White Paper

9 February 2015; revised 5 April 2016

Visualizing Financial Market Data in Three Dimensions

Ryan Stauffer
ryan.p.stauffer@gmail.com

Abstract

This paper describes a method of presenting financial market data as an interactive, three-dimensional surface. When compared with commonly used market charting techniques, these new visualizations can provide a way to improve the speed and depth of human analysis and understanding. This overview is an introduction to the concepts of the proposed visualization method and is in no way intended to be exhaustive or complete.

Outline

Introduction

The financial markets generate a tremendous amount of data. When confronted with this seemingly incomprehensible wall of data, we have several options. One solution is to simply let computers do the heavy lifting, a solution to which quantitative and algorithmic trading funds subscribe. However, are computers aiding human understanding of the markets, or is our reliance on them simply creating a "black box" of analysis and execution? Is human oversight being further and further removed from the reality on the ground?

This paper does not present another set of computational analyses to be let loose on financial datasets, but instead focuses on the goal of improved *human* understanding, in order to allow more complete *computer-aided* analysis. To this end, it proposes a visualization technique that will engage our human familiarity with three-dimensional objects, which (in contrast to pure numbers) we intuitively understand due to their ubiquity around us.

Common Representations of Financial Data

The conventional methods for looking at market data confine us to a two-dimensional view, which we can quickly illustrate by looking at several representative charts.[1]

We start with a view of the S&P 500 index from 2010 to 2012.

Figure 1. S&P 500 Index, 4 Jan 2010 to 31 Dec 2012 (Source: Yahoo! Finance)
Unlabeled trading volume data is displayed in bar format below the line chart

This two-dimensional view gives us a great view of the price motion of the index. The index, a formula-based weighted average of its component stocks, eliminates most of the idiosyncratic variation of individual stocks, allowing larger trends to show through.

We can see an example of this individual variation by looking at a chart of Apple, Inc. (AAPL) over the same period.[2]

Figure 2. Apple, Inc. (AAPL), 4 Jan 2010 to 31 Dec 2012 (Source: Yahoo! Finance)

This AAPL chart clearly allows us to see more individual details, but we have now lost our view of what the broader market was doing over the same period. Was the rise and decline of AAPL's stock price in early 2012 part of a broader market movement, or was it specific to AAPL? We can lay the charts next to each other and compare them, but this process is time consuming and inefficient. In addition, it is difficult to find common reference points in two side-by-side charts, increasing the likelihood of error.

A better solution is to overlay the S&P 500 chart on top of the AAPL chart. For such an overlay, it is typical to focus on relative rather than absolute price movements by indexing both stocks to a shared starting value. The chart below indexes AAPL and the S&P 500 to a starting value of 0%, thereby tracking total return since 4 Jan 2010.

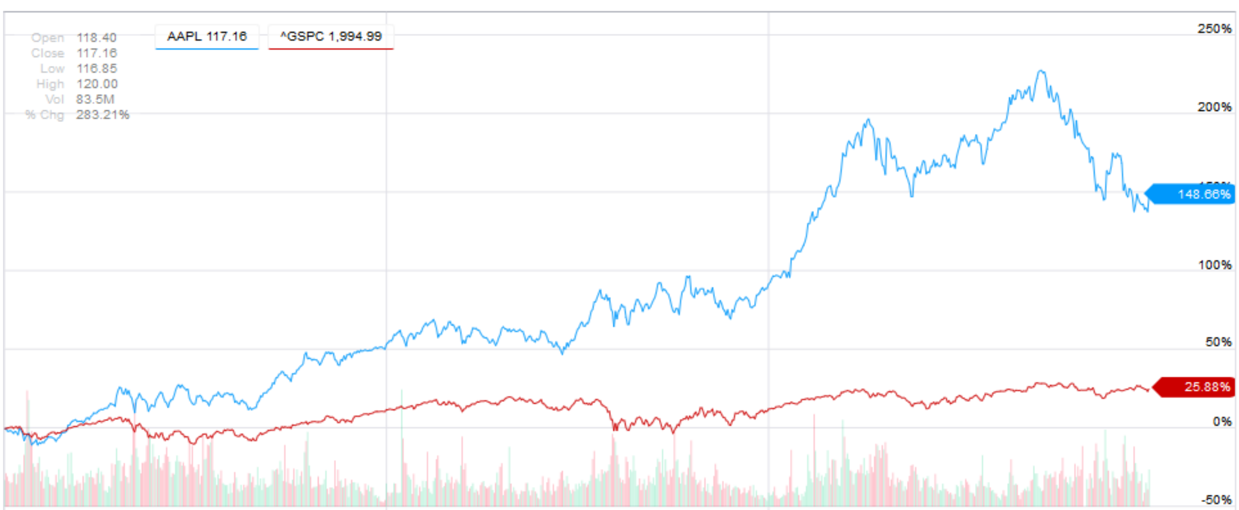

Figure 3. Comparative chart, AAPL and S&P 500, 4 Jan 2010 to 31 Dec 2012 (Source: Yahoo! Finance)

At first glance, an advantage of indexing is that it we can easily expand the data set to include multiple stocks.

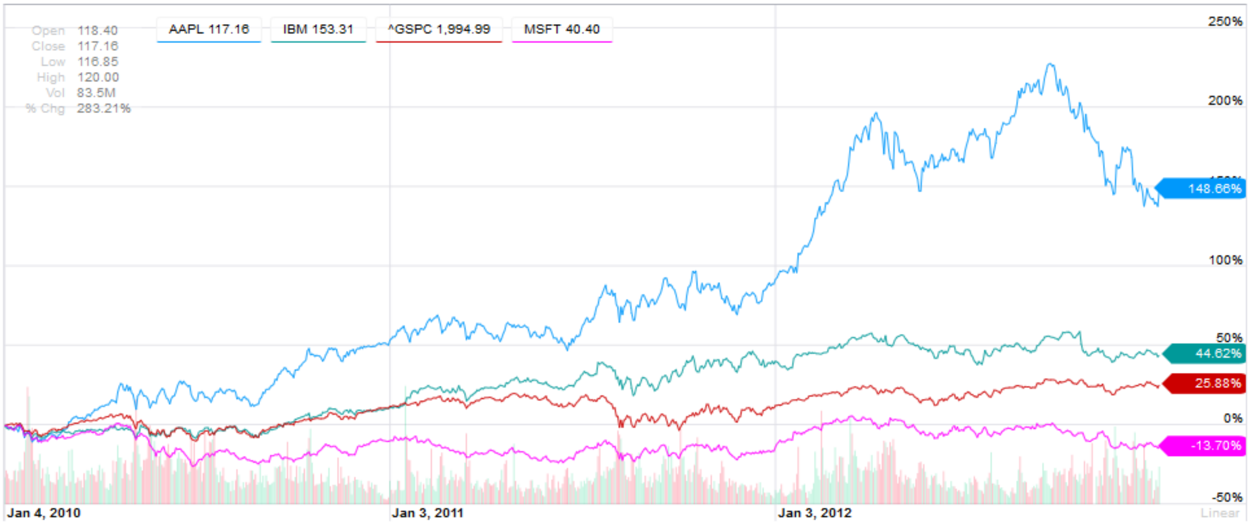

Figure 4. Comparative chart, AAPL, IBM, MSFT and S&P 500, 4 Jan 2010 to 31 Dec 2012 (Source: Yahoo! Finance)

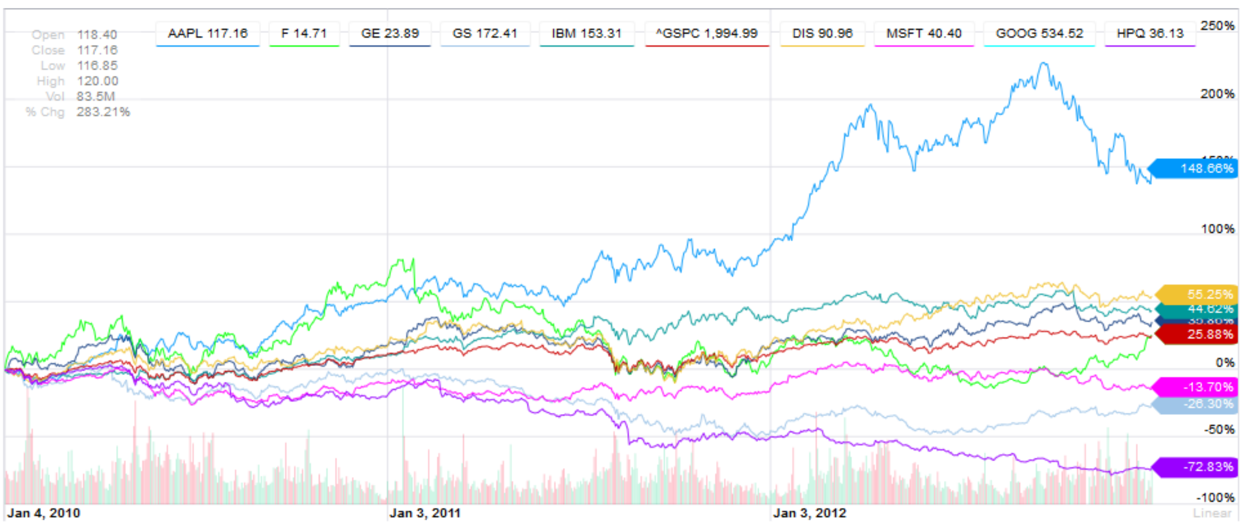

Figure 5. Comparative chart, 10 stocks, 4 Jan 2010 to 31 Dec 2012 (Source: Yahoo! Finance)

However, even when we show just 10 different stocks – only 2% of S&P 500 components – we encounter severe limitations in readability and understanding (as well as exhausting our color palette). While the “index and overlay” method can be useful in comparing returns of small groups of stocks or indices, we are still faced with an inability to view broad amounts of information both quickly and intelligibly.

A Proposed Alternative Using Three Dimensions

Instead of limiting ourselves to two-dimensional "flatland," we can use three dimensions to better represent our S&P 500 data.

We first define our three dimensions. We will utilize Time (*t*) as our x-variable and the individual Company, *C* as our y-variable. For example, C_1 might be Apple (AAPL) and C_2 might be Ford (F). We define Price (*P*) as our z-variable. Since we hope to view many stocks simultaneously, we will define price using an "indexed price" model. We will set our initial indexed price equal to 100 and use a formula to calculate prices thereafter.

$$r_{0,1} \equiv \frac{P_1 - P_0}{P_0};\ I_1 = I_0\left(1 + r_{0,1}\right) = I_0 \frac{P_1}{P_0}$$

where I_n is the indexed price at time t=n, P_n is the actual observed price of the stock at t=n, and $r_{n,\,n+1}$ is the return on the stock during the period from time t=n to time t=n+1.

Using a three-dimensional axis, we plot the three variables as a *surface*.

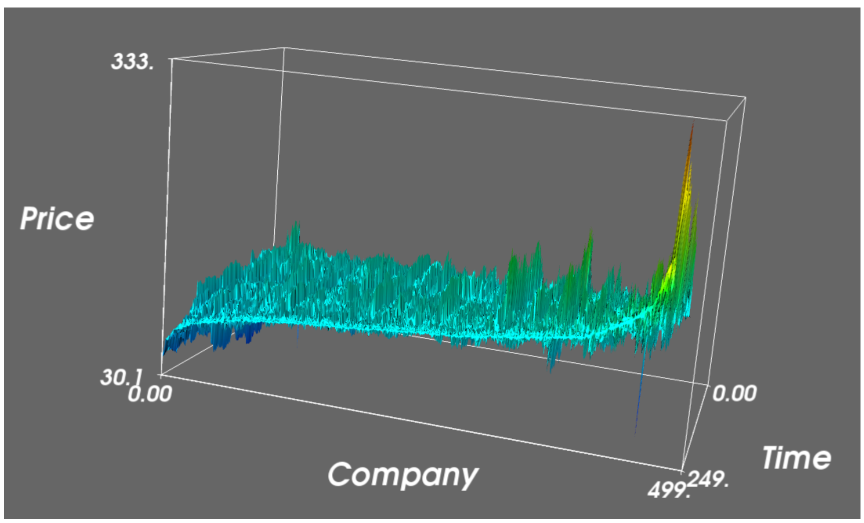

Figure 6. Initial prototype diagram of S&P 500 stocks, visualized using Mayavi2

Left stationary, this projection has two key issues. First, it is clear that a given view is likely to obscure certain points in the dataset. For example, a "hill" might obscure a "valley." Secondly, a simple two-dimensional representation of our three-dimensional surface offers little improvement over our previous charts.

Therefore, the key to this visualization's success is its user interactivity. By allowing the user to "physically" interact (at least in a virtual sense) with our surface, we provide a significant boost to understanding.

In order to interact with the 3-D surface, we should allow for the following primary interactive functions:

1. Rotation and Zoom: This is the key to simulating an actual physical object with which the user can interact. The functionality must be easy and intuitive, and the movement of the surface must react in real-time to user manipulations. By rotating the surface, the user can view features that may be hidden from a single vantage point. Zooming allows the user to focus in on specific details. In theory, the zoom will allow greater levels of detail to be seen, such as higher frequency pricing data (by the hour, minute, second, and beyond). This is a feature not currently available in standard two-dimensional charts, for

which the data frequency is either manually specified by the user, or automatically set by the program based on the time period displayed. For our 3-D surface, the combination of both zoom and rotation functionality provides the user with a fully *investigative* interactivity.

2. Extend/Shorten axes: The user will be able to view additional data in the visualization by means of extending the axes. A key consideration for the extension of axes is that the processing time for the visualization will increase significantly as more points are plotted. This may also increase the lag time experienced during manipulation of the surface. The specific functions along each axis are as follows:

- The x-axis (Time) can be extended/shortened in order to show a larger/shorter time period of data. The size of the axis affects the number of dates to plot.
- The y-axis (Companies) can be extended/shortened to show more/fewer companies. For example, the user may begin the visualization with the stocks of the S&P 500, but later decide to include Russell 2000 stocks. These additional companies will be added to the y-axis, increasing the number of companies for which pricing data is plotted.
- The z-axis (Price) needs no user adjustment because it will automatically scale to include the necessary price levels in the scope of the current dataset. This scaling can take place by setting the axis algorithmically prior to plotting.

3. Selection of single data points and cross-sections: Point data will be primarily used to audit data quality and search for outliers, as well as providing an anchor for notating specific global or local maxima and minima on the surface. Cross-sections provide a two-dimensional view with additional data. Points and cross-sections must be easily navigable via both mouse and keyboard. A breakdown of the cross-sections is as follows:

- x-z (Time-Price) – Selecting this cross-section provides a standard price chart for the selected stock. It can also include return and volatility metrics, a subplot of daily returns, Beta calculations for the stock, and company identifying information.
- y-z (Company-Price) – Technically this would yield the relative price levels of all companies at a specific point in time. However, the usefulness of these relative price levels suffers from the arbitrary indexing method (in which all prices began at 100). On the other hand, variations of *returns* for a specific period can be very meaningful. For example, in the period from t=0 to t=1, how do the returns of all companies compare? Which companies realized returns that differed significantly from the median return of that time period?
- x-y (Time-Company) – This could potentially be displayed as a two-dimensional diagram of the results of a manifold learning algorithm such as t-SNE.

Computer-Enhanced Organization of the Y-axis

Maximum usefulness of the visualization will only be realized through the proper and logical organization of the y-axis. Companies should ideally by arranged to minimize price variation with their neighbors, or else the surface may show too much "roughness" overall. This would limit our perception of any but the largest trends and patterns.

Companies should therefore be arranged so that there is high *covariance* among y-axis neighbors. This high covariance will likely arise among industry and sector groupings. Any idiosyncrasies that diverge from these sector groupings will be of notable interest to the trader or analyst.

For the actual organization on the axis, manifold learning algorithms may be very useful. For example, the t-SNE algorithm[3] developed by Laurens van der Maaten and Geoffrey Hinton may have an application here.

It is important to note that the covariance relationships among companies will change over time. However, for an intelligible surface visualization we can only view one y-axis organization for a given time period. There are two options to deal with this limitation. The first option is to use the "organization algorithm" every time a new "view" occurs (i.e., companies are added, or the viewed time period changes). This may ensure the most optimal neighborly relationships, but it will also significantly increase processing time. In addition, the constantly shifting company alignments will significantly decrease interactivity and user comprehension. Our minds require constants in order to perform successful pattern recognition; shifting this order will deny our minds the possibility of forming these constants.

A better solution is to use a more robust – and likely slower – algorithm to seek out a "good-enough" y-axis organization over a wide variety of time-periods. Perfection is not necessary, and since aesthetics may also play a part, it may be beneficial to use human developer-aided manipulation for the final ordering. This same ordering will then be held constant – and any changes will be noted to the user (in the style of a "version update").

A final note on is that many manifold learning algorithms are generally utilized to project high-dimensional data onto *two* dimensions. In this case, the goal is project information onto a single dimension, the y-axis. It is unclear at this time what the complexity of this task will be.

Uses of the Proposed Visualization

The proposed visualization could offer significant improvement to our ability to understand the vast amount of market data that is created on a daily basis. The above solution focused on only a small subset of the financial markets. The equity markets provide a prime testing ground for such a visualization because the boundaries of the data are relatively confined. In our S&P 500 example, we have only to deal with data for just over 500 stocks (505 in March 2016), and this data is (relatively) easy to find and manipulate. S&P 500 components are only rarely delisted or changed, and there is readily available public information about such changes. Equity options, on the other hand, are constantly expiring and being replaced. If we were to look at options on the S&P 500 index – the index only, not the component stocks – we would be facing over 50,000 individual listed options in only a 5 year period. While the visualization described above could eventually provide significant understanding of the complexities of options markets, this is a task best tackled after an initial visualization model is developed and proven.

With an equity market visualization, our understanding of the impact of different events and individual stock's responses to larger trends will be enhanced. For example, during periods of large volatility, do some stocks react more violently than others? Do certain companies repeatedly react first to market shocks? How successful are industry classifications for grouping stocks? We can try to answer these questions now, but such answers generally rely on models and statistical analyses. In the words of Benoit Mandelbrot, "the conclusions yielded by [statistical tests] evaluate both the model and test in some inextricable combination, from which little of use can be inferred."[4]

When dealing with large datasets, we limit ourselves by relying solely on pure numbers – numbers with which computers are certainly better at dealing. However, by turning these numbers into an interactive object – including a virtual one such as a visualized surface – we are able to put more of our *human* senses to use. By *seeing* and *moving* the data, we can hopefully gain a more intuitive understanding, and then allow this intuition to better inform our models and tests. In this way, we can hopefully begin true computer-*aided* analysis. Most importantly, as long as humans are still the ones shaping economic and financial policy through laws and actions, human understanding of the data carries significantly more value than "black box" computer solutions.

---

[1] The representative two-dimensional charts are sourced from Yahoo! Finance for three reasons. First, the charts are aesthetically appealing and simple to navigate. Second, Yahoo! Finance offers an accessible API for pulling stock price information, providing easily implementable, language-agnostic access to data. This leads to the third reason, which is that Yahoo! Finance data offers the best combination of quality and ease of access currently available on the internet.

[2] It should be noted that all charts are based on the *adjusted* closing price of the stocks involved. The adjustment takes into account dividends and stock splits, in order to keep consistency of price movements (and calculated returns) across all historical periods.

[3] Laurens van der Maaten provides descriptions and links on his website, http://lvdmaaten.github.io/tsne/

[4] Mandelbrot, Benoit (1997), "Three Fractal Models in Finance: Discontinuity, Concentration, Risk" in *Economic Notes*, Banca Monte dei Paschi di Siena SpA, vol. 26, no. 2-1997, p.171-212