

Guru Analysis: Using Python to Test Classical Financial Theories in the Modern Day

Senior project in partial fulfillment of the requirements of the Computer Science major at

Yale University

Zachary Kreiser

1. Introduction

If one could predict the future performance of stocks, the monetary benefits would be potentially endless. It makes sense, then, that many highly respected financial minds have attempted to create criteria and formulas to identify the stocks which will give the best return on investment. Some of these “Gurus” of finance have centered their criteria on conservative financial fundamentals such as strong dividend records and low debt ratios, while others have moved away from traditional indicators and have focused more on momentum and newer metrics such as active daily users. Each of these roadmaps for investing is backed up by its own theory and examples, and the advice of one guru can often directly contradict that of another, making it difficult to determine for certain whether an investment is sound or shaky. Luckily, if a guru has provided a concrete formula for evaluating stocks, these rules can be automated and back-tested in order to test the efficacy of that formula. This project aims to do exactly that - to automate the evaluation of stocks based on the formulas developed by two prominent financial gurus, and to back-test these formulas to determine whether they hold up in the modern age.

Specifically, the project aims to answer these two questions:

1. If one had invested in companies from the S&P 500 5 years ago based on the wisdom of Benjamin Graham and Peter Lynch, how would the investment have fared?

2. Does a correlation exist between if a company meets Graham or Lynch's guidelines and its total returns in the past 5 years? If so, how strong of a correlation?

1.2 Hypothesis

1. Based on Graham's focus on solid financials and Lynch's emphasis on growth, I predicted that if I back tested each formula for five years, stocks given a high score by the Graham formula would see modest growth while avoiding catastrophic loss during market downturns, while stocks scored highly by the Lynch formula would rise and fall more with the market, and that stocks scored highly by either formula would exceed the total market's returns.
2. With regards to a correlation, I predicted that scores returned by the Graham and Lynch formulas for a given company at a certain time would correlate positively with its total returns between that time and the present.

2. The Gurus

Benjamin Graham was 1894 and is commonly known as the father of value investing. His writings, including classics such as *Securities Analysis* and *The Intelligent Investor* represent a conservative view of financial analysis, emphasizing strong financial fundamentals such as low debt and price to earnings ratios and strong

dividend record. In keeping with his view that a company can be best evaluated based on its financial statements, he provided formulas in *The Intelligent Investor* to use these data to categorize investments as worthy or unworthy. This project determines a company's "Graham Score" using his formula for the enterprising investor (which is slightly less conservative than the formula for the defensive investor). Ideally, this formula will recommend companies with strong financials and minimal downside. The criteria are as follows:

- Price to earnings ratio must be below 15
- Current ratio (current assets divided by current liabilities) must 1.5 or higher
- Long term debt divided by net current assets must be 1.1 or lower
- Earnings over the past 5 years must be positive
- The company must have some current dividend record, i.e. it must have paid out a dividend in the past 12 months
- The company's earnings from the current year must be higher than the company's earnings five years ago
- The price to book ratio must be below 1.4

Peter Lynch was born in 1944 and gained a reputation as a legendary investor while serving as manager for the Magellan Fund at Fidelity Investments. During Lynch's tenure as manager between 1977 and 1990, the fund averaged nearly 30% annual returns, making it the best performing mutual fund in the world. As a result, Lynch is known by many as the best mutual fund manager in history. Lynch's investment

philosophy is markedly less conservative than Graham's, as it is willing to forgive less than stellar financials if a company's growth rate justifies it. This philosophy is best demonstrated through one of Lynch's favorite indicators, the Inverse Dividend-Adjusted Price to Earnings to Growth ratio. This project determines a company's "Lynch Score" based on this famous ratio, the criteria for which is given in Lynch's 1989 book *One Up on Wall Street*:

"Find the long-term growth rate (say, Company X's is 12 percent), add the dividend yield (Company X pays 3 percent), and divide by the p/e ratio (Company X's is 10). 12 plus 3 divided by 10 is 1.5" (199)

According to Lynch, any rate less than 1 is poor, above 1.5 is ok, and above 2 is ideal.

3. Technologies

Python was used for all coding due to its compatibility with Jupyter Notebook as well as its data science toolkits such as numpy and matplotlib.

Jupyter Notebook is a web application that allows developers to create documents which contain live code as well as equations and visualizations. Its ability to run isolated blocks of python code makes it the perfect tool for data scientists to perform data cleaning and transformations, numerical simulations, statistical modeling, data visualization, machine learning, and more. (via <https://jupyter.org/>)

Intrinio is a financial data platform which provides developers with an API to pull financial data including prices, fundamentals, ratios, and indicators for thousands of companies. Both current and historical data are available, an essential feature for backtesting formulas. (via <https://intrinio.com/>)

4. Methods

4.1 Functions

The functions and code to run the analysis are contained in `Guru_Equity_Analysis.ipynb`. The functions called by the user are `guru_analysis` and `plot_growth`, which each call respective helper functions. Here I will describe the program flow of each.

`guru_analysis` takes the name of a guru (either 'graham' or 'lynch'), a number of years to backtest, and a dictionary of companies to test, where the keys are the company names and the values are the respective tickers. It runs the respective guru analysis (either `Graham_recommends` or `Lynch_recommends`) on each company to get their respective scores at the specified number of years ago. It returns a list of objects containing the company name, the guru score, and the total return on investment had the company been bought at the specified number of years ago.

`Graham_recommends` and **`Lynch_recommends`** take a stock ticker and a number of years ago to collect data from. These functions are analogous in that they each

calculate a score for an investment at a certain time, based on the criteria given above for each Guru. They do so using helper functions which access the Intrinio API, clean the data for inconsistencies, and determine the right data to use. Often with smaller companies and some larger, Intrinio does not hold the right data needed to make the analysis. In these cases, an error is returned and the company being analyzed is excluded from any further analysis.

Graham_recommends returns a score from 0 to 1. There are 7 criteria, and a company gets a score from 0 to 1 in each of them. The returned score is the average of its scores for each criterion. Some criteria are simple pass fail while others can get a fractional value. For example, the dividend record criterion is pass/fail. If a company has a current dividend record, its score for that category is 1, and otherwise it is 0. The price to earnings is on a scale, so if the company's p/e ratio exceeds the threshold of 15, it receives a score of 1 minus the percent difference between the p/e ratio and 15.

Lynch_recommends has no limits on its return value. It simply returns the company's inverse price to earnings to growth ratio.

get_fundamental is the main function used to retrieve data about a company. It first calls `get_data_tenyr`, which makes the actual call to the Intrinio API and returns 10 years of quarterly data so that the datapoint from the right time can be found. Since financial data are often inconsistent, with filings missing for periods of months or years,

the function **get_datapoint** is then called on the ten years of data to ensure the returned datapoint fits in with the time period being analyzed. It does so by finding the datapoint taken closest to the current date minus the specified number of years, then ensures that datapoint's date is close enough to the desired time (within a year), or sets its value to 'nm' (not meaningful) otherwise.

plot_growth finds the monthly growth of a hypothetical investment in a company or group of companies, then plots it, optionally against the returns of an index such as the S&P500. It performs this task by calling `growth_of_investment` or `growth_of_bucket` on the stock or group of stocks provided, then uses the library `matplotlib` to graph the results.

growth_of_investment determines the monthly value of a hypothetical investment in a company at a certain date based on total returns. It does this by calling `Intrinio` to get the company's market cap each month since the specified date, adding back the dividends to get the total value for each month, then calculating the change from the previous month's total value to get the total growth rate. From there, calculating the hypothetical investment's monthly value is as simple as beginning with the starting amount, and multiplying it by the next month's total growth rate until the present is reached.

get_monthly_prices and **get_yearly_dividend** get their respective data from Intrinio and clean the data for inconsistencies in order to get the datapoint taken at the correct time, much in the same way that **get_datapoint** does.

add_dividends is where a company's dividend yield is added to its market cap to get its total value. To do this, the function takes in a list of monthly market caps for a company and a list of the dividend paid out by the company each year. For market cap data taken in the current year, the previous year's dividend value is used as an extrapolation since Intrinio only provides dividend data on a yearly basis on December 31. The dividend is added to the market cap month by month, so January 2014's total value equals the market cap plus one twelfth of 2014's dividend, February 2014's total value equals the market cap plus two twelfths of 2014's dividend, etc.

growth_of_bucket calls **growth_of_investment** for each of the stocks in the list it takes in, then consolidates the growth of an investment in each company into the growth of an investment in the companies together by taking the mean total value at each month across all companies queried.

4.2 Experiment

Using the tools above, I calculated the Graham and Lynch scores for every company in the S&P 500 based on their financials from five years ago. In order to avoid survivorship bias, the list of companies analyzed were taken from a list of S&P 500

members from five years ago. Since Intrinio holds 10 years of data for most companies and analysis requires 5 years of historical data for a company, 5 years was chosen as the amount of time to back test.

With the list of historical S&P 500 members having been imported, `guru_analysis` was run both for Lynch and Graham's formulas on the companies based on their financials from 5 years ago. To test the growth of an investment in high scoring companies, the top 25% of companies based on each formula were taken and analyzed using **`plot_growth`** over the period beginning 5 years ago. To test the correlation between the stocks' Graham and Lynch scores and their total returns, the numpy function `corrcoef()` was used.

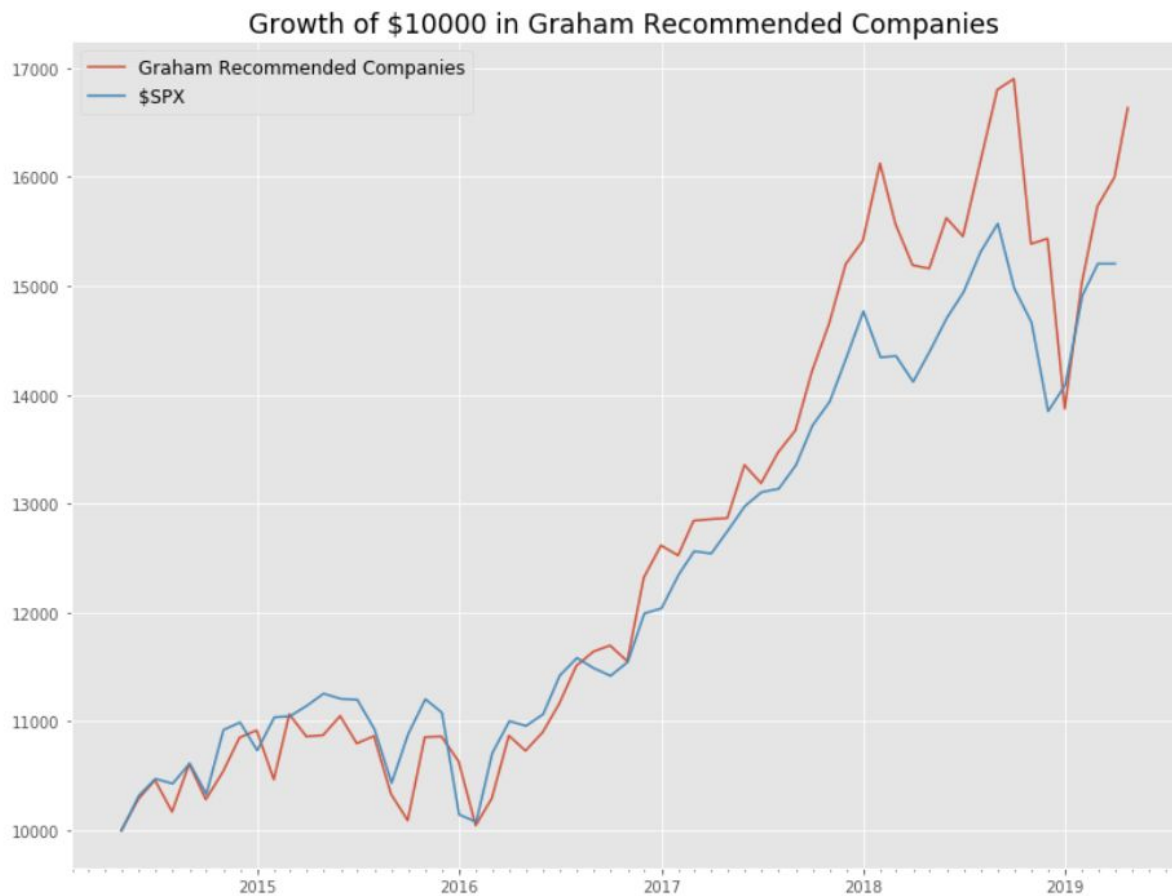
5. Results

5.1 Charts

```
plot_growth(graham_good_scores, '2014-04-17', index='$SPX', title='Graham I
```

Your investment of 10000 in this bucket on 2014-04-17 would be worth 16630.72 today

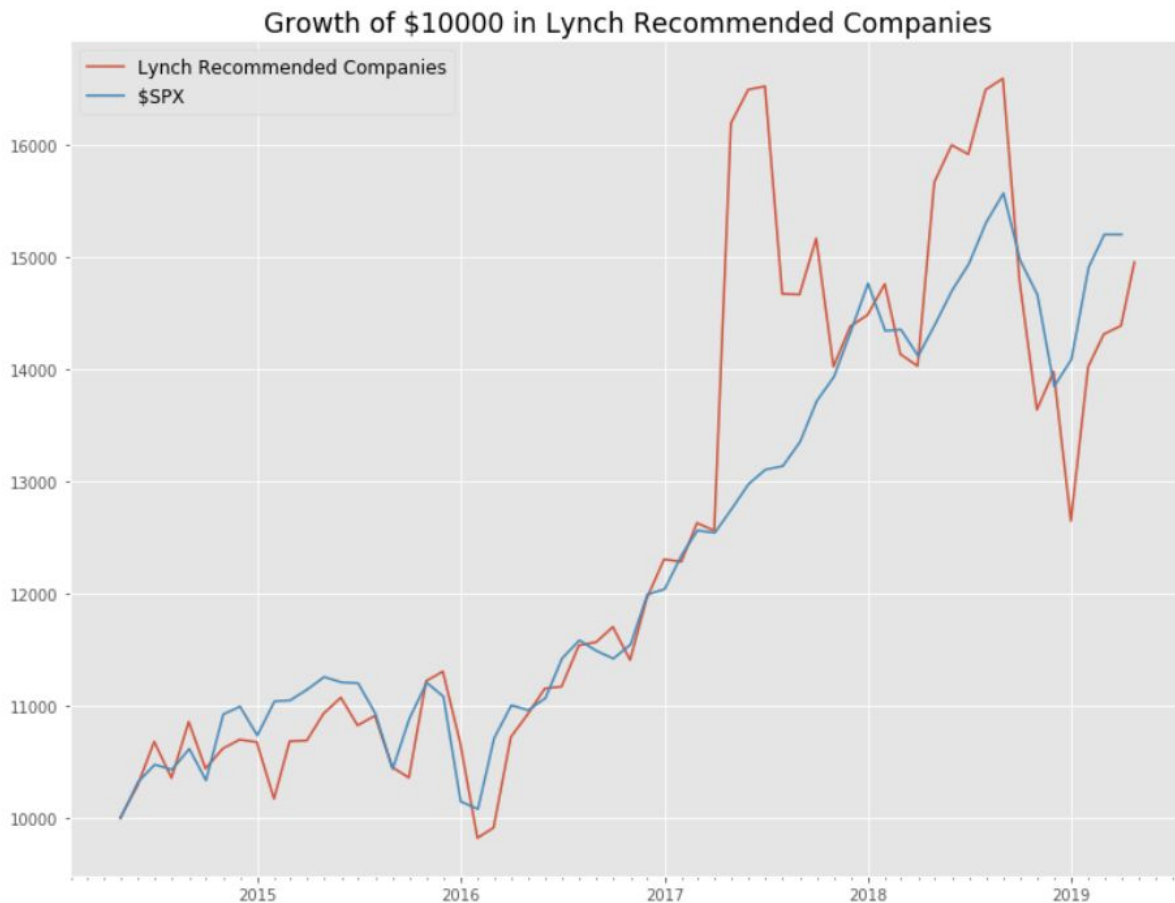
Your investment of 10000 in \$SPX on 2014-04-17 would be worth 15200.99 today



```
plot_growth(lynch_good_scores, '2014-04-17', index='$SPX', title='Lynch Re
```

Your investment of 10000 in this bucket on 2014-04-17 would be worth 14954.57 today

Your investment of 10000 in \$SPX on 2014-04-17 would be worth 15200.99 today



5.2 Data

The top 25% of companies based on Graham scores saw a total return of 66.3% over the 5 year period analyzed. The top 25% of companies based on Lynch scores saw a total return of 49.5% over the same period. Meanwhile, the S&P 500 overall appreciated by 52.0%.

The correlation coefficient between a company's Graham score and its total return over the time period was found to be -0.0024, while the correlation coefficient

between a company's Lynch score and its total return over the time period was -0.04996.

6. Discussion

The first revelation of note from this experiment is that no significant correlation exists between either a company's Graham score or the Lynch score and its total returns over the 5 year period, proving part 2 of my hypothesis wrong. While this finding does not prove either of these financial gurus wrong, it does support the argument that the market is highly complex, and predicting a stock's performance cannot be reduced to a simple formula. The market is moved by millions of investors, and regulators attempt to make it so that they all act on the same information. Therefore all of this information is typically accounted for in a stock's price, making any attempt to pick a stock based on its financial information no better than picking randomly.

While the lack of correlation means that a company with a high Graham score is in no way guaranteed to have a high return, the top 25% of Graham scoring companies did in fact significantly outperform the S&P 500 over the same time period. Looking at the graph of how a \$10,000 investment in these companies would have fared over time, it would seem to support the piece of my hypothesis regarding companies recommended by Graham's formula. For the first two years, the Graham recommended companies follow the S&P closely, occasionally dipping slightly below, then in late 2016 they exceed the index and never dip back below, even during market downturns. It is

possible that these companies avoid falling too far because of their strong financial backing as indicated by their high Graham scores

The companies with high Lynch scores, on the other hand, swing wildly above and below the market's returns, sometimes totally uncorrelated with the market's up and down turns, such as with the massive up-swing in 2017. This is likely due to the high price to earnings, high growth companies that are more likely to be highly volatile but that can still receive a high Lynch score based on their high inverse price to earnings to growth ratio. Since these companies do not need solid financials to get a high score, it is possible that they lack the safety cushion of the Graham recommended companies.

8. Problems Encountered

- **Problem:** Initially I was using a different API to pull company data from a company called SimFin, but the data from that service were missing a number of important indicators needed to evaluate companies via the Graham and Lynch formulas
 - **Solution:** Switched to a different API service (Intrinio) which provided all the necessary data. Switching APIs required me to rewrite much of the work I had done up to that point
- **Problem:** When evaluating a company's financial state 5 years ago, it is important that all of the data being examined were from around the same time. However, some companies' filings were inconsistent, and pulling 5 years of

quarterly data for long term debt (for example) would only return one or two filings per year. This meant finding the right filing to use based on its position in a list of quarterly data would not be accurate in every case.

- **Solution:** Wrote a number of functions to determine what datapoint was taken at the correct time such as `get_date()`, `closer()`, `is_later()`, `find_closest()`, and `get_datapoint()`.
- **Problem:** Some data is filed with a null value, or a value of 'nm' (not meaningful) and these values can break the program when it is doing calculations on them. Sometimes these values can still work within the model (e.g. a company lists a P/E ratio of 'nm' whenever it has negative earnings) and sometimes it means that data was not found, meaning the model does not have enough information.
 - **Solution:** Included checks for if values are null or 'nm' and handled those cases correctly for each situation, interpreting those values correctly if they are meaningful or throwing the company out if important data points are missing.
- **Problem:** Evaluating a company's performance over 5 years using only share price is difficult because of stock buybacks and splits, and it does not account for dividend returns.
 - **Solution:** Wrote a formula to determine the total return of investing in a company over a time period which uses the growth of the market cap and the dividend yield to give a better picture of total return on investment.

- **Problem:** Evaluating the Graham and Lynch scores for every one of the 16,000 companies in Intrinio's database would require more API calls than my license is allowed.
 - **Solution (partial):** Reduced the scope of my project from evaluating all 16,000 companies to just the companies on the S&P 500.

9. Future Work

If I had more time and resources, the first thing I would do is try to expand the program so that it can evaluate all of the companies in Intrinio's database instead of solely companies in the S&P 500. This would require little more than a higher tier Intrinio license which could make more API calls per day. A bigger sample size this would provide would make for even better insight into whether Graham and Lynch's formulas are good roadmaps for investing.

While this project did not establish a correlation between how well a company meets Graham or Lynch's metrics and its long term performance, I do believe those metrics can be a good jumping off point when researching a company to invest in, and Graham's formula in particular gives a good picture of if a company's financial state is strong. Therefore, I believe that the tool I created with this project could be useful to someone looking to invest in equities. Therefore, the next future improvement I would make to this project is the creation of some sort of platform with which users could evaluate stocks based on Graham and Lynch's formulas, and get a list of companies which get the highest grade based on their metrics.

Finally, I would like to improve this project by including more functions to evaluate companies based on the wisdom of other financial Gurus. Graham and Lynch are two titans of securities evaluation, but there are more financial minds who have distilled their investment philosophies into lists of criteria and rules, and it would be interesting to see how those fare as well.

References

1. Bajkowski, John. *Adjusting the Benjamin Graham Enterprising Investor Screen*.

Retrieved from

<https://www.aaii.com/journal/article/adjusting-the-benjamin-graham-enterprising-investor-screen>

Graham, Benjamin (2009). *The Intelligent Investor*. New York: Harper

Lynch, P., & Rothchild, J. (2000). *One up on Wall Street: how to use what you already know to make money in the market*. New York: Simon & Schuster.