

Homework 1

Rules: document your results in a Word file. You should not insert code into the documentation (unless specifically requested). Insert the plots you create into the documentation as well. In the documentation you should indicate which task you are writing about. **You are not allowed to use for-cycles unless instructed otherwise!**

Tasks:

- 1) Load the S&P500 weekly dataset!
- 2) Set the dataframe index to the dates contained in the dataframe, then drop that column as it is no longer useful and only would make everything harder. Copy the output from the console into the documentation!
- 3) Convert the prices of the Dell and HP stocks to simple gross returns, simple net returns and log-returns, and store them in a new dataframe (both stocks in one dataframe, every different return-type in a different dataframe, and there should be no NaN-s in them)!

$$\text{simple gross return: } R_t = \frac{P_t}{P_{t-1}}$$
$$\text{simple net return: } R_t = \frac{P_t - P_{t-1}}{P_{t-1}}$$

$$\text{log - return: } r_t = \log \frac{P_t}{P_{t-1}}$$

- 4) Plot the different returns. What are your conclusions?
- 5) Create a plot with four subplots: the ones on the left should contain the prices; the ones on the right should contain a return of your choice. The first row should contain the Dell, the second the HP. Give titles to the subplots. Compare them!
- 6) Calculate the mean and standard deviation of the returns of the two stocks. Which stock would you buy and why?
- 7) How many times did the stocks achieve at least 10% return? Calculate this with every return type and insert your code into the documentation! (It should be 3 lines of code altogether!)
- 8) Implement a basic pairs-trading strategy! Evaluate how well it would have worked! This is a bit heavier task, you can use for-cycles for this one.
 - a. Copy the original dataset that contains the prices into a new dataframe! If you have NaN-s in some columns then you should fill them with the last non-NaN value. Drop the columns that still have NaN-s. (Hint: fillna, dropna). Did any columns fall out compared to the original? (hint: dataframe.shape)
 - b. For now keep only the first 10 columns – this way you will see results sooner, but later you can remove this to include every asset.
 - c. Convert all the prices to simple gross returns!
 - d. The strategy is described as follows
 - i. There are n assets in the dataset
 - ii. There are t observations in the dataset
 - iii. The cross-correlation matrix with lag k is defined as

$$C \in R^{n \times n}$$

$$C_{ij} = Cor(R_{i,k:t}, R_{j,1:t-k})$$

iv. Finding the most correlated pair, i and j:

$$i, j = \arg \max_{i,j} \{|C_{ij}|\}$$

v. Implement and evaluate the strategy:

	$C_{ij} < 0$	$C_{ij} > 0$
$R_{j,t} < 1$	buy	sell
$R_{j,t} > 1$	sell	buy