

Stock Price Analysis Visualization Tool

Josiah Buxton*

Christopher Godley†

Brian Lubars‡

Kenneth Hunter Wapman§

Computer Science Department
University of Colorado Boulder

ABSTRACT

The goal of this final project was to create a stock price analysis visualization tool that would aid individuals in making informed decisions about the stocks they buy and sell. We created a webpage that informs a novice reader about stock price technical indicators in a narrative format. It also allows the user to interact with two different visualizations. One visualization introduces the technical indicators and demonstrates their relationships between each other. The second visualization gives the user complete control in generating price and technical indicator graphs of a stock price dataset.

1 INTRODUCTION

Stock trading can be very risky business when one is uneducated about the stocks he/she is buying. Technical indicators have been made in the past to give traders an indication as to when a stock/how much will rise or fall. Visualization tools have also been made, but many of them are inflexible and only contain a few technical analysis functions. The lack of a robust and extensive stock price data visualization tool is the motivation for this project.

Our goal was to provide a user interface that would allow a user to interact with a large set of stock price data and overlay different technical indicators on top of time series data. We realized that if we were to employ all of these technical indicators, there would be a good chance that amateur and novice traders/analysts would be unfamiliar with the characteristics of these indicators. In order to address this, we decided to expand the scope of our project to incorporate a learning experience where the user would scroll through a short lesson on each of the technical indicators before accessing our visualization tool.

2 RELATED WORK

Stock data visualizations are a critical part of financial trading. Without interpretable visualizations, a trader may miss a new trend as it forms when it would have been obvious with the right tools. Yahoo is one of many sites out there that provide convenient and powerful stock tools. Without the need of an account, anyone can go to finance.yahoo.com [4] and view relevant trading information about any stock. This was the leading inspiration behind our visualization, as well as many others as we discovered in our research. We wanted to duplicate the basic functionality of that site with our own spin and organizational requirements. The foundation of this was the ability to display any stocks value over time in a consistent view. These views would be supplemented by the other functions we implemented to analyze the trends. There are many resources openly available online that discuss methods of how to best analyze stock data. One of these that was extremely useful was the Technical

Analysis from the University of Cambridge [2]. This is a 183 page analysis of various stock indicators and trends. In this document, they talk extensively about the various technical aspects of stock modeling, but they also show and discuss many ways to visualize this data. In many ways, this was a great technical document to dive into, but most of these methods simply generate various two-dimensional plots. Some interesting multi-dimensional work has been done by Joseph and Indratmo in their paper Visualizing Stock Market Data with Self-Organizing Map [5]. Instead of the traditional two dimensional scatterplots, they used an unsupervised learning algorithm to plot and group stocks onto a three dimensional grid, where stocks are organized two dimensionally with the third dimension is proportional to the company's market capitalization. The paper claims serious benefits from clustering stocks together and treating stock clusters as individual nodes when making trading decisions. Its an intriguing look into the use of more than two dimensions for stock visualization, but ultimately they compared their results to the Yahoo Finance as well, as it is truly such a robust tool. Due to this, we will try not to include multi-dimensional visualizations in our work here. From a users perspective, there may be reason to view stocks at an individual level or at a more global level, however, where the general trend of the stock market is just as important. This is what the self-organizing map from Joseph and Indratmo is aiming to do, but the paper from Simunic [6] shows that similar results can be found by simply using a heatmap and two dimensional space to reflect the trends across all stocks plotted in that space. Each plot represents a grid in the overall visualization, with zooming functionality for broad market views, or more specific stock trends. This was a much more favorable style of visualization than the Joseph paper, but again they compared their results to the functionality provided by the Yahoo Finance site, reiterating just how powerful and pervasive that format is. Certain aspects from this visualization would be great to incorporate, however, as displaying more than one stock at a time is a common need.

Even after more extensive research, we were left surprised with how good the Yahoo model actually is, and therefore were not ashamed to emulate it as much as necessary to get the results we desired.

3 DESIGN PROCESS

3.1 Data Wrangling

We used the AlphaVantage API [1] in order to obtain real-time stock data. We made queries to capture data for six different stocks from the time of the stocks inception. We captured the stocks price data through a time series daily adjusted module which returns the open, close, high, low, volume, and adjusted close prices of an array of dates. We then used a variety of other modules to capture SMA, EMA, STOCH, RSI, MACD technical indicators. We decided upon these indicators after researching commonly used analysis tools on investopedia.com [3]. All of this data was stored into a python dictionary and then outputted into json files. The json files are read in with javascript and displayed using D3. We chose to work with D3 because we were very comfortable with making two dimensional plots out of loaded json data and as a programmer, you have a lot of

*e-mail: josiah.buxton@colorado.edu

†e-mail: christopher.godley@colorado.edu

‡e-mail: brian.lubars@colorado.edu

§e-mail: kenneth.wapman@colorado.edu

control over particular aspects of the data.

3.2 Visualization 1: Narrative Introduction

We wanted to leverage what we learned about data narratives/storytelling in Week 15 here, to take something which is normally dry and help draw the subject in. We drew on ideas from [7] to first set the stage, introduce the characters, and tell a memorable story. The stage is the world of stocks and public companies, the characters are the data and the technical indicators, and the story we're trying to tell is that these indicators have questionable predictive power – as the saying goes, "Past performance is no guarantee of future results!" To lay out the narrative, we did a brief sketch of a storyboard, but this is a simple narrative so the storyboard was overkill. At the end, we wanted to also give the reader a chance to play with what they learned in a free-form format, so Visualization 2 is a flexible exploration tool that doesn't strictly fit in the narrative format.

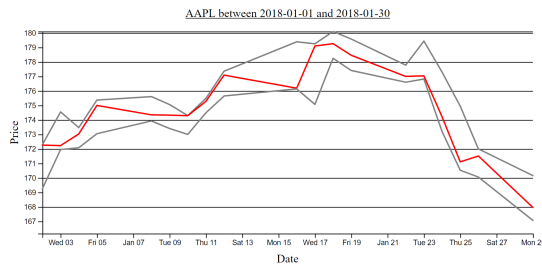


Figure 1: This depiction is the first visualization that is shown within our narrative introduction process.

3.3 Visualization 2: Analysis Tool

After our research, we knew that the most practical visualization tools used two dimensional scatter/line plots in order to plot time series data. We decided to use both for all of the different data types we gathered. We wanted to give the user the ability to drill down to any type of technical indicator or price and plot a range of data from the dataset. Selecting the technical indicator is done by selecting one of the five buttons displayed immediately above the graph. To the right of the graph, option boxes are shown and allow the user to select different stocks, price time series, time range, and trendlines.

When one of the five technical indicator buttons are selected, a series is either drawn on top of the price or a new graph is generated based on the scale of the technical indicator. This was done in hopes that the user would be able to more easily discern the information. When any of the options are selected in the option boxes to the right of the visualization, a new time series of prices is loaded from a json file and drawn over the visualization space.

With the basic stock plotting working, we wanted to add a supplement that would be useful. The trendline drop down menu offers 'none', 'linear', or 'curve-fit' views. The selection is caught by triggering an on-change event to update an internal variable in the javascript. A function is called by the change event that also calls the `make_plot()` function so that the appropriate trend line can be drawn. For the linear fit, an intermediary function was required to generate the linear points. The mean of the originally view-filtered data is calculated and then a \hat{y} value is calculated based on the overall trend. This is the same function called for the curve fit, but when the filtered data is passed through to the same function a fitted curve is maintained. A `lineFunction` was defined to use the `d3.curveLinear()` to plot the line. To populate the drop down menu, another function was written to grab the appropriate values from the `data/trendlines.json` file. More trendlines may be added here, but

these are the two implemented. The next trendline to implement may have been a rate-of-change curve, or even a trend prediction, but these would require additional mathematic and statistical functions which would take much more time to implement.

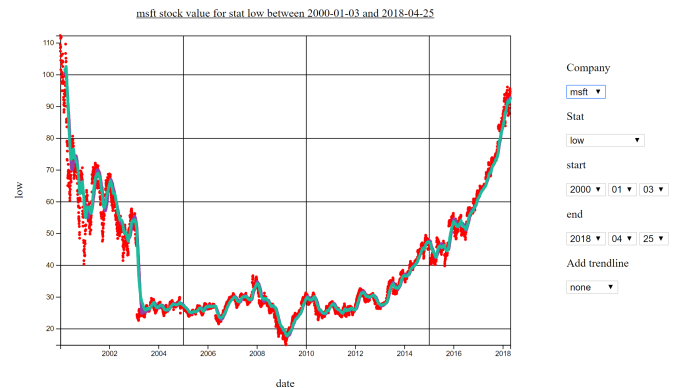


Figure 2: This depiction shows the default screen for the visualization tool that we developed. On the right hand side, the option boxes are available for the user to select a company, drilled down statistic, time range, and trendline.

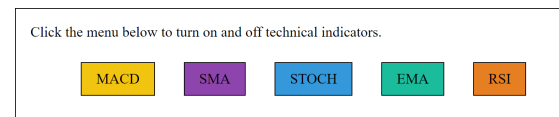


Figure 3: This depiction shows the menu for the different technical indicators that we have available for each of our 6 stocks. Clicking on the button draws the technical indicator either on the stock price graph or on a separate graph depending on the scale.

4 DISCUSSION

REFERENCES

- [1] Alpha vantage api. <https://www.alphavantage.co/documentation/>. Accessed: 2018-03-10.
- [2] Cambridge technical analysis of stock prices. https://www.mrao.cam.ac.uk/~mph/Technical_Analysis.pdf. Accessed: 2018-05-03.
- [3] Investopedia. <https://www.investopedia.com>. Accessed: 2018-03-10.
- [4] Yahoo finance visualization tool. <https://finance.yahoo.com/quote/%5EGSPC?p=%5EGSPC>. Accessed: 2018-04-28.
- [5] K. imuni. Visualization of stock market charts. In *Proceedings from the 11th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision 2003 (2003), Plzen-Bory (CZ), 2003, 2003*.
- [6] J. Joseph and I. Indratmo. Visualizing stock market data with self-organizing map. pp. 488–491, 01 2013.
- [7] K. L. Ma, I. Liao, J. Frazier, H. Hauser, and H. N. Kostis. Scientific storytelling using visualization. *IEEE Computer Graphics and Applications*, 32(1):12–19, Jan 2012. doi: 10.1109/MCG.2012.24

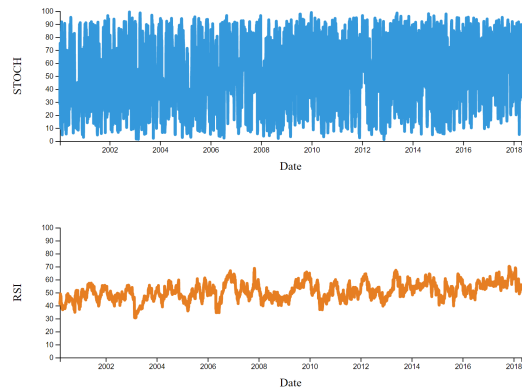


Figure 4: This depiction shows examples of two of the different technical indicator graphs that are drawn below the stock price graph if their respective option is selected in the menu.