results. We then list the limitations of this study and also propose solutions to said limitations. The final section concludes.

2 Literature Review

Clustering using unsupervised learning algorithms and expectation management have been, in their respective fields, studied at length. However, the intersection of these two fields has attracted sparse attention from academic and professional spheres. At the time of writing this paper, we could only find a handful of literature that has addressed the topic of clustering financial stocks using unsupervised machine learning algorithms. Even rare is the literature on the use of this clustering as a mechanism to predict future stock

prices. This section elaborates on the current work in this field and establishes the importance of this research.

Rashidi and Analoui (2007) proposed, based on the similarity measure between time series of various financial stocks, using a modified K-means clustering algorithm to cluster stock market companies. They further applied this algorithm to the analysis of companies which are in the Dow Jones (DJ) index to identify similar temporal behavior of traded stock prices. Basalto et al. (2005), however, applied chaotic map clustering instead of the modified K-means to approach the same problem and the same set of companies.

Doherty et al. (2005) used TreeGNG, a hierarchical clustering algorithm, on time series data to identify groups that were neatly clusterable. However, the only attribute used was the closing prices of stocks, thereby making this approach very narrow. Nanda et al. (2010) used stock returns at different times from the stocks of the Bombay Stock Exchange for the fiscal year 2007–2008 in order to achieve the task of management of portfolios. They showed that K-means cluster analysis builds the most compact clusters as compared to other techniques such as SOM (Self Organizing Maps) and Fuzzy C-means for classification of stock data.

Momeni et al. (2015) used financial statement data of three industries in the TSE (Tehran Stock Exchange) for the year 2012 to classify all companies using profit criteria as attributes. AHP (Analytic Hierarchy Process) was used to prioritize the attributes while K-means clustering was used for classification. On the other hand, using probabilistic approaches to determine the location of a company within its cluster, Kuo et al. (2005) developed a variant of K-means that used TWCV (Total Within Cluster Variance) to iterate and find the optimum clusters and their composition.

As it can be observed, while a host of methods have been used in order to cluster and classify companies, there are only a few that have used the clustering to propose further uses. We aim to follow in the steps of the few and use clustering to predict the stock price of a given company based not only on its performance but also the performance of other companies within its cluster.

The most important aspect of this research is that this branch of clustering provides a first pass at classifying new companies, an approach that has not existed before in the financial or the computing domain. For instance, once a company can be clustered based on particular attributes (see the next section for more details), the companies within its cluster can provide a reliable forecast for the future performance of the new company. This approach significantly reduces the amount of information needed to cluster companies and also paves the way for more realistic and broad-based clustering.

3 Methodology and Data

3.1 Dataset

Since the problem of interest is two-fold, we create two distinct datasets to be used in different stages of the pipeline. The first dataset consists of 18 financial ratios computed for each of the S&P 500 Companies. These ratios encompass various aspects of the company, including operational efficiency, profitability, liquidity and the market premium offered by investors on their stock price. Data on these ratios was gathered by scraping online publications of each company's income statement and balance sheet for Financial Year 2017-2018, which were then used as inputs to the relevant formulae. Specifically, we use the following ratios:

1. Price-to-Operating Cash Flows
2. Price-to-Sales
3. Enterprise Value-Multiple
4. EBIT-per-Revenue

5. Net Profit-Margin
6. Return on Assets
7. Return on Equity
8. Return on Capital Employed
9. Receivables Turnover
10. Payables Turnover
11. Inventory Turnover
12. Acid-Test Ratio
13. Cash Ratio
14. Current Ratio
15. Debt Ratio
16. Debt-Equity Ratio
17. Total Debt-to-Capitalisation
18. Dividend Payout Ratio

We then transform these ratios by dividing them by the industry average of that ratio. For instance, Apple Inc. ('AAPL') is categorised as a *Technology* company. As such, each of Apple Inc.'s ratios is divided by the *Technology* industry's average for that ratio. The transformed ratios represent the performance of a company relative to its industry, which is a pseudo-ranking measure. It has a ranking component because the industry average for each industry i is computed using all companies c_i that belong to industry i within our sample of S&P 500 Companies. Since the S&P 500 index technically contains 505 entries, we first removed the additional entry that existed for 5 companies. Specifically, we retained only the *Class A* stocks for Alphabet Inc., Discovery Corp., Fox Corp., News Corp. and Under Armour. This yielded a matrix of dimensions 500×18.

The other dataset created for this exercise consists of concatenated Time Series that contains the Stock Prices for each of the S&P 500 Companies for the three-year period spanning from January 1, 2016 to January 1, 2019. The time interval for these series is daily, resulting in 756 observations for each company. This yields a matrix of dimensions 756×500, with each row representing a date and a column representing a company. We chose a daily time series as opposed to a longer interval to maximise the number of observations without resorting to using prices at an hourly or even a minute-interval, since these tiny intervals are not suitable for an exercise geared towards setting medium-term expectations of price path.

3.2 Clustering

We adopted the **k-means clustering** algorithm for the first stage of this exercise, implemented through the *scikit-learn* library in Python. There were a number of key parameters to be identified, all of which are listed below:

1. n - number of clusters.
2. n_init - number of times the algorithm is run with different centroid seeds.

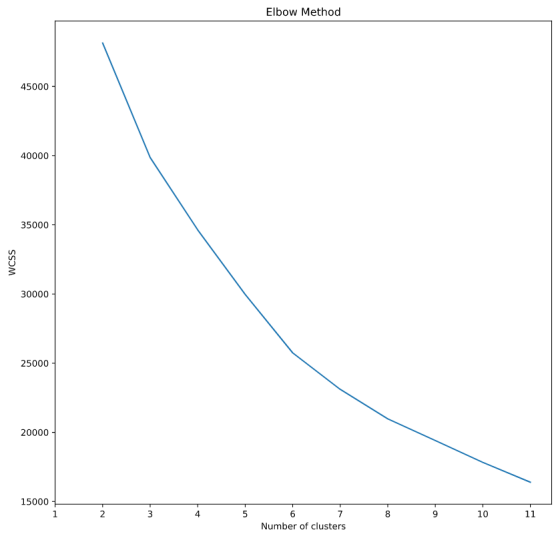
3. *algorithm* - either naive k-means (*lloyd*) or triangle inequality-based variation (*elkan*).

The choice of parameters was processed in a sequential fashion. First, the *algorithm* was set to *elkan* and a grid search was employed to identify the ideal values of n and n_init. These were found to be 7 and 10, respectively. Then, the *algorithm* was set to full (*lloyd*) and this process was repeated, yielding the same results. Since the *elkan* algorithm is well-suited to *dense* data like in this case, we choose it as the solver.

The grid search involved computation of multiple performance metrics used to compare different cluster sizes (n). Three of these required target labels for computation, for which we provided them with the *Industry* of the company. Typically the class labels passed to these measures represent the actual labels that the clustering aims to replicate, however that is not the case here. Rather, we use the *Industry* labels with these methods purely to evaluate how companies from the same industry are being spread across clusters. If anything, it is desirable to have companies in clusters that are generally filled with companies from other industries. From a finance perspective, this process is akin to creating multiple diversified portfolios, since we aim to find similar companies (in a financial sense) across industries. The performance metrics used to compare cluster size (n) are:

1. *Homogeneity Score* - Proportion of clusters that have companies with a homogeneous *Industry* label.
2. *Completeness Score* - Proportion of *Industries* for which all companies are in the same cluster.
3. *V-Measure* - Harmonic mean of Homogeneity and Completeness.
4. *Inertia* - Mean squared distance between each company and its closest centroid.
5. *Silhouette Score* - It is defined for each instance i as $(d_{i,j} - d_c)/max(d_{i,j}, d_c)$, where $d_{i,j}$ is the mean distance to other instances in the nearest cluster and d_c is the distance across clusters.

Figure 1: Plotting Inertia for different values of n



The illustration above plots the *Inertia* for different values of n, and we choose the point at which the curve has a slight kink - a process also referred to as the *Elbow* method.

Ideally, we would prefer a value of n that minimises *Inertia* and maximises the *Silhouette Score*. As a byproduct, it is also preferable if the value of the first three measures is as low as possible. The table below summarizes the observed values of these metrics for different values of n:

Table 1 - Performance Measures for k-means clustering

Number of Clusters	Homogeneity	Completeness	V-Measure	Inertia	Silhouette
2	0.002	0.207	0.005	48142	0.928
3	0.006	0.182	0.011	39866	0.834
4	0.007	0.212	0.013	34636	0.833
5	0.009	0.211	0.018	29967	0.832
6	0.014	0.189	0.26	25742	0.778
7	0.024	0.211	0.042	23105	0.588
8	0.021	0.208	0.038	20974	0.696
9	0.23	0.198	0.041	19410	0.646
10	0.051	0.153	0.076	17826	0.234
11	0.050	0.149	0.075	16394	0.284

Based on the above results, we pick 6 to be the value of n, since the increase in *Inertia* by choosing a smaller value, say 5, is greater than the increase in the *Silhouette Score*. Likewise, by choosing a larger value of n, we witness a greater drop in the *Silhouette Score* than the drop in *Inertia*. Thus, we find 6 to be the optimal value for the number of clusters, given the data used.

3.3 Linear Regression

Once each company is assigned into a cluster, we break up the dataset consisting of stock price data into smaller chunks. Each chunk contains the data just for companies within a single cluster, resulting in a total of 6 smaller datasets, each with the dimension $756 \times N_i$, where N_i represents the number of companies in cluster i (same as the companies in dataset i).

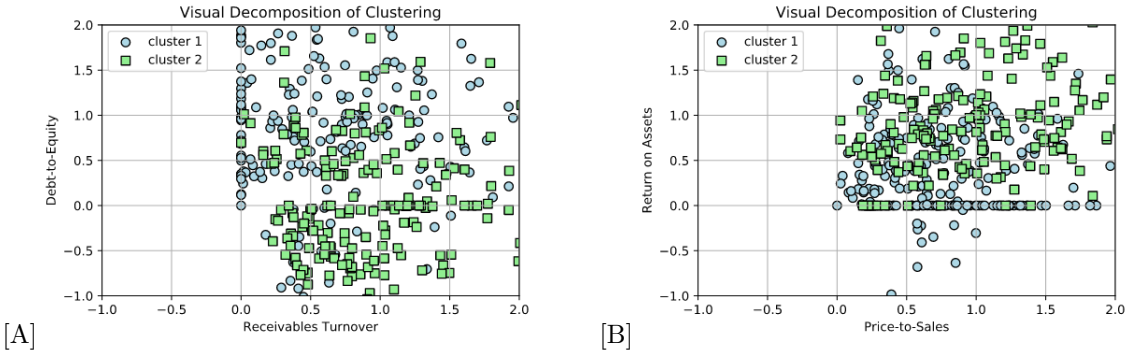
We then operate on each cluster i individually, where we iterate over every single company C_i in that cluster and use C_i's stock price as the target variable y and the stock price for all other companies $C_{j \neq i}$ as the input features X. Thus, y is a vector of length 756, whereas the feature matrix is of dimensions $756 \times (N_i - 1)$. We then pass this data through a **Linear Regression**, used through the *scikit-learn* library in Python. Thus, the algorithm essentially is:

```
for i in clusters:
choose cluster i
     for company j in cluster i:
         y := stock price data for company j
            X := stock price data for all other companies k in cluster i
            LinearRegression(X, y)
```

Once the β coefficients from the regression are retrieved, they are stored in a **weights matrix** for cluster i. Since there are 6 clusters, we make a corresponding set of 6 weights matrices, each of dimension $N_i \times (N_i - 1)$. Thus, the weights for each company $C_{i,k \neq j}$ from the regression of company j in cluster i are stored in row j of weights matrix i. As a result, each company j in cluster i has a total of $(N_i - 1)$ weights. Since each $\beta_{j,k}$ asymptotically represents $Cov(Price_j, Price_k)/Var(Price_j)$, we interpret $\beta_{j,k}$ as a financial similarity coefficient between companies j and k in cluster i.

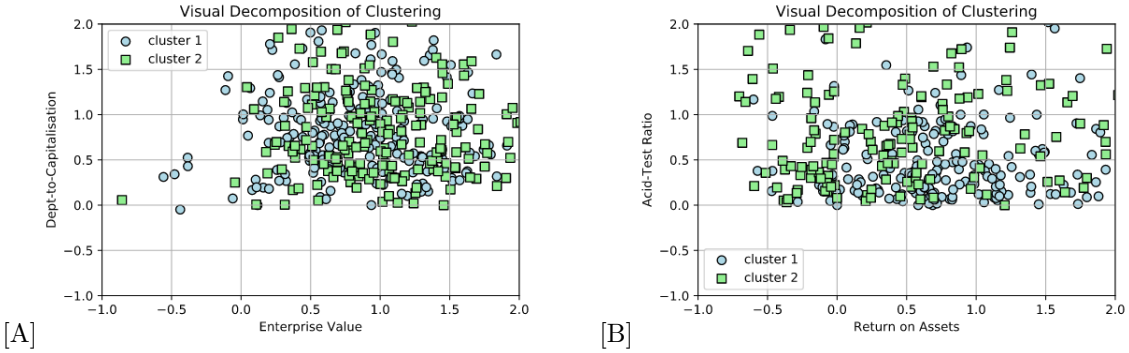
4 Results

In our exploration we find that the data, at least in 18 dimensions, is not very separable. Nonetheless, we attempt to visualise our results by plotting the two most populated clusters on a combination of two dimensions below:



[A] [B]

Figure 2: Clusters decomposed by (A) Receivables Turnover and Debt/Equity (B) RoA and Price/Sales



[A] [B]

Figure 3: Clusters decomposed by (A) EV Multiple and Debt Capitalisation (B) RoA and Acid-Test Ratio

In Figure 2(a), companies with a higher Receivables Turnover and lower Debt/Equity tend to be in Cluster 1, whereas companies with a higher Debt/Equity and somewhat lower Receivables Turnover generally appear in Cluster 2. This indicates that Cluster 1 contains companies that are more efficient in getting their short-term payments from clients and do so without the overhead of long-term debt, indicating that these companies are financially more stable. More simply, these companies are at least above-average performers in terms of financial soundness. On the other hand, companies in Cluster 2 are those with greater Debt/Equity, which makes them more leveraged than the average company. This could indicate companies that are currently growing and have taken debt to fuel their expansion, which could also explain the negligibly worse receivables turnover (longer credit times and longer-maturity repayment schedules could be used by these

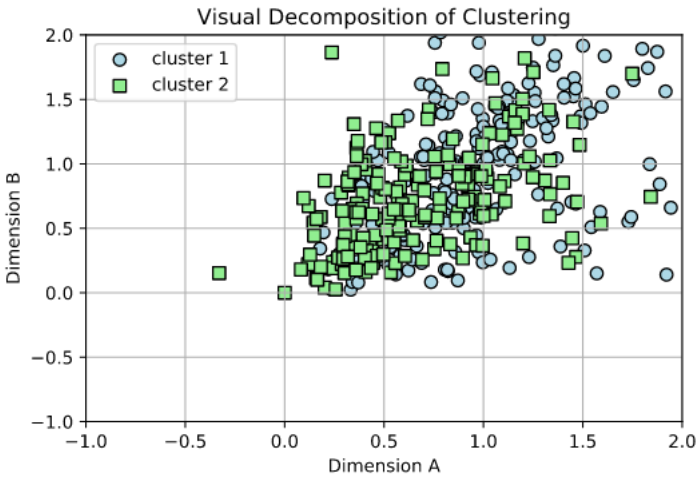
companies to entice their clients - consumers or businesses).

Likewise, in Figure 2(b), companies classified into Cluster 2 are those with a lower-than-average Return on Assets, even though their Price-to-Sales ratio is not markedly different from companies in Cluster 1. This suggests that companies in Cluster 2 are those that did not perhaps make the best available use of their assets—possibly taking risks that did not pay off, or not innovating enough—when compared to their industry averages. In other words, these companies are less efficient with their use of capital than their counterparts in Cluster 1. From an investor's standpoint, companies in Cluster 1 represent a better avenue to grow their investment, which is akin to saying that they are better-than-average performers (thereby explaining the sometimes higher Price-to-Sales that the market is willing to pay to buy into these companies' growth).

In Figure 3(a), however, we observe that there isn't a clear way to distinguish between the differences (little as they may be) between Clusters 1 and 2. It appears that not all measures used in this analysis add great value to the clustering method, particularly ones that are not very heavy-tailed. Conversely, in Figure 3(b), it is once again easier to distinguish between Clusters 1 and 2. Specifically, companies in Cluster 2 tend to have a lower Return on Assets, irrespective of their Acid-Test Ratio values. This indicates that Cluster 1 contains companies that are able to generate higher-than-average tangible returns for a given level of investment, even if that means tying up a larger-than-average portion of their capital in inventory. Thus, companies in Cluster 1 seem to be large-scale enterprises, who handle large volumes of inventory with minimal costs arising from mismanagement, i.e. companies that can take advantage of economies of scale.

We also present the visualisation of the same clusters using Principal Component Analysis to transform the input dimension to 6 dimensions, as opposed to the 18 dimensions initially used. The results are shown below:

Figure 4: Plotting Clusters 1 and 2 on the two principal components of input data



When viewed through the latent dimensions, the separation between Cluster 1 and Cluster 2 is still only partially clear, since there is significant interspersion at the center of the graph. Nonetheless, companies in Cluster 1 score higher on both latent dimensions when compared to Cluster 2 companies, on average. Thus, it is reasonable to say that Cluster 1 broadly contains high-performing companies—at least from a financial standpoint—, while Cluster 2 contains relatively low-performing companies.

Although we present the results only for the two most populated Clusters, the interpretation for the

remaining clusters is similar. Essentially, the clusters can be described as below (number of companies in the cluster is in square brackets):

1. Cluster 1 [241] - high-performing companies
2. Cluster 2 [237]- low-performing companies
3. Cluster 3 [10] - very-high performing companies
4. Cluster 4 [6] - very-low performing companies
5. Clusters 5 [4] and 6 [2] - contain outliers that perform very differently on a handful of financial metrics.

Lastly, for illustrative purposes, for a random selection of 3 companies we list below four very similar companies within their cluster:

Table 2 - List of similar companies

Name of Company	Company 1	Company 2	Company 3	Company 4
AMD	Ecolab Inc.	CIGNA Corp.	Edison International	Best Buy Co. Inc.
Apple Inc.	Alliant Energy Corp	Amphenol Corp	Allstate Corp	BlackRock
Lab Corp. America	CenturyLink Inc.	Citizens Financial Group	Iron Mountain Inc.	Western Digital

5 Limitations

Our approach, in its current state, has two main limitations. This section outlines them and contains proposals to resolve them.

5.1 Sample Selection

We selected the S&P 500 Companies for this project, since together they are representative of the US Stock Market as a whole. However, these are all fundamentally not very different companies, in both scale and performance, since these are large companies that have an enormous market capitalisation. Consequently, there isn't enough variation in their performance across the financial metrics chosen for evaluation in this exercise. As a result, the transformed input features (ratios) are not ranking-based methods, which leads to the companies appearing very squashed in the 18 dimension input space.

By expanding our sample to include many more companies of varying sizes and performance, we would be able to better cluster these companies. This is likely to be the case because adding those companies would change the within-sample industry averages of the different metrics used to measure a company's performance. Since the current sample includes large, successful corporations, the performance of each of these companies—using the new industry averages from a much larger sample—should be enhanced, especially in comparison to smaller or poorly-performing companies within their industry. Thus, performance-based clusters should become more easily to separate, since the variation within these metrics should increase across the board.

5.2 Feature Selection

After visualising the results of the clustering algorithm, it became apparent that there was insufficient variation within the data. Consequently, the clustering performance was not ideal. To alleviate this, we propose expanding the set of input features by including two key aspects of a company's performance:

First, a textual description of their historical and current operations. This text can be sourced from EDGAR filings for each publicly listed companies, specifically from their 10-K filings. After all, these filings also contain a section wherein the company management must give their verdict about their medium-term expectations of their company's future. By mining this text, vectorizing it through TF-IDF and applying Latent Semantic Analysis to it, we should be able to address the qualitative aspect of a company's operations and create better clusters that would contain only companies that are similar from both qualitative and quantitative standpoints.

Second, by projecting the financial metrics onto a higher dimension space to improve separability within the sample. This projection can be further augmented by an aggregation of the investor sentiment of the company across various stock market advisory services, such as Zacks Advisory. The sentiment in question is a categorical variable from a defined scale, which should also be a dense representation of experts' expectations of the company's near-term future. Adding this to the analysis should create a truly comprehensive feature space that should result in much improved clustering.

6 Conclusion

Our primary aim for this exercise was to find companies that were similar to a given company, allowing a potential investor to use those companies as a means of forming and managing expectations about the company's future stock price path, particularly for new companies with limited past data. We find that it is possible to cluster companies into performance-based categories, such that a company's stock price path can be semi-reliably predicted using other companies in its cluster. Nonetheless, we do not assert that this framework necessarily allows an individual to predict the stock price of a company at a particular point in time. Rather, it allows investors to potentially use a larger (in a temporal sense) dataset of historical stock price data to chart expectations for a new company, i.e. create a focused confidence interval for its future price path. Thus, we believe that this research, if effective, can help assuage a key problem in time series forecasting - overcoming the small-sample problem to reliably manage expectations.

7 References

Basalto, N., Bellotti, Roberto, Carlo, F., Facchi, Paolo and Pascazio, S. "Clustering stock market companies via chaotic map synchronization". *Physica A: Statistical Mechanics and its Applications*, 2005, issue 345, pp. 196-206.

Doherty, Kevin, Adams, Rod and Davey, Neil. "TreeGNG - Hierarchical topological clustering." *ESANN 2005 Proceedings - 13th European Symposium on Artificial Neural Networks*, 2005, pp. 19-24.

Kuo, R.J., Wang, H.S., Hu, Tung-Lai and Chou, S.H. "Application of Ant K-Means on Clustering Analysis." *Computers and MAthematics with Application*, 2005, issue 50, pp. 1709-1724.

Momeni, Mansoor, Mohseni, Maryam and Soofi, Mansour. "Clustering Stock Market Companies via K-Means Algorithm." *Kuwait Chapter of Arabian Journal of Business and Management Review*, 2015, issue 4, pp. 1-10.

Nanda, S.R, Mahanty, Biswajit and Tiwari, Manoj. "Clustering Indian stock market data for portfolio management." *Expert Systems Applications*, 2010, issue 37, pp. 8793-8798.

Rashidi, Parviz and Analoui, Morteza. "Modified k-means algorithm for clustering stock market companies", *Iran University of Science and Technology*, 2007.