Linear Algebra Analysis Techniques Applied to the Stock Market

As long as the stock market has been around, people have been trying to profit off of it. With the advent of computers and more advanced mathematical/statistical techniques, the opportunity and desire to use them in the financial markets has grown. In this paper, I will try to use knowledge gained from the class Matrix Analysis for Data Science (CAAM 334) to implement and learn new techniques for data analysis in relation to the stock market. I understand that getting novel information that is useful in real-life situations is very improbable, so if the techniques are useful in verifying common-knowledge ideas about how markets work, this project would be successful.

I tried three different techniques which include: Principal Component Analysis to determine the framework of the Dow Jones Industrial Average, Linear Least Squares to investigate the importance of different financial information in determining the price of a stock, and Fast Fourier Transforms to determine if the financial markets (and individual stocks) operate with a certain predictable frequency.

For Principal Component Analysis, I had the hypothesis that by looking at all the stocks in the Dow Jones Industrial Average and doing an analysis of their prices throughout time, we would be able to find a differentiable factor that would inform which stocks are better to add to an investment portfolio.

For Linear Least Squares, I had the hypothesis that when looking at the previous close price, the average daily volume traded in the last ten days, the market cap, the trailing price to earning, and the shares outstanding, we would be able to find this through the minimization of x in $|Ax-b|_2$. The matrix A is the data for each stock, b is the prices that were recorded the next day, and x is the coefficient vector that would give knowledge into how important the different financials were. I expected that the process would result in a coefficient for the previous close price to be around 1, however, I was not sure what the coefficients would be for the other information.

For an FFT analysis of the Dow Jones Industrial Average, I had a hypothesis that we would be able to see boom and bust cycles on display in the long term, but also see correlation between the most dominant frequency and the actual stock price over shorter periods of time.

Principal Component Analysis for the Dow Jones Industrial Average

Principal Component Analysis (PCA) stems from Singular Value Decomposition (SVD) which is a linear algebra technique that utilizes eigenvalues and vectors to determine the most important information in recreating a certain matrix. SVD decomposition is useful in image compression and other fields where saving memory is important. I thought that SVD may be useful in the finance setting because completely recreating stock data would not be useful as it would not be as applicable on new information, but getting the most important values (or principal components) may be useful as a generalization. Furthermore, in PCA because all of the principal components are orthogonal to each other, adding more principal components only increases precision in recreating the original matrix. However, there tends to be a drop off in effectiveness.

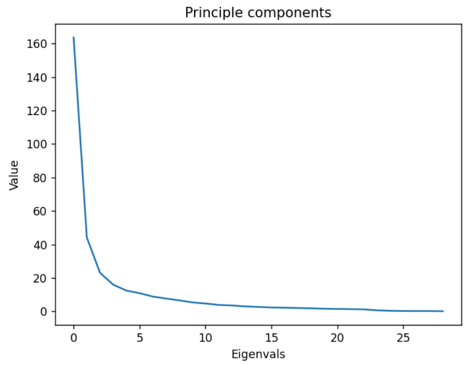
I used all the stocks in the Dow Jones Industrial Average as I believe they will get a good general grasp of the market. One thing to note is that I gave equal weight to each stock, so the graphs (and total price) will be different than the actual DJI price. Here are their tickers:

```
['aapl', 'msft', 'unh', 'v', 'jnj', 'wmt', 'jpm',
                 'pg', 'hd', 'cvx', 'ko', 'dis', 'csco', 'vz',
                 'nke', 'mrk', 'intc', 'crm', 'mcd', 'axp',
                 'amgn', 'hon', 'cat', 'ibm', 'gs', 'ba', 'mmm',
                 'trv', 'wba']
```

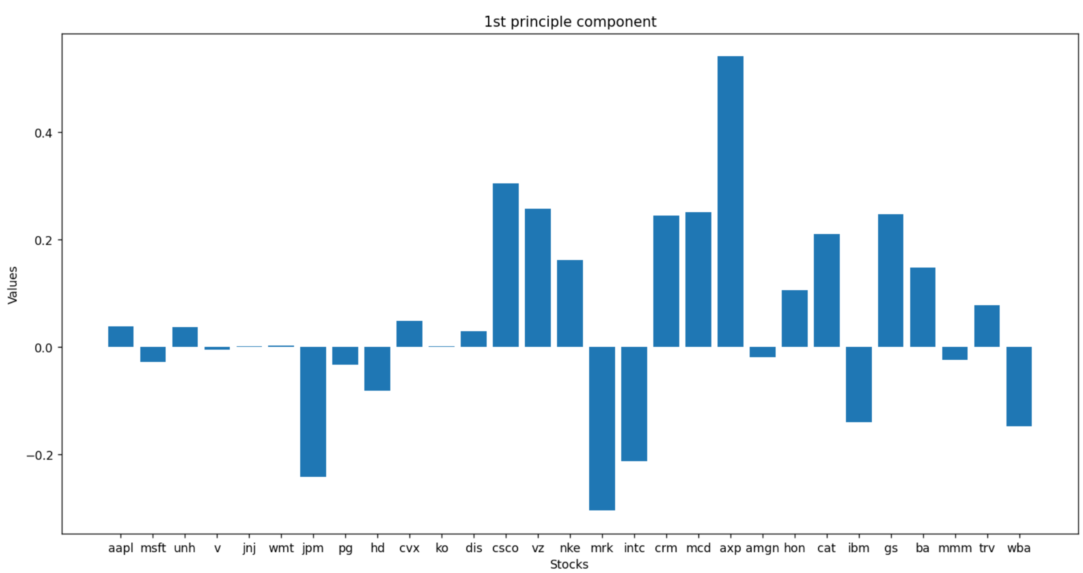
The first thing I did was to translate the stock data into change-per-day data. This standardizes the way we analyze the stocks without having to worry about their individual prices. It also makes it so the data is all relatively near the x axis (hence a certain amount of linearity). The next was to find the covariance matrix. The covariance matrix tells us the correlation between two stocks. If the values are positive, this means that the two stocks increase or decrease together. Notice on the covariance matrix below, all the values are positive. This may mean that in general, stocks in a common market will increase or decrease together, based on the economic health of the market and its participants.

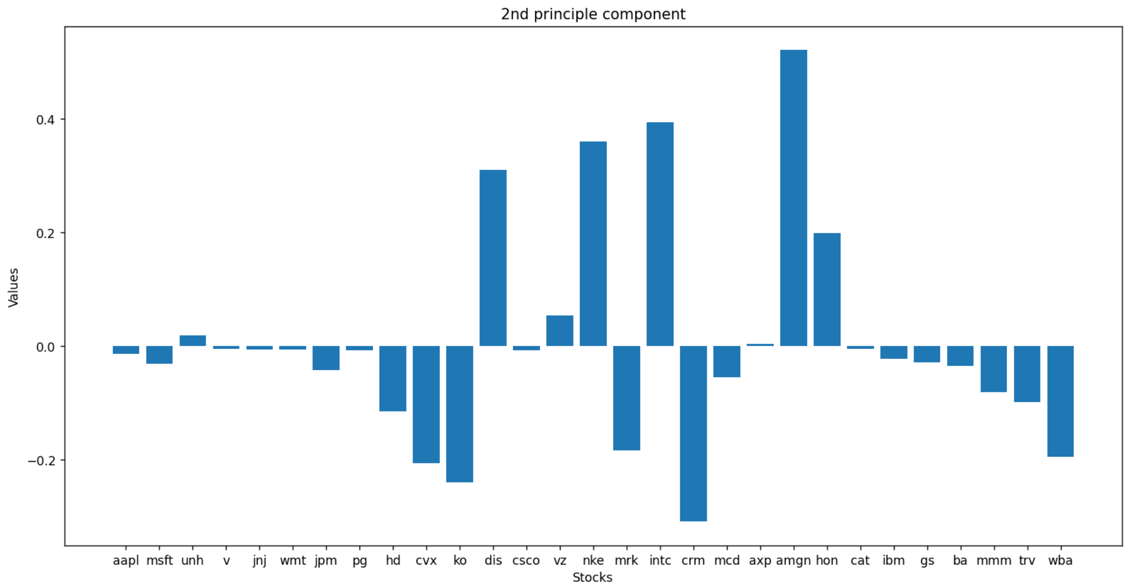
	aapl	msft	unh	...	mmm	trv	wba
aapl	10.298482	13.913352	10.557501	...	3.831803	3.008883	1.282115
msft	13.913352	30.109853	16.685475	...	5.802442	4.158547	1.984834
unh	10.557501	16.685475	59.758412	...	4.386550	7.572744	2.022593
v	7.748593	11.814217	11.706712	...	3.953881	3.165437	1.198112
jnj	2.301479	3.592807	6.853399	...	1.449227	1.643958	0.511478
wmt	2.593768	4.254760	4.947696	...	1.293983	1.165564	0.641781
jpm	3.659140	5.453348	6.502091	...	2.512665	3.015388	0.777094
pg	2.873686	4.639615	6.720958	...	1.687710	1.586464	0.624571
hd	11.073050	19.361918	13.467163	...	7.398559	3.923203	2.196848
cvx	4.268933	6.314828	8.128707	...	1.836195	3.530254	0.870937
ko	1.260887	1.989786	3.000208	...	0.756272	0.797894	0.282888
dis	4.845305	8.035829	4.864299	...	2.548551	2.080981	0.913718
csco	1.671002	2.461077	2.498633	...	0.830027	0.804344	0.251480
vz	0.681758	0.963941	1.893085	...	0.427510	0.416529	0.153610
nke	5.957020	9.433509	5.772559	...	3.285494	2.292909	0.958173
mrk	0.907495	1.269220	5.345000	...	0.628503	1.263305	0.200148
intc	1.756557	2.919537	2.051987	...	0.929944	0.651803	0.289751
crm	9.511780	17.343988	10.232639	...	4.122738	2.752234	1.296916
mcd	4.807338	7.300191	11.232224	...	3.062980	3.469500	1.028102
axp	7.050018	10.191896	8.930722	...	3.629374	3.646741	1.157020
amgn	2.997715	4.505773	9.609635	...	2.781352	3.025548	0.863660
hon	5.412909	9.004035	9.373716	...	4.311257	4.092050	1.043594
cat	6.521991	9.604752	9.620740	...	4.855105	5.524300	1.445084

The next step is to find the eigenvectors and eigenvalues in the covariance matrix. All of the eigenvalues of the covariance matrix will be positive and their magnitude implies their importance. Thus, in order to find the most important eigenvector, I found the largest eigenvalue. I understand that it takes more than one principal component to make up a matrix, so I tried to determine a good place to cut off the principal components. I wrote code that cut off adding any more eigenvectors if there was less than a ten percent difference in magnitude between the eigenvalue and its predecessor. Here are eigenvalues that I kept: [163.5872782918699, 44.18961165843452, 23.114461887730663, 15.998701868905936, 12.464189983056439, 10.9016305198936, 8.893429809734107, 7.713094800746904, 6.616410112700846, 5.3907684172903645, 4.749585970709282]. There are 9 eigenvalues that were kept. If one is to look at the graph, this corresponds well to an "eye" test of minimal returns.

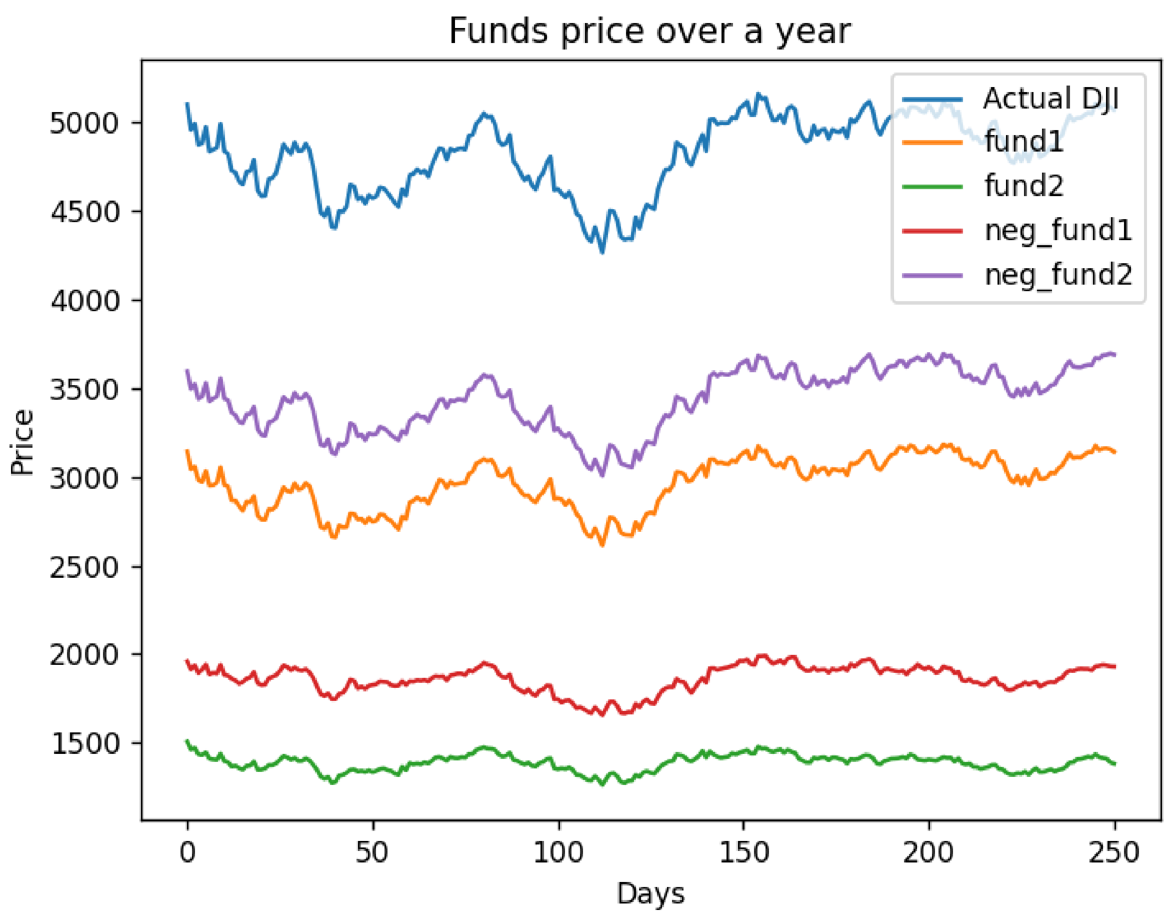


For the purpose of this paper, I took the first and second largest eigenvectors and looked at their associated eigenvectors. I wanted to then create portfolios for all the stocks that have positive values and all the stocks that have negative values for each eigenvector. The reasoning behind this was something similar to that of the fiedler value and vector in graph theory, and the idea that perhaps that there is some sort of differentiation (associated with positive or negative values) that then results in some increased or decreased performance.





Thus, portfolios were created with such results:

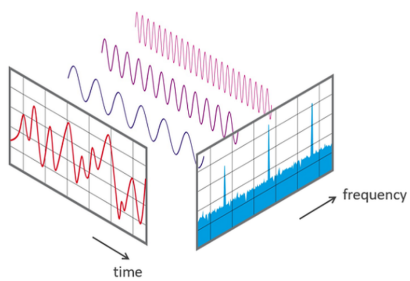


*Note that the prefix "neg" just means all of the stocks that were not included in the other fund.

It is interesting to note that neg_fund1 and fund2 are more stable. Perhaps the weight vectors comment on stability of stocks more so than performance. However, the main takeaway is that all portfolios follow the same general pattern. What this may imply is that stocks move within the context of the market. So no matter how one groups the stocks, they all have some fluctuations at similar times. While “the market” is an abstract idea, we have shown that diversification (through modeling the market) can protect against rogue stocks (that may individually decrease in price), which for the average investor, is perfectly sufficient.

FFT analysis of the Dow Jones Industrial Average

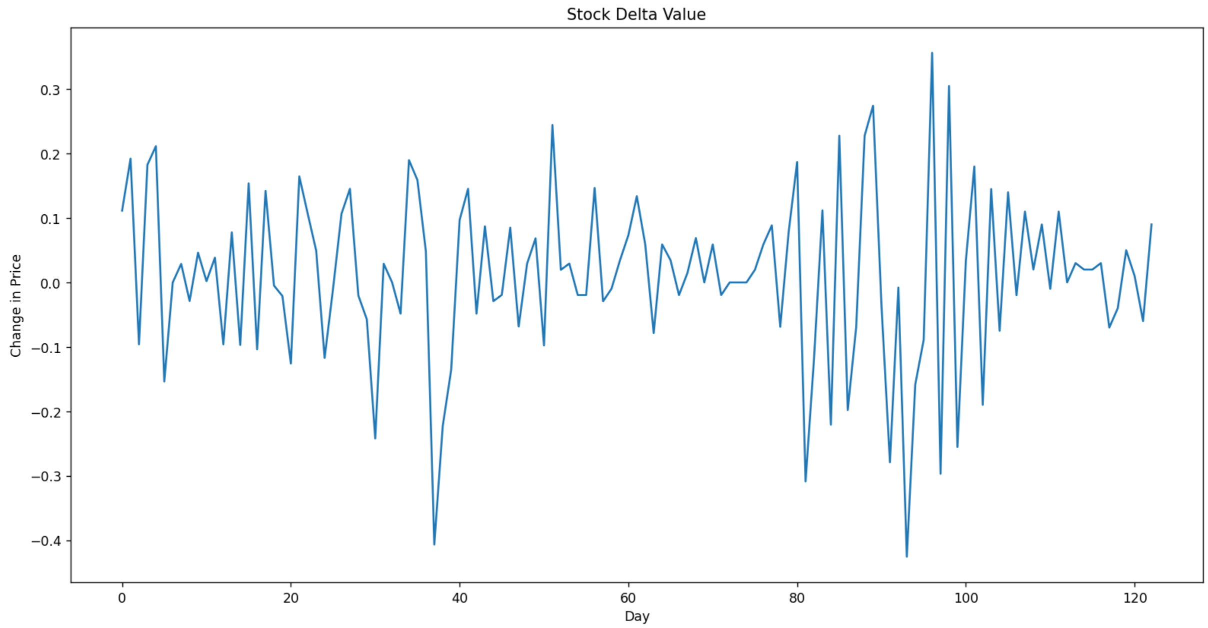
The Fast Fourier Transform (FFT) is a signal processing technique that stems from the Discrete Fourier Transform (DFT). The FFT translates discrete signals in the time domain (in our case, stock data) into the frequency domain with a more efficient run time as opposed to the FFT.



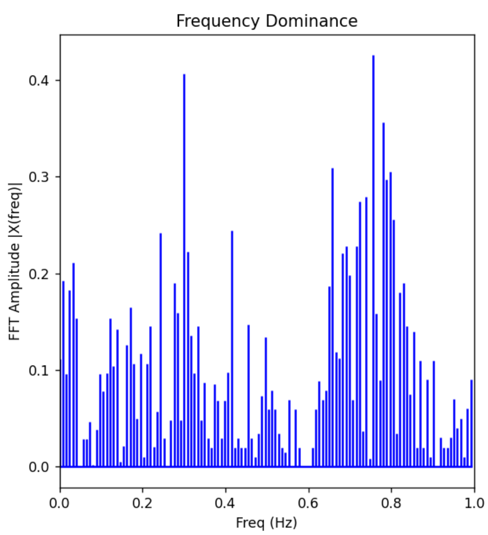
This is a change of basis.

Through this technique, any discrete signal can be decomposed into sin waves of different frequencies. In the linear algebra sense, these signals can be written in a matrix where each column is of a certain frequency k, and is modeled by e^(i2πk) (a Fourier Basis). We can use the characterization to project our signal with the Fourier Basis to get to the frequency domain. The matrix is orthonormal so finding the inverse, and therefore finding the original signal is also very easy

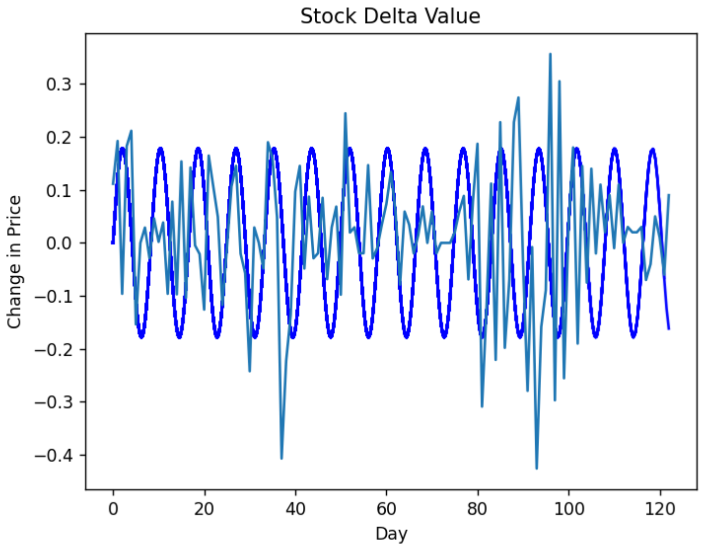
In our specific case, we are trying to take the frequencies that compose the stock data. So we can run an FFT on the data. This is what the original data looks like:

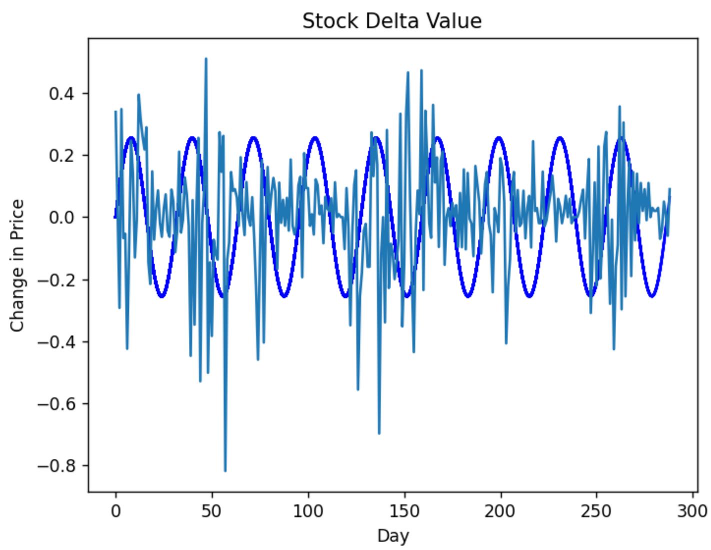


Then, given information about all of the frequencies that make up the signal from the FFT, we can choose the most dominant frequency that makes up the stock data.

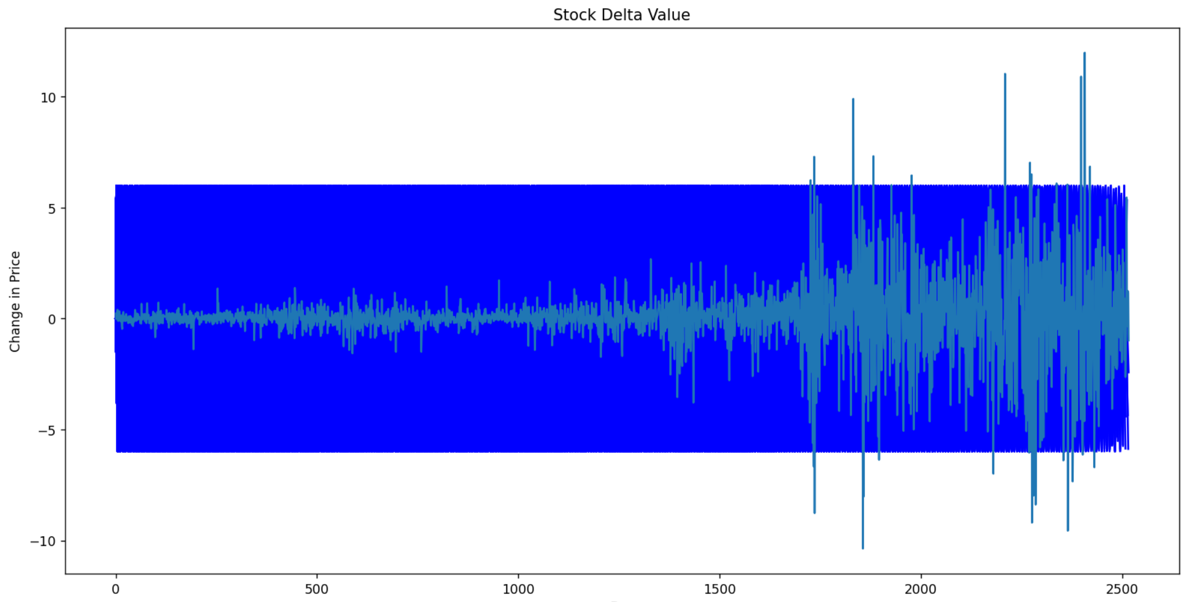


Finally, we choose the most dominant frequency and see how it compares with our original stock data.





One thing to note is that this works best on periods of 6 months to a couple of years. The problem with having anything smaller or larger is that many times, the stock price can have very large fluctuations that make getting an accurate dominant frequency much harder (or at least fit worse visually). For example, taking a stock like Apple over a period of ten years gives these results.



It is hard to take away anything monumental from this, but we do know, because of the math behind the process, that the frequency we found does correlate with the original stock data. The reason it seems to "blow up" at larger time intervals is because only one sample is taken per day. If one were able to get data points at every minute, these longer time intervals may be possible to model, but unfortunately, such information is not available to me. FFTs for sensors usually take many data points every second, but in our case, we actually have a very small data set, even if we go back decades.

In summary, we have found a way to model stocks through signals, but because of a lack of data, conclusions for larger periods of time are not possible.

Linear Least Squares to Determine Relative Importance of Financial Information

As explained previously, the Linear Least Squares Method is used to assign good approximate weightings for an equation of the form $\alpha_1 x_1+\alpha_2 x_2+...+\alpha_k x_k=b_1$. By having a large number of x and b values, in this case x being certain the previous close price, the average daily volume traded in the last ten days, the market cap, the trailing price to earning, and the shares outstanding for a certain stock, we can create a matrix, multiplying it by a coefficients vector to determine the current price. We know that minimizing x in $|Ax-b|_2$ is the same as solving $A^T Ax = A^T b$. So, by collecting the data for the Dow Jones Industrial Average (although I had to exclude stocks BA and WBA for lacking information), we can determine the importance of all the different factors. Through this calculation, these were the results:

The coefficient associated with previous close price was: 0.99967505;
Average volume traded in the past 10 days was: -1.25814378e-07;
Market cap was: -1.49062319e-12 ;
Trailing price to earnings was: 0.00117073:
Shares outstanding was: 6.07700021e-10

When trying our determined coefficients on some other stocks (TSLA, CROX, FOXA, LMT) these were the results:

For TSLA the predicted price was: 147.95795777 and the actual price is: 187.0399932861328
For CROX the predicted price was: 145.56209173 and the actual price is: 139.0399932861
For FOXA the predicted price was: 33.51476976 and the actual price is: 33.9900016784668
For LMT the predicted price was: 490.28868178 and the actual price is: 489.6400146484375

What we can observe from this is that the coefficients are not all that far off in terms of weightings. Also, some of the predictions are not all that far off. As expected, the previous close price was considered the most important factor in determining the next price. What was interesting to see was that the trailing price to earnings was the second most important factor. This also makes sense because price to earnings is used in the real world to determine the future price, and is a much more influential factor than market cap, shares outstanding, and average volume traded. Furthermore, the fact that average volume traded is more influential than market cap and shares outstanding makes sense because it is a much more time/situation dependent factor.

What the Linear Least Squares approximation method has done is verify the conventional wisdom that the previous price will influence the next price the most, but also that trailing price to earning and volume traded are somewhat important to the future price.

Conclusion

Through linear algebra techniques applied to the stock market, we were able to verify common knowledge about the stock market through linear least squares, determine that Fast Fourier Transforms can be applied to stocks to get a general sense of the fluctuations that they may go through as well as determine why, in our specific situation, they may not work as well for longer periods of time, and that PCA may result in determining the stability of certain funds as well as verifying the time-old wisdom that diversification is the best strategy.

There are also many ways to extend the techniques used and take advantage of the yahoo query (stock data scraping) module. One way that was already hinted at was gathering more data for each stock to analyze longer periods of time for the FFT. This can be done through using a more professional service to gather minute-by-minute information. It seems that yahoo query only allows a certain amount of queries per day when someone wants to get 1 minute intervals.

Furthermore, with more time, I would have been able to dive more into papers on [PCA] and on [FFTs] as related to the stock market. I would like to try the methods and techniques they used, as well as see how their results compare to mine. For the Linear Least Squares approximation, I would like to compile a list of more stocks and select more financial information for each in order to compare the weights to see what information is most important. Yahoo query is able to get data on news ratings, and so figuring out how to incorporate this into the calculation could result in more accurate and timely results. I could also use ChatGPT to analyze the headings of news articles to get more of a sense of what opinions on the stock are.

I had an absolute blast with this project and would love to continue expanding on it in the future as I learn more math skills. Thank you for a great semester and have a great summer!

-Gregory Chekler

Sources:

Cody, Devin. “An Intuitive Interpretation of the Fourier Transform (or the Link between Fourier Analysis and Linear Algebra) · Devin Cody.” *Devincody.github.io*, devincody.github.io/Blog/post/an_intuitive_interpretation_of_the_fourier_transform/.

“Intuitive Understanding of Euler’s Formula – BetterExplained.” *Betterexplained.com*, 2019, betterexplained.com/articles/intuitive-understanding-of-eulers-formula/.

Tng, Daryl. ““Quantamental” Approach to Stock Trading Using the Fourier Analysis.” *Intuition*, 28 Aug. 2021, medium.com/intuition/quantamental-approach-to-stock-trading-using-the-fourier-analysis-58f64792290.

Nazrul, Syed (Sadat). “PyRevolution.” *GitHub*, 21 Apr. 2023, github.com/snazrul1/PyRevolution/blob/master/Puzzles/DSP_For_Stock_Prices.ipynb.

Guthrie, Doug. “Yahooquery.” *Yahooquery.dpguthrie.com*, yahooquery.dpguthrie.com/.

Rome, Scott. “Eigen-Vesting I. Linear Algebra Can Help You Choose Your Stock Portfolio.” *Scott Rome*, 18 Mar. 2016, srome.github.io/Eigenvesting-I-Linear-Algebra-Can-Help-You-Choose-Your-Stock-Portfolio/.

Jaadi, Zakaria. "A Step by Step Explanation of Principal Component Analysis." *Built In*, 4 Sept. 2019, builtin.com/data-science/step-step-explanation-principal-component-analysis.

Nikolai Janakiev. "Understanding the Covariance Matrix." *DataScience+*, 3 Aug. 2018, datascienceplus.com/understanding-the-covariance-matrix/.

"Interpret the Key Results for Principal Components Analysis." *Support.minitab.com*, support.minitab.com/en-us/minitab/21/help-and-how-to/statistical-modeling/multivariate/how-to/principal-components/interpret-the-results/key-results/.

"FFT in Python — Python Numerical Methods." *Pythonnumericalmethods.berkeley.edu*, University of California Berkley, pythonnumericalmethods.berkeley.edu/notebooks/chapter24.04-FFT-in-Python.html.