CS-GY 6313 / CUSP-GX 6006: Data Visualization - Spring '24

Homework #3

This homework is due 2 weeks after the release of this assignment: Monday April 17th, 2024.

> *This homework consists of three (3) required segments, totaling up to 15 points. Points are more likely to be awarded if code implementations come with comments explaining your process, even if the answer is incorrect. Submit your code as a* `.zip` *file together with your code and any other submittable items such as figures or writeups. You are permitted to use external Python libraries for your work.*
>
> *Adhere to NYU's Academic Integrity rules. Your answers and implementation must be your own. If you cooperated with another student from the course, please list either their name or NetID either as a comment in the code implementation or in Markdown. If you relied on external resources, websites, publicly-available code, etc. to answer any part of this assignment, you are expected to add an attribution to the original source material (ex. a comment, or a URL).* ***Using ChatGPT to generate code is explicitly not allowed and will result in an automatic 0 points for this assignment.***

Temporal Data Visualization (15 points)

In this homework, you will be working with visualizations of stock data. For the purposes of this assignment, we'll be using pre-extracted datasets regarding stocks for 4 major tech companies:

- **Meta (formerly Facebook):** `./datasets/meta.csv`
- **Amazon:** `./datasets/amzn.csv`
- **Apple:** `./datasets/aapl.csv`
- **Google:** `./datasets/goog.csv`

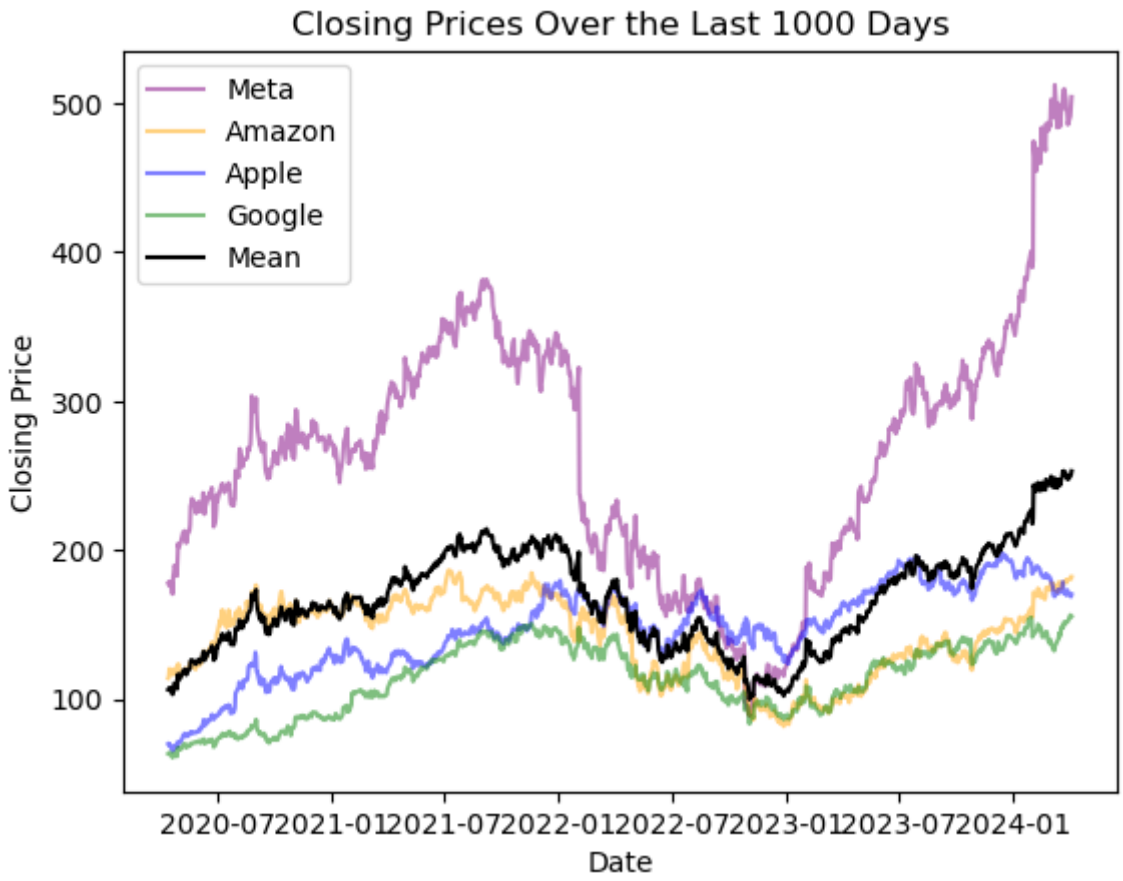
The data for each consists of stock data for the last 1000 days since the release of this assignment. You'll notice that there are 3 important columns:

1. `dates`: the datetime the stock data was recorded.
2. `closing price`: the raw price or cash value of the last transacted price in a security before the market officially closes for normal trading. It is often the reference point used by investors to compare a stock's performance.
3. `volume`: The number of shares traded for that stock. Traders usually look at this metric to determine liquidity of the stock.

Throughout this assignment, we'll be generating visualizations of these data and analyzing what these visualizations mean for each company. For code work, we'll be using `pandas` to get the specific information we want (e.g. closing price). We'll provide example code on how to load in the required data, but your task is to complete the code by following the instructions below.

Part 1: Curve-Based Visualization (5/15 points)

Temporally visualize the closing prices for the 1000 trading days of the 4 stocks as well as the mean (average) closing price across all 4 stocks for each day. Make sure to visualize all data in a single diagram. Color-code the plot so that each stock has a different color. Add a legend to let us know which plot represents which stock/data. An example is provided below:



Hints: Take advantage of some of `pandas`'s in-built functions. You don't necessarily need to use these, but here's some reminders:

- `df.tail(n)`: returns the last `n` rows of a dataframe.
- `df.reset_index()`: sets the first row's index to `0`.
- `df.copy()`: returns a copy of a dataframe. Changing this copy does not change the original dataframe.

Grading Metric:

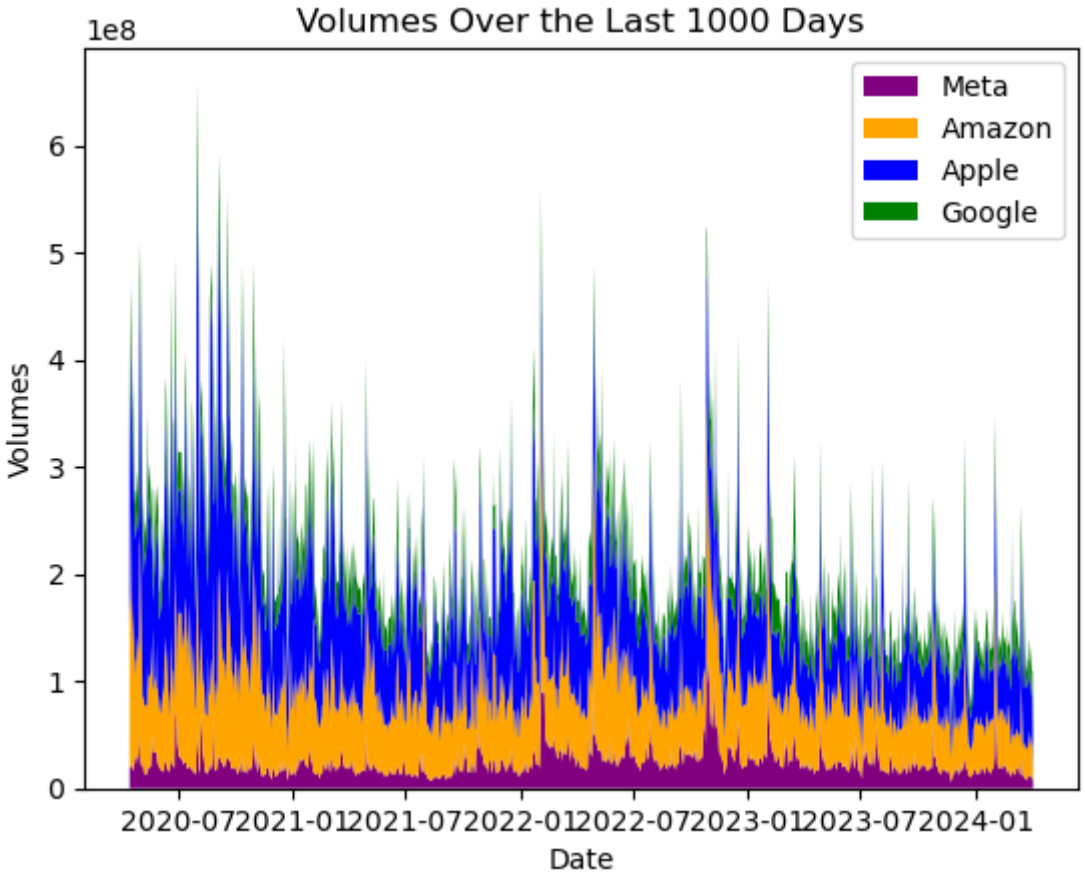
- We'll run your code to see if the visualization is generated properly. (1 point).
- Points are assigned based on how accurately the implementation follows the instructions:
  - Use of a single plot diagram (1 point)
  - Use color-coding to visually distinguish each stock and the mean data (2 points)
  - A legend is provided to identify what data which plot represents (1 point)

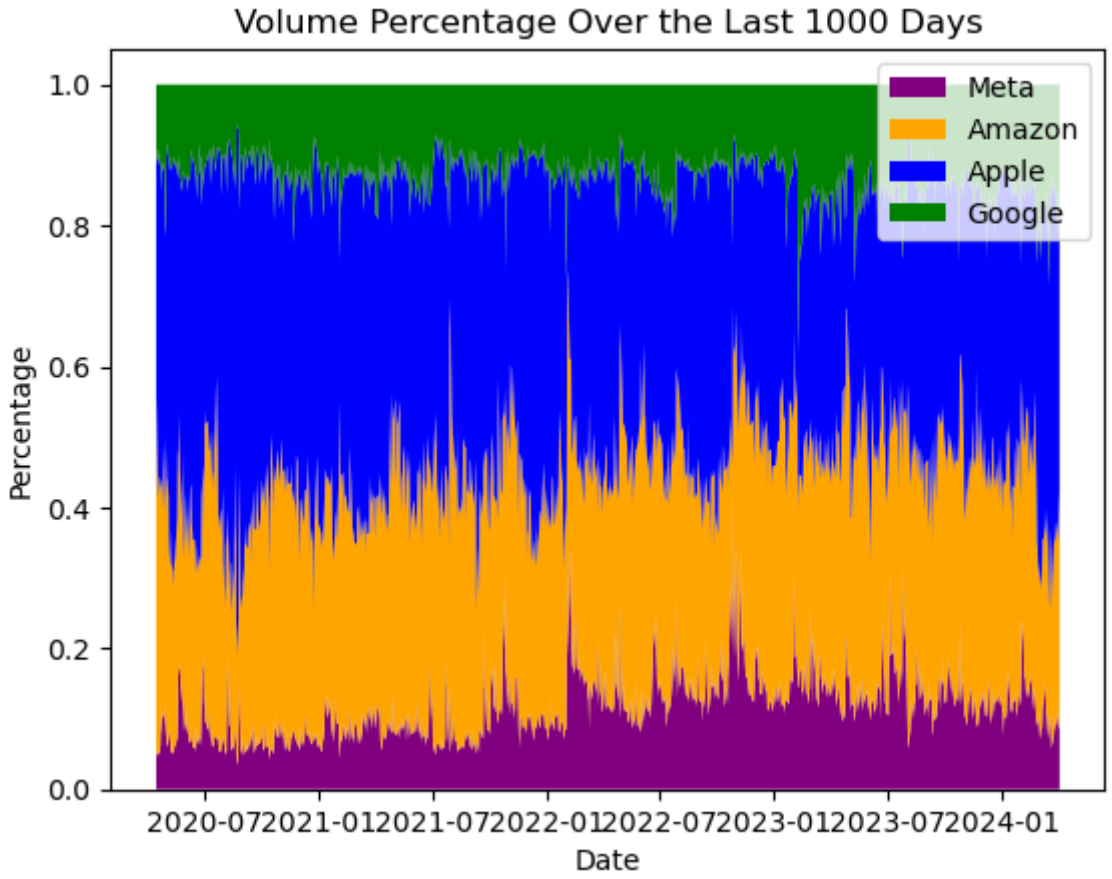
Part 2: Stacked Area Visualization (8/15 points)

Your task here is to plot two different plots. The first plot is the *stacked area chart* of the *volumes* of the 4 stocks. The second plot is the same, except you need to plot the ***percentage*** *stacked area chart*. To achieve the latter, you will need to find the percentages of the total volume for each of your stocks on each day.

Make sure to color-code your area plots so that it's easy to distinguish the stocks, and add a legend to help us identify which data represents which stock. There is no need to visualize the mean data here.

Below are examples of what your plots might look like.





Grading Metric:

- We'll run the code to check if two distinct visualizations are produced - either as subplots of a larger figure or as two separate figures. (2 points)
- For each plot:
    - Use of a single plot diagram for each subplot (1 point indiv., 2 points total)
    - Use color-coding to visually distinguish each stock and the mean data (1 point indiv., 2 points total)
    - A legend is provided to identify what data which plot represents (1 point indiv, 2 points total)

Part 3: Meaning behind the Data? (2/15 points)

We have three plots we generated from the stock data, but what do they each mean? Let's take some time to analyze these plots specifically. Within 2-5 sentences for each, answer the following:

1. Do you notice any trends in the closing prices of each company's stock? What might these patterns indicate regarding the performance of these tech companies? (1 point)

2. Who occupies the biggest volume percentage across the four companies? What does this tell you about the performance of these tech companies? (1 point)

Feel free to write down your analyses as markdown inside the `hw3.ipynb` notebook or as a standalone document.