Data Wrangling Project

An Analysis of Financial Metrics and Stock Performance

Matt Lancaster & Advika Shah

GitHub Repository Link: https://github.com/mdlancaster/Data-Wrangling-Project.git

Introduction:

Often, a company's stock price rises or falls substantially in response to an earnings announcement, but why is that the case? What about those reported financial metrics causes investors to revise their valuation of—and expectations for—the company? This project is dedicated to making sense of this complex relationship between a company's fundamentals and its stock performance on the Nasdaq exchange.

By identifying patterns and relationships between a company's financial position and its position in the market, our team was interested in producing insights valuable to both the company and the investor. For company management, these insights could benefit their business strategy and cost-reduction initiatives as well as the framing of their earnings announcements. For investors, understanding how financial indicators like net income or debt levels relate to stock price can improve their stock selection, asset allocation, and risk management. Every 10-K and 10-Q tells a unique financial story about a company, and this analysis is focused on examining how that story is perceived by investors and reflected in the market.

Data:

nasdaq_final.csv: Our first source of data was scraped off the Nasdaq Stock Screener website which includes current (as of 5/3/25) information on all Nasdaq-listed stocks. After scraping this information off the Nasdaq Stock Screener, we created a pandas data frame with six columns (Features 1-6 in the Data Dictionary below) and 6,561 rows (each representing a Nasdaq-listed company). This dataset is shown in GitHub as *nasdaq_final.csv* and its corresponding code is shown in the *nasdaq_scraping.ipynb* notebook.

yfinance_data.csv: Our second source of data was obtained through the yfinance Python library. It was not feasible to use yfinance to download stock information for all 6,561 Nasdaq-listed companies due to computing limitations, so our team randomly sampled 750 tickers from the Nasdaq data frame to download stock information on instead. This way, we were able to still represent the variety of sectors, stock prices, and market capitalizations present in the Nasdaq dataset while not exceeding our computing constraints.

After downloading the stock information, we created three data frames (final_bs, final_is, and final_cf) and assigned the companies' relevant financial indicators from their quarterly balance sheet, income statement, and cash flow statement to each data frame, respectively. Additionally, we created another data frame (stock_prices) for which the companies' stock prices from 1 month, 6 months, and 1 year ago were assigned. We inner joined the three financial metric data frames together on the Ticker column to create a merged data frame (fin_stmts). We then inner joined fin_stms with stock_prices on the Ticker column to produce our final yfinance data frame, yfinance_data. We noticed a few columns in yfinance_data contained a significant number of NaN values, so we dropped the columns with more than 100 observations missing. This left us with 570 rows (companies) and 20 columns (financial metrics and historical stock prices). The process used to produce this final dataset, *yfinace_data.csv*, is shown in the *yfinance_data.ipynb* notebook on GitHub.

final_data.csv: Our final dataset was created by inner joining the Nasdaq dataset and the yfinance dataset on the Ticker column. This resulted in a dataset with 25 columns and 570 rows. The code used to merge the two data frames and conduct some final data cleaning (e.g., reformatting columns' data types for later analysis) is shown in *data_final.ipynb* and the final dataset is saved as *data_final.csv.*

Data Dictionary – final_data.csv

Feature	Description
Ticker	Ticker symbol of the company's stock
Name	Company name
Last Sale	Last sale price of company's stock (as of 5/3/25)
Net Change	Net change in sale price from 5/2/25 to 5/3/25
Percent Change	Percent change in sale price from 5/2/25 to 5/3/25
Market Cap	Market capitalization (current share price * shares outstanding)
Shares Issued	Number of shares issued
Total Assets	Current plus Non-Current Assets
Net PPE	Net Property, Plant, and Equipment (book value of long-term, tangible assets after deducting depreciation)
Accounts Receivable	Amount of money the company is owed for goods/services sold on credit
Cash and Cash Equivalents	Amount of cash-on-hand plus highly liquid investments
Total Debt	Amount of short- and long-term debt
Accounts Payable	Amount of money due to suppliers/vendors for goods/services purchased on credit
Stockholders Equity	Residual value of assets after deducting liabilities
Retained Earnings	Amount of profit reinvested into the company
Working Capital	Difference between current assets and current liabilities
Total Revenue	Amount of income generated from selling products/services
Operating Income	Income generated after deducting operating expenses (EBIT)
EBITDA	Earnings before interest, taxes, depreciation, and amortization
Net Income	Profit after deducting all expenses
Capital Expenditures	Amount spent on long-term, tangible assets
Free Cash Flow	Cash flow generated after deducting cash flows that maintain operations and capital assets
1 Month Stock Price	Company stock price on 4/3/25
6 Month Stock Price	Company stock price on 11/3/24
1 Year Stock Price	Company stock price on 5/3/24

Analysis:

Question 1: **What is the relationship between company size and the percent change in stock price over one year?**

Reasoning & Hypothesis: This question explores the relationship between a company's size (as proxied by Market Cap) and the company's stock performance over the past year. We were interested in seeing whether larger- or smaller-cap companies exhibited larger one-year returns and what the variability was in those returns. We hypothesize that larger companies will have a larger, one-year percent change in stock price.

Process: We began by engineering a feature in our final dataset for the percent change in stock price over one year [(Last Sale – 1 Year Stock Price) / 1 Year Stock Price]. To measure company size, we created another feature, cap_quartile, and used pd.qcut() to label companies 'Micro Cap,' 'Small Cap,' 'Mid Cap,' or 'Large Cap' based on the quartile their market capitalization fell within. We created a violin plot (Fig. 1) to visualize the distribution of percent changes in stock price across these four quartiles, including the mean and median return for each group.

Results: Small Cap companies exhibited the widest variability in their one-year returns, followed closely by Micro Cap companies. Mid Cap and Large Cap firms showed more stable returns; however, unlike what we hypothesized, their mean and median return did not seem to be substantially higher than smaller-cap companies. From an investment perspective, this pattern could guide portfolio strategy—risk-averse investors may gravitate towards larger, more stable firms while those seeking a higher risk-reward investment might focus on smaller-cap stocks.

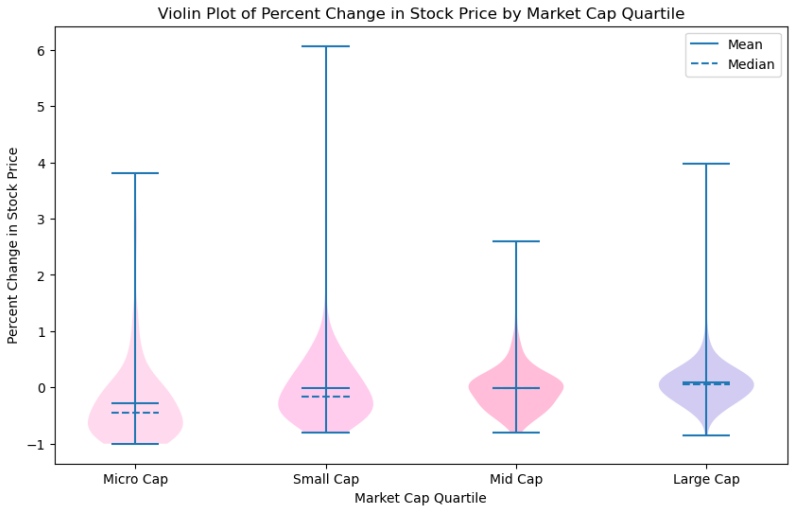

Fig. 1: Violin Plot with Distribution of Percent Change in Companies' Stock Price by Market Capitalization

Question 2: **How does the average stock price differ among companies categorized by profitability levels?**

Reasoning & Hypothesis: This question sought to identify whether there were any significant differences in average stock price between companies that were grouped by their profitability (as proxied by Net Income). It seems obvious that a company would be deemed successful if it were profitable, but we were curious to know whether this sentiment was shared by the market and for what "tier" of profitability. We predict the more-profitable groups of companies will exhibit a larger average stock price.

Process: We created a column named "Profitability" in our final dataset wherein companies that had a positive Net Income were labeled "Profitable" and companies with a negative Net Income were labeled "Unprofitable". For profitable companies, we used pd.qcut() to create three quantiles for which companies received a label of "Marginally Profitable," "Profitable," or "Highly Profitable" depending on which quantile their Net Income fell within. This process was repeated for unprofitable companies but with the labels "Marginally Unprofitable," "Unprofitable," and "Highly Unprofitable". We then grouped the companies by their profitability group and computed each group's average last sale price.

Results: The results are visualized in Figure 2 below and were largely in line with our expectations—the more-profitable groups exhibited higher average stock prices than the less-profitable groups. Application-wise, this result could be useful to investors as a benchmark for comparing a target company's stock price with its profitability-group's average as well as for identifying any anomalies (i.e., unprofitable companies with unusually high stock prices).

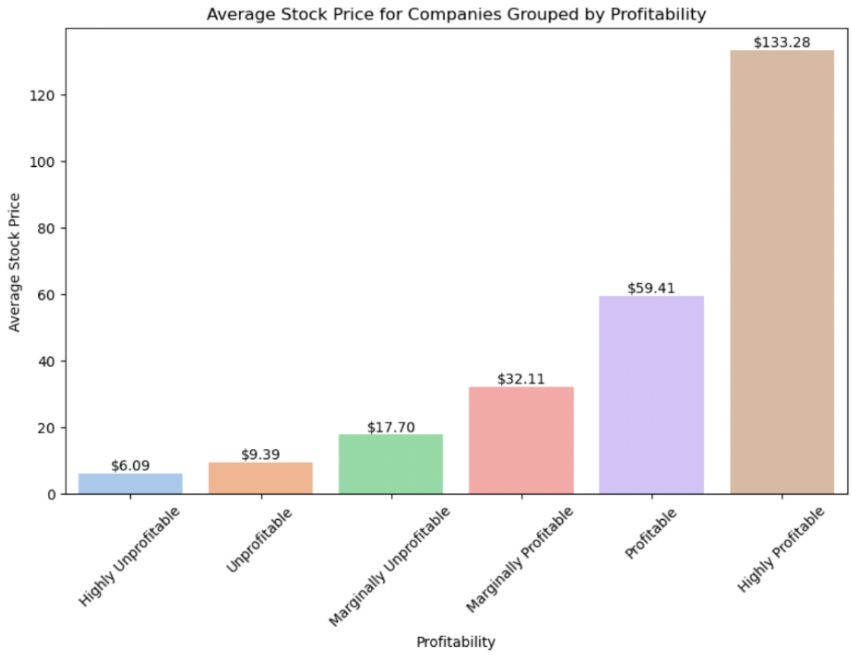

Fig. 2: Bar Chart of Avg. Stock Price by Profitability Group

Question 3: **Is there a statistically significant difference between companies' average historical stock prices (one-year vs. one-month and six-month vs. one-month)?**

Reasoning & Hypothesis: Our team was interested in investigating the relationship between company stock prices across different time periods, so we decided to use t-tests to compare companies' average historical stock prices. We hypothesize that there is a statistically significant difference between the one-year and one-month stock prices as well as between the six-month and one-month prices.

Process: To get a sense of the distribution of one-month, six-month, and one-year stock prices, our team decided to conduct univariate analysis by creating histograms (Fig. 3-5 below) for each period. To conduct the t-test for one-month versus six-month average prices, we began by dropping missing values from the final dataset and then we ran a paired t-test from scipy.stats for the 1 Month Stock Price and 6 Month Stock Price columns. This process was repeated for the 1 Month Stock Price and 1 Year Stock Price columns. We chose to use a paired t-test (ttest_rel) since we hypothesized that the stock prices across different time periods were related.

Results: The t-test for one-month and six-month stock prices resulted in a test statistic of -4.22 and a p-value of effectively 0 which indicates a significant difference between the average stock prices. While this was in line with our expectations, the t-test for one-year and one-month prices did not follow our prediction by resulting in a test statistic of -1.380 and a p-value of 0.169—implying no difference between the two average stock prices. These results indicate that stock prices may show more volatility in a shorter timeframe and stabilize over longer periods.

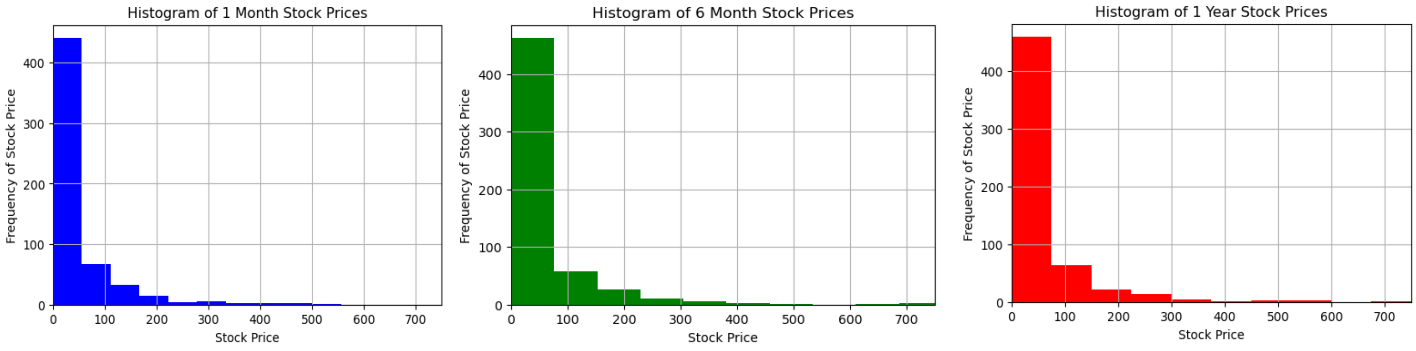

Fig. 3-5: Histograms of 1-Month, 6-Month, and 1-Year Stock Prices (from left to right, respectively)

(Machine Learning) Question 4: **To what extent can Total Assets, Total Debt, Net Income, Stockholders' Equity, and Market Capitalization predict the direction of stock price changes?**

Reasoning: This question focuses on predicting whether the one-year return for a company's stock was positive or negative given a collection of financial metrics. We wanted to better understand what predictive power, if any, a company's financial metrics had in relation to stock behavior.

Feature Selection: The features we chose as predictors were among those with the lowest number of NaN values and each had a unique insight into the full financial picture of a company. Total Assets gives insight into the scale, resources, and stability of a company; Total Debt indicates how leveraged a company is; Net Income measures the profitability of a company; Stockholders' Equity represents the net worth of a company in the eyes of investors; and Market Cap reflects a company's size and the market's valuation for the company.

Methods: We began by transforming the pct_change_stock_price feature (created in Question 1) into a dummy variable, pct_change_dummy. A positive return was labeled 1 and a negative return was labeled 0. After assigning Total Assets, Total Debt, Net Income, Stockholders' Equity, and Market Cap as the x variables and pct_change_dummy as the y variable, we split the data (80/20) into training and test sets. Four classification models—K-Nearest Neighbors, Decision Tree, Support Vector Classifier, and Logistic Regression—were then fit onto the training data.

Results: The Support Vector Classifier performed best with an accuracy of 73% and an F1-Score of 0.65, as shown in the table below. Logistic Regression, in contrast, underperformed with an accuracy of only 46%. These findings show promise in using financial metrics to predict a positive or negative stock return; however, additional features like broader, economic indicators may be required to improve the accuracy of the predictions further.

	Machine Learning Method	Accuracy	F-1_Score
0	KNN	0.56	0.57
1	Decision Tree	0.59	0.60
2	SVC	0.73	0.65
3	Logistic Regression	0.46	0.48

Table 1: Model Performance in Predicting Stock Price Direction Using Financial Metrics

(Machine Learning) Question 5: **Can key financial indicators (Net Income, Stockholders' Equity, Market Capitalization, Total Debt, Total Assets, and Cash and Cash Equivalents) be used to predict whether a company has more accounts payable than accounts receivable?**

Reasoning: This question seeks to predict whether a company's payables exceed its receivables given a collection of financial metrics. Predicting the relationship between payables and receivables interested us because it could help determine whether a company is facing difficulties collecting their receivables or at risk of defaulting on their payables.

Feature Selection: The features used in our analysis were chosen largely by the same criteria as Question 4. Cash and Cash Equivalents was included to provide insight into a company's cash management which is closely tied to the company's payables and receivables.

Methods: We created a new dummy variable, accounts_dummy, in our final dataset to indicate whether a company's Accounts Payable exceeded its Accounts Receivable (1 = A/P > A/R, and 0 = AP < A/R). This dummy variable was used as our target and Net Income, Stockholders' Equity, Market Capitalization, Total Debt, Total Assets, and Cash and Cash Equivalents were used as predictors. We split the data using an 80/20 split and fit the same four classification models (K-Nearest Neighbors, Decision Tree, Support Vector Classifier, and Logistic Regression) on the training data.

Results: This time, the KNN model performed best, achieving an accuracy of 65% and an F1-score of 0.62. Logistic Regression again underperformed with the lowest accuracy (37%) and F1-score (0.20). The results of our models (shown below in Table 2) indicate a moderate relationship exists between a company's financial profile and its balancing of payables and receivables which could be beneficial in warning of a company's cash mismanagement.

	Machine Learning Method	Accuracy	F-1_Score
0	KNN	0.65	0.62
1	Decision Tree	0.59	0.60
2	SVC	0.63	0.49
3	Logistic Regression	0.37	0.20

Table 2: Model Performance in Predicting if A/P Exceed A/R Using Financial Indicators

Conclusions:

Overall Conclusions: Our first key takeaway is that large companies represent a more stable investment over a one-year timeframe, as their returns exhibited far less volatility than smaller companies in our first analysis. Our second analysis showed that groups of more profitable companies command a higher average stock price than groups of less profitable companies. This reflects the value attributed to net income by investors and could be a useful tool for benchmarking potential investments. Our third analysis indicated that companies' average stock prices show more volatility in the short-term and stabilize over longer periods—an insight that could benefit investors' investment horizons. Lastly, our machine learning analyses suggest that certain financial metrics are not limited to predicting only the relationship between other metrics (as in Question 5) but can be used to predict stock behavior as well (as in Question 4).

Project Limitations: The main limitation in our analysis was centered around using yfinance to create our second dataset, *yfinance_data.csv*. Due to computing limitations, we weren't able to download information for all of the companies in the Nasdaq dataset and instead had to randomly sample a collection of companies which, although fairly representative, wasn't as comprehensive as we had hoped. Secondly, there exists a lot of variety in the ways companies report certain financial metrics (e.g., 'Net Income' vs. 'Net Income Available to Shareholders') which wasn't fully captured by yfinance and resulted in a lot of NaN values. Especially for our machine learning questions, our accuracy suffered as a result of having to drop all these values.

Suggestions for Future Analysis: For future analysis, our team's first recommendation would be to supplement yfinance data with scraped information from the SEC's Electronic Data Gathering, Analysis, and Retrieval (EDGAR) system. This could bolster the financial metrics used in the analysis by filling NaN values provided by yfinance and allowing for more metrics (and quarterly statements) to be included. Similarly, our group would recommend including more market characteristic data in the analysis. Specifically, we suggest including a greater number of historical prices which, in conjunction with more financial statements, could make for an interesting analysis regarding the short-term effects of an earnings report on a company's stock price.