A Visual Analytics Approach to Stock Market Data

Ahmed Awny

Abstract—In the financial world, financial time series data is massively generated, where it is highly complex yet highly needed. Being so, solutions do exist in mining patterned structure with-in aspects of the data across various features. This report presents an analysis process in attempting to determine a structure through dimensionality reduction and clustering visualisation for an aspect of interest, the stock-price daily return. Once validated for structure, it is then proceeded with applying a visual technique that enables a viewer to easily indentify characteristics of price variations across companies and industries. It ultimately visually clusters companies' time series daily return data according to their similarity. This helps to assess the behavior charactertics of companies in same cluster, different industry and of companies in same industry, different cluster where behaviour uniqueness and un-precented patterns across un-precedented industries can be identified.

1 Problem Statement

In the real world, looking at historical data of stock prices is very important to make investment decisions however just looking at a time series stock-price representation for any given company won't give much insight. As an investor you want to buy in the right time and sell in the right time and more importantly buy in the low-risk and highly-anticipated companies. But in-order to do so a good understanding of the underlying trends of the historical prices is needed. In theory, one would have look at every single historical time series for all the potential companies and try to determine a pattern for the company as well as a possible pattern between other companies in the same industry and even companies outside its industry. The only way to understand how a company is performing is by comparison to the general pattern. So, in this analysis, it is attempted to produce visuals as a result of clustering companies in the NYSE and NASDAQ based on stock-price movements (daily-return). The main question that is tried to be answered in this analysis is; by calculating the daily returns for each company, does it necessarily mean that all companies of the same industry will fall under the same cluster based on the daily-return, if not, does that mean we can conclude that companies of different industries have similar stock-price movements and what does this mean? Is there any pattern in the price-movements of the clustered companies in specific time-frames? Also, other possibilities of insight will be discussed in the reflection section.

2 State of the Art

The main inspiration for the visual analysis in this paper was from an analysis by researchers from the University of Konstanz [1]. They focused on explorative analysis and visualization of large information spaces in the financial domain (stock market data). In one of their visual methods they have proven to cluster companies in the same/different industry based on stock price similarity. How they visualised it was indeed very interesting. They allowed the cluster centroid (graph) for each cluster to be at the top as a column head, and below it in that same column are the company stock sequences that match the given centroid the closest (essentially the ones that have fallen in that respective cluster). It was fascinating to see how stocks of different industries were clustered according to the trajectories of their raw stock-price time series data. In this analysis we try to replicate this but with the daily return. Here the power of clustering is really felt and especially visualising the outcome of clustering in a meaningful way can really extract patterns and trends hidden in immensely large and complex datasets. For their clustering [1] they used both the k-means and hierarchal methods.

Moreover, the accuracy of these clustering algorithms is important in any real-life scenario, as decisions can be made based on the respective visuals. For this reason, an effective range of clustering methods should be performed and compared when validating any cluster output.

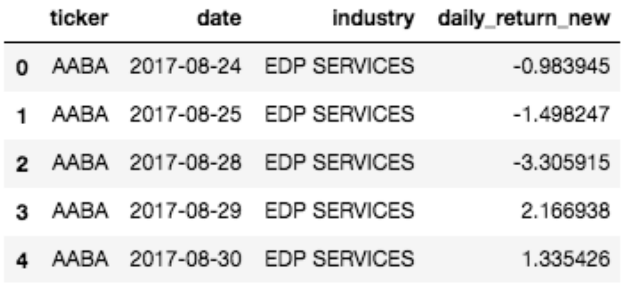
In a second paper, Jianfei Wu [2] introduces an algorithm that uses stock industry information directly in parallel with time series sub-sequences for mining intrinsic patterns with-in the respective industries of the stock data. He uses the stream sliding window concepts. A training window and an evaluation window were used adjacently. In the training window, the algorithm detected significant industries, and built key patterns for them respectively. Whether an industry shows coherent behavior is recognised by the algorithm. When coherency is detected, fundamental patterns are extracted. Those patterns were more stable when compared to clusters found by the DBScan algorithm. Through this comparison, his proposed algorithm proved effective.

3 Properties of the Data

The datasets were found on Kaggle. One contained the stock prices (ticker, open, close, adjusted close, low, high, volume and date). The other dataset contained features inclusive of the ticker, exchange market (NYSE or NASDAQ), company name, sector and industry. As this is mainly based on time series analysis, the data is exclusively either numerical, strings and datetime objects. The datasets are merged on the ticker feature, which exists in both datasets and later feature engineering will be present to narrow data needed for analysis. Originally the stock data time-frame ranges from (8/05/2013 to 24/08/2018), however this will be narrowed down to the final year (24/08/2017 to 24/08/2018) to make the tasks more manageable in-terms of time. This is due to the dataset originally having 20,973,889 (21 million approx.) observations. By taking this 1-year date range, shrinks down the shape to 1,368,161 observations. The number of observations was shrunk further to 494,610 as a result of the explanation in the next paragraph.

It was realised that a lot of null values existed in industry feature. To avoid this the data was filtered respective to the top 12 industries based on the counts. This solved the null value problem as well maintained the data for the most recurring industries, which has a positive effect on the certainty of the analysis.

After several steps of feature engineering and data-frame merging, this is the final look (fig. 1 below) of the data before transformation (pivoted; explained later).



	ticker	date	industry	daily_return_new
0	AABA	2017-08-24	EDP SERVICES	-0.983945
1	AABA	2017-08-25	EDP SERVICES	-1.498247
2	AABA	2017-08-28	EDP SERVICES	-3.305915
3	AABA	2017-08-29	EDP SERVICES	2.166938
4	AABA	2017-08-30	EDP SERVICES	1.335426

Fig. 1. Pre-transformation data-frame

The daily return feature seen above will be introduced later in the report. Outlier detection is performed on it (see fig. 2 below). To do this the interquartile range method was used.

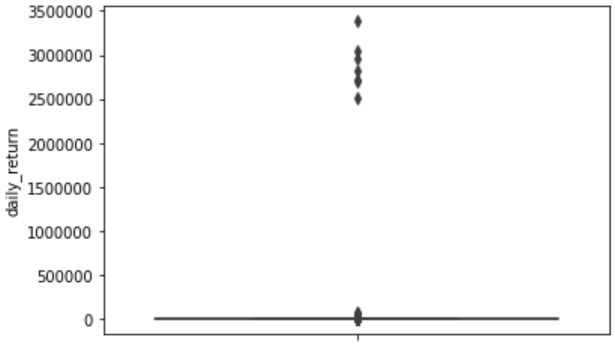


Fig. 2. Boxplot of the daily return data before outlier handling.

The boxplot is not visible due to the deviation of the data due to the outliers. The outliers are not removed but replaced with the upper and lower whisker values of the daily return data. There was a total of 49,783 outliers detected out of 494,610 daily return values (essentially the number of observations).

4 ANALYSIS

4.1 Approach

The objective is to make use of the data with-in the perspective of Visual Analytics to help satisfy the research questions.

Just by looking at the data with the naked eye, some columns will not be used with the analysis. As the analysis is based on the industries, attributes such as the exchange market (NYSE or NASDAQ) and company names are not needed and redundant (given that the ticker feature is available) respectively.

Data pre-processing and feature engineering

Initially we want to pick industries with the top ticker counts. The count of the industries of the distinct tickers will then be used to determine the top 12 industries, which will be taken into analysis given their high relative proximity of occurrence and the rest will be disregarded.

Next comes engineering a new feature, the daily return. In this analysis, the daily return is what will determine the stock-price movements. It will be calculated for each observation respectively.

At this point a new data-frame will be crafted to only include the variables that will be needed for the next phase; dimensionality reduction and clustering. However, before entering the next phase a final transformation is needed for this new data-frame. The data needs to be transformed such that each unique value in the 'date' column becomes an individual feature, where each date feature will represent its stock-price return value for each ticker (company) and its respective industry. All the date features will therefore represent all the daily-returns across the respective date range (1-year) for any given ticker-industry combination. Each row will therefore have ticker, industry and all the respective daily returns across the whole year.

Any categorical data will be removed temporarily when applying clustering and dimensionality reduction as the respective algorithms with-in cannot handle categorical data, only numeric.

Dimensionality reduction and clustering

In this phase, two dimensionality reduction techniques will be applied. The first is Principal Component Analysis (PCA) and the second is t-Distributed Stochastic Neighbour Embedding (t-SNE). This is two compare the effectiveness of the distribution of the daily-return data with-in three components. The k-means clustering algorithm will then be applied to determine cluster labels for the daily return data. The components from the best dimensionality reduction performer will be used to super-impose the k-means labels' output on the reduced information and visualise the clustering result. The best k parameter (no. of clusters) will be chosen through the elbow-method based on the model inertia versus a varying k plot. Dimensionality reduction and clustering is important to prove the potential industry-based or cluster-based groupings of the data based on the daily return to justify moving forward with the analysis and making the effective visuals.

A final plot will be visualised showing a sample example of the aim of the analysis. This is by combining the respective time series graphs of the tickers' (companies') daily return in the same subplot column if they are in same cluster and color-coding each company graph based on the industry it belongs to.

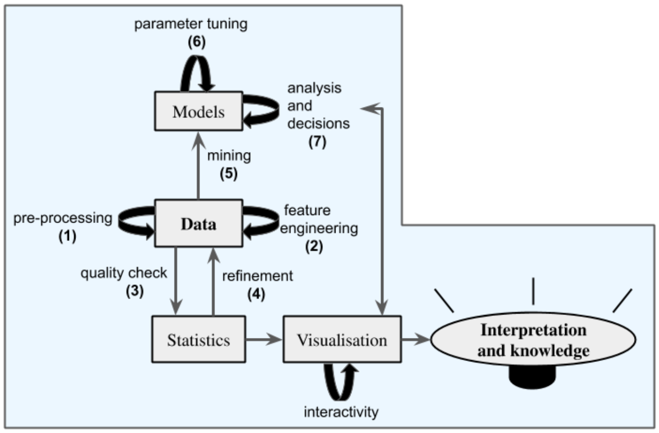


Fig. 3. Flow diagram showing a summary of the process procedures

4.2 Process

This section is used to describe the visual material that has been extracted to make decisions as-well as represent outcomes respective to the approaches specified in the section before.

Data pre-processing and feature engineering

After the data is merged on the ticker and filtered based on the time-frame that is wanted, the top 12 industries are to be determined as mentioned in the previous section. To do so we want to select the distinct ticker rows for each industry. Note that the data contains duplicate tickers based on distinct dates, however we want to eliminate this so the duplicates of all tickers for any given industry are dropped and the remains are put into a new data-frame. The 'value_counts' function is then used to determine the number of tickers for each industry (essentially the number of companies in each industry).

This data-frame is then finally merged with the original data-frame on the industry feature, where all the non-inclusive industry rows were cancelled out.

As for the new daily return feature, it will be calculated through computation for each observation using the following equation:

$$R_t = \left(\frac{I_t}{I_{t-1}} \times 100\right) - 100$$

where

$$R_t = index\ return\ for\ time\ t$$

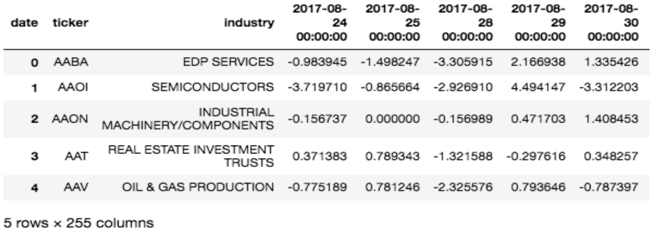
$$I_t = index\ level\ at\ time\ t$$

$$I_{t-1} = index\ level\ at\ the\ previous\ period\ time\ t-1$$

In the case of this analysis, the data was lagged by 1 row in the negative direction to obtain the day ahead information in the current row respectively. The open stock-prices were used to make the calculation. In relation to the equation above, for any given row or current day, $I_t = open\ price\ (1\ day\ ahead)$ and $I_{t-1} = open\ price\ (today)$, and with these values the calculation was made where

$$daily\ return = \left(\frac{open\ price_{t+1}}{open\ price_t} \times 100\right) - 100$$

Transforming the data for the clustering analysis required using the 'pivot_table' function in Python [3]. This function has allowed the date feature observations to distinctively become the features, indexed on the ticker and industry (see fig.4 below)



date	ticker	industry	2017-08-24 00:00:00	2017-08-25 00:00:00	2017-08-28 00:00:00	2017-08-29 00:00:00	2017-08-30 00:00:00
0	AABA	EDP SERVICES	-0.983945	-1.498247	-3.305915	2.166938	1.335426
1	AAOI	SEMICONDUCTORS	-3.719710	-0.865664	-2.926910	4.494147	-3.312203
2	AAON	INDUSTRIAL MACHINERY/COMPONENTS	-0.156737	0.000000	-0.156989	0.471703	1.408453
3	AAT	REAL ESTATE INVESTMENT TRUSTS	0.371383	0.789343	-1.321588	-0.297616	0.348257
4	AAV	OIL & GAS PRODUCTION	-0.775189	0.781246	-2.325576	0.793646	-0.787397

5 rows × 255 columns

Fig. 4. Transformed data-frame

See how the data-frame was transformed compared to the original data-frame in section 3; properties of the data, above. After this transformation, un-explained null values (NaNs) were introduced. The forward-fill and backward-fill methods (based on the columns) were applied together and filled all the null values with a fairly good estimate.

Dimensionality reduction and clustering

***note to reader:** each point on all forthcoming scatter plots is a different company and each shade or colour is a different industry.

For principal component analysis (PCA) scaling is usually needed but since we are dealing with only one variable (daily return), this step is not needed. Upon performing PCA, the explained variance for each component was plotted to determine the weighting of information across the 6 components.

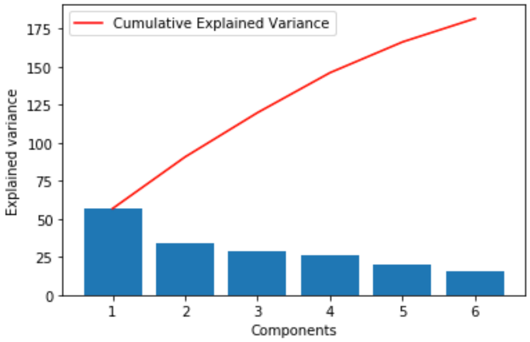


Fig. 5. Explained variance versus PCA components.

As shown above the explained variance is highest in the first component, and at the second component, is first biggest drop seen. It has been proceeded with using three components to maximise the cumulative explained variance as well as maintain the visualisation aspect.

As it is evident below the 3-component PCA plot to the right projects the distribution of the data much better than the 2-component PCA plot to the left.

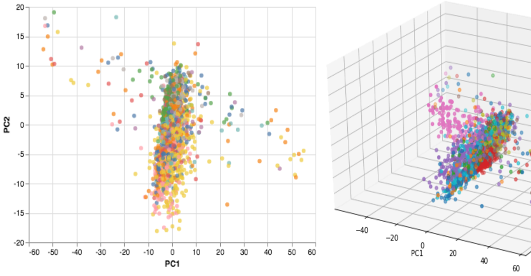


Fig. 6. A) 2-component PCA plot. B) 3-component PCA plot.

The fact that PCA transformation is not effectively color-grouped by industry is because it is not necessary that the daily return of companies in the same industries will be similar. On the other hand, the fact that there is some good evidence of color-grouping shows that a-lot of companies in the same industry do have similar price movements.

Below is the t-distributed Stochastic Neighbour Embedding (t-SNE) transformation with two (left) and three (right) components. The 3-component transformation for the t-SNE portrays the projection of the data much less than the 3-component PCA, but still shows considerable evidence of a patterned structure with-in the daily return data in conjunction with the industries.

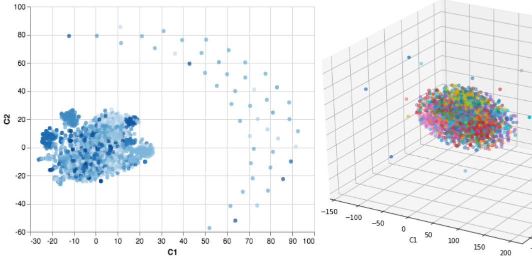


Fig. 7. A) 2-component t-SNE plot. B) 3-component t-SNE plot.

Clustering was the next step in the process. The k-means algorithm is fitted onto the transformed data. All categorical and non-numerical variables are removed since k-means only works as expected with numerical-only data (daily return data only), in this case looking at fig. 4; ticker and industry columns were consequently dropped. The optimal number of clusters (k) was decided using the elbow method based on the model inertia.

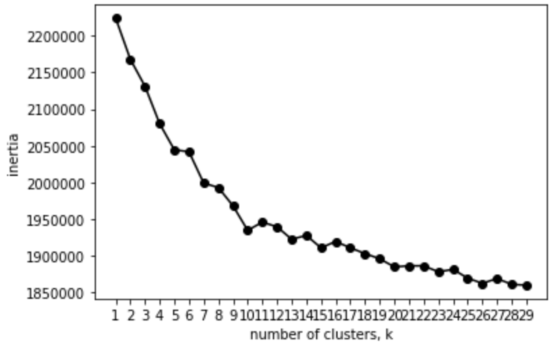


Fig. 8. Elbow method; optimal k value

From the graph above (Fig. 8), it is evident that after the 11th cluster the change in the value of inertia is no longer significant. With that being so, the k-means model is built with 11 as the k value.

The clustering output is visualised by being super-imposed onto two and three components from the PCA producing a 2D and 3D plot respectively as shown below.

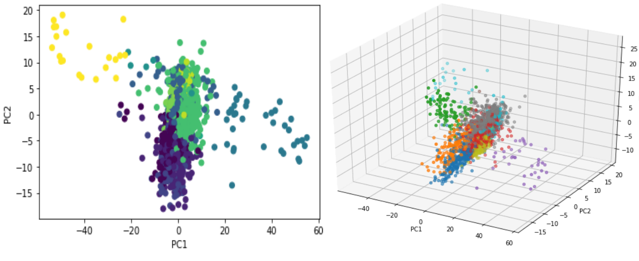


Fig. 9. A) 2-component PCA k-means cluster-analysis plot. B) 3-component PCA k-means cluster-analysis plot.

As visible from the plots in fig. 9, a cluster-analysis based on three components (B) determines clearer and well-grouped clusters. This is mainly due to that three components holds more information cumulatively making the respective plot more defined. Plot B also portrays a more sophisticated projection of the data where the clusters are more noticeable.

Finally, the super-imposition of the k-means output on the t-SNE components showed interesting visuals as-well, relaying an un-supervised structure with-in the data.

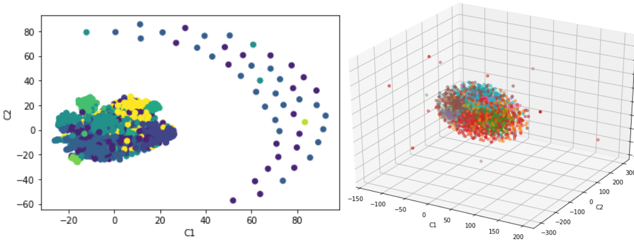


Fig. 10. A) 2-component t-SNE k-means cluster-analysis plot. B) 3-component t-SNE k-means cluster-analysis plot.

Given this analysis, in-terms of the engineered feature (daily-return); it can be confirmed that there is indeed an un-supervised structure of the daily return data. Superimposing the cluster output on both of the dimensionality reduction techniques gives an additional degree of confirmation that an un-supervised structure exists. In-terms of PCA and t-SNE plots (fig.6 and fig.7), there is a mediocre grouping of the data. However, that doesn't mean there is something wrong with the data, but purely because indeed not all companies with-in the same industry have similar stock-price variations, hence the colours are not well-grouped together. But given that un-supervised learning (k-means) analysis gives much better grouped data (grouped coloured points) signifies that there is indeed a structure with-in the data based on the daily return but not entirely based on the respective industry of each company.

This nudges an answer to the research question that indeed some companies of different industries will-fall under the same cluster; which is proven by the noticeable colour grouping difference between fig. 9B and fig. 6B.

This theory will be visualised and critically analysed in the results and reflection sections below. The visualisation method will be based subplots being placed in a specific figure column based on conditional statements of which cluster does the respective row of data belong to as well has color codes using a dictionary to represent the industry of the respective company. Only two clusters of a sample of the data considered and visualised for space and visual-clarity purposes.

4.3 Results

A final plot is made to visualise the clusters from a time series perspective. This plot is found to be very interesting because of the visualisation technique used for the clusters.

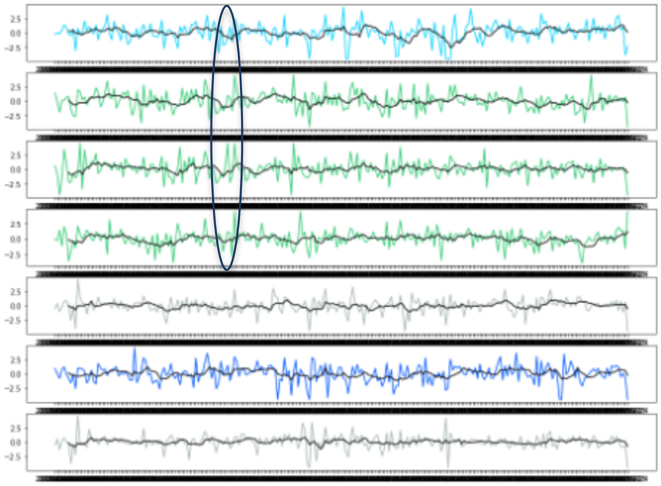


Fig. 11. Time series visual representation of companies' daily return with-in cluster 2.

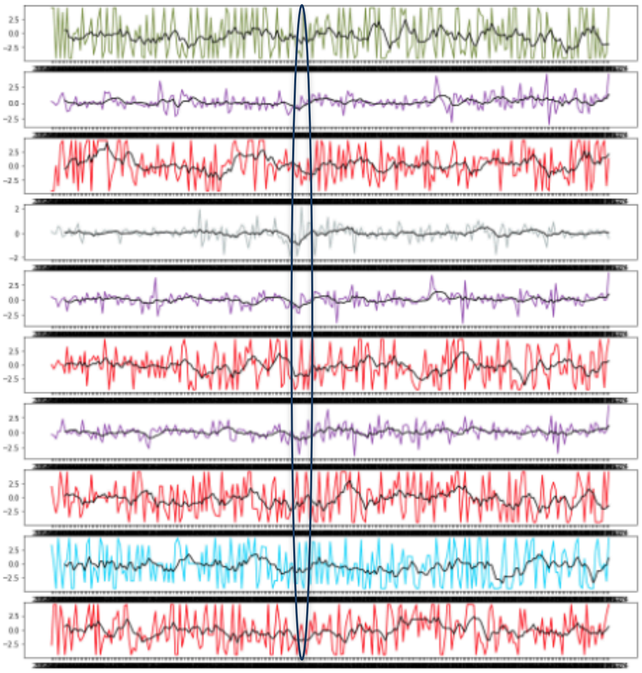


Fig. 12. Time series visual representation of companies' daily return with-in cluster 3.

The plots above are very good visual representations of time series clusters in a way the it makes it easy to draw time-defined suppositions from them, and it was indeed one of the main intentions of this analysis; signifying the importance of such plots in helping shape conclusions and give users insight. Each column of sub-plots is one cluster. For this example, two clusters are shown for a sample of the first 50 rows of the data. Each graph represents a different company. The colour of the line plot in each graph represents the industry of which the company belongs. The coloured line is the raw daily-return data. The black line on each graph is the moving average of the raw daily return data for 7-day windows.

As evident from the plots, the sequences in the graphs of each cluster show a strong degree of homogeneity and/or coherence.

5 Critical reflection

In this analysis, it is mined for a patterned structure of an engineered feature (daily return) of companies in conjunction with market-sector data. Dimensionality reduction and clustering has proved effective in outlining a worth-wile structure in the daily return data through visualisation.

Performing the moving average on the data was a very important step in advancing the final visualisation. It is a key aspect of soothing a noise-filled plot helping in deducing patterns in the visuals.

Taking a look at the visuals in fig. 11 and fig. 12, the time range was supposed to be on the bottom of each subplot however due to complications with the library functions, they could not be binned without affecting the plot shape. However, the time-range is 1-year between 24/08/2017 and 24/08/2018.

The advantage of these plots is that each represent a cluster, so they are highly likely to have similar price-movements. Also, the graphs in each cluster are underneath each other, which helps in identifying parallel trends with-in a specific time-frame.

Looking the plots again, a trend that was noticed for each cluster is circled. If a closer look is taken at the moving-average sequence in the circled time-frames, a pattern across different companies is noticed. There is a homogeneous drop in the daily return sequence with-in a very similar time-frame, and a more coherent sequence drop when the identical industries are looked at (i.e. identical colour). This deduction projects several hypotheses that analysts can dig into. Analysts can take a closer look at the news from various sources during this specific time-frame. If the news does actually give reason for this drop in daily return, then it be asked; what kind of news is it? Given that this news affected several companies across different industries the same way, then a proposition can be made that based on a specific-kind of news, industries (x, y, z) are similarly affected. Furthermore, this can also help with investment decisions, for example understanding what are the factors that have such a well-rounded effect on companies, therefore helping determine portfolio risk based on the chances of the same or similar factor occurring. It can also help identify which investments are the most attractive or the most stable industries. It's with this kind of dynamic thought process where you can get great insight from such plots.

Another interesting perception is that looking the blue graph in the two plots, they are companies of the same industry but have fell under different clusters, cluster 3 (fig. 12) appears to be more volatile than cluster 2 (fig. 11), where the same can said about these two blue companies. Here it can be checked what makes companies in the same industry behave differently by stock-price variations; e.g. company size etc. and validate with other companies in same industry through the rest of the data (i.e. which behavior is considered unique and why).

An interactive version of the plots will be much better to easily relay graph information (i.e. company name, industry as well as the date and daily return respective to where the cursor hovers over the line).

Table of word counts

Problem statement	257
State of the art	348
Properties of the data	349
Analysis: Approach	507
Analysis: Process	1145
Analysis: Results	203
Critical reflection	518

REFERENCES

[1] H. Ziegler, M. Jenny, T. Gruse and D. Keim. Visual Market Sector Analysis for Financial Time Series Data. In Proceedings of the *IEEE Symposium on Visual Analytics Science and Technology*, pp. 1–8, Oct. 2010.

[2] J. Wu, A. Denton, O. Elariss and D. Xu. Mining for Core Patterns in Stock Market Data, *IEEE Conf. Data Mining Workshops*, pp. 1-6, Dec. 2009.

[3] Python.org. (2019). *Welcome to Python.org*. [online] Available at: https://www.python.org/ [Accessed 22 Dec. 2019].