Stock Market Analysis Capstone Project

PIERIAN DATA

Python for Finance

- Welcome to your first large project!
- This project will guide you through using all the skills we've covered in the first half of the course.

Python for Finance

- Everything you need is located under the folder:
- Stock-Market-Analysis-Capstone-Project
- There is an exercise notebook and a solutions notebook, as well as csv files for you to use.

Python for Finance

- Note that this project exercise is optional.
- You can treat it as an exercise or jump to the solutions lecture and treat it as a code along project!

Python for Finance

- There are a few bonus tasks (such as the candlestick visualization) that require exploring the documentation, feel free to skip these.

Python for Finance

- Another major part of this project is to slowly introduce a few new financial concepts, basic things like returns and cumulative daily returns.

Python for Finance

- The methods for these financial analysis techniques are described thoroughly in the notebooks, so quite a bit of reading will be involved with this project to make it a full learning experience!

Python for Finance

- One last note, the notebook will want you to use pandas-datareader, however, some geographical locations, firewalls, or computer settings may limit your use of pandas-datareader, so all csv files are provided just in case!

- Let’s explore the exercise notebook so you can decide how you would like to approach the project!

Let’s get started!

Stock Market Analysis Solutions Part Four

PIERIAN DATA

- This last part of the project focuses on cumulative returns, let’s discuss what that actually means!

- A cumulative return is the aggregate amount an investment has gained or lost over time, independent of the period of time involved.

Python for Finance

- This is different than just the stock price at the current day, because it will take into account the daily returns.
- Keep in mind, our simple calculation here won't take into account stocks that give back a dividend.

- Lets us say there is a stock 'ABC' that is being actively traded on an exchange.
- ABC has the following prices corresponding to the dates given...

Date	Price
01/01/2018	10
01/02/2018	15
01/03/2018	20
01/04/2018	25

Python for Finance

- Daily Return : Daily return is the profit/loss made by the stock compared to the previous day.
- A value above one indicates profit, similarly a value below one indicates loss.

Python for Finance

- It is also expressed in percentage to convey the information better.
- When expressed as percentage, if the value is above 0, the stock has given you profit otherwise it is a loss.

Date	Daily Return	%Daily Return
01/01/2018	10/10 = 1	-
01/02/2018	15/10 = 3/2	50%
01/03/2018	20/15 = 4/3	33%
01/04/2018	25/20 = 5/4	20%

- Cumulative Return:
    - While daily returns are useful, it doesn't give the investor an immediate insight into the gains he or she had made till date, especially if the stock is very volatile.

Python for Finance

- Cumulative return is computed relative to the day investment is made.
- If cumulative return is above one, you are making profits else you are in loss.

Python for Finance

Date	Cumulative Return	%Cumulative Return
01/01/2018	10/10 = 1	100 %
01/02/2018	15/10 = 3/2	150 %
01/03/2018	20/10 = 2	200 %
01/04/2018	25/10 = 5/2	250 %

Python for Finance

- The formula for a cumulative daily return is: $i_i=(1+r_t)*i_{t-1}$
- Here we can see we are just multiplying our previous investment at i at t-1 by 1+our percent returns.
- Pandas makes this very simple to calculate with its cumprod() method.

Python for Finance

- Let’s jump to the jupyter notebook and code through this!