Python Test for Candidates

Python results files should be well documented

Think carefully about the performance of your algorithms.

Question 1. ***Time Series Comparison.***

The goal in this question is to compare two datasets of securities.

In question1.zip you will find 3 CSV files:

1. *bbg_data_final.csv* – list of securities and the price and total return price series for each security.
2. *quandl_data_final.csv* – list of securities
3. *match_final.csv* – list of tickers to match the tickers from bbg_data_final.csv to the tickers from quand_data_final.csv.

The function *find_differences* in the file question1.py should find out for each security whether the "Price" time series from bbg_data_final matches the "Close Price" time series from the quandl_data_final.csv dataset. It should also find the the start date of the price series of time series.

The output should be either printed or saved in CSV file format.

For example, the matching line for the security WVG in the output.csv file will be:

Quandl Ticker, BBG Ticker, Quandl Data Start Date, BBG Data Start Date, Match since common start date?
…
…
"ABC", "WVG US Equity", "2000-01-03", "1990-01-02", "False"
…

Please implement the function *find_differences* in the file *question1.py*, and provide the output file.

Question 2. ***Diversification Ratio.***

Assuming a portfolio of N assets, and a weight w_i for each asset, we define the rolling DR ratio as follows:

$$DR(t, ws) = \frac{\sum_i weight_i(t) \cdot \sigma_i(t, ws)}{\sigma_p(t, ws)}$$

Where:

$$\sigma_i(t) \; is \; the \; standard \; deviation \; of \; asset \; i \; daily \; returns \; at \; time \; \boldsymbol{t}, \; taking \; into \; account \; a \; window \; size \; \boldsymbol{ws} \; (last \; \boldsymbol{ws} \; daily \; observations)$$

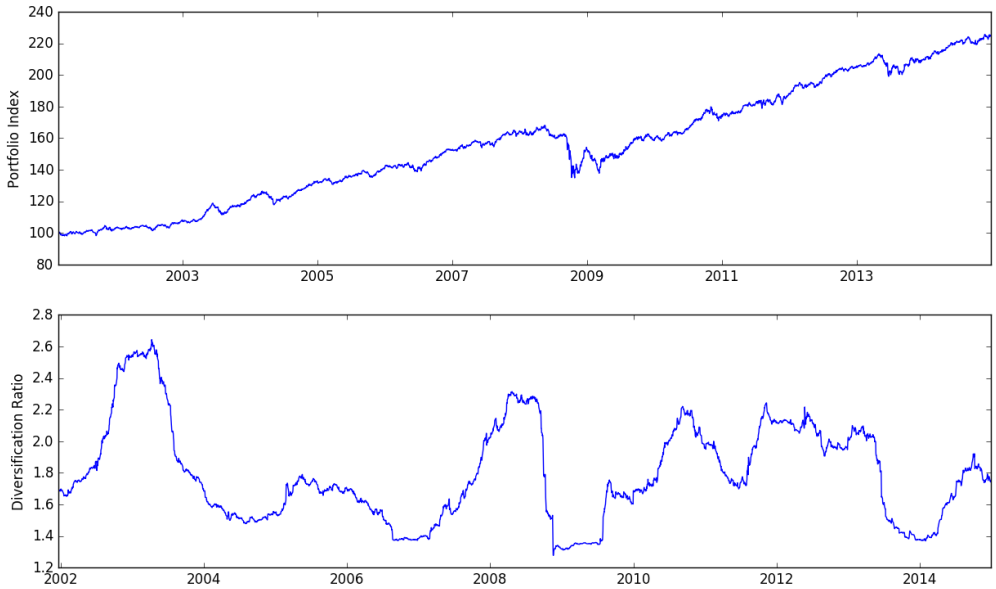
$$\sigma_p \; is \; the \; standard \; deviation \; of \; the \; portfolio \; daily \; returns \; at \; time \; t, \; taking \; into \; account \; a \; window \; size \; \boldsymbol{ws} \; (last \; \boldsymbol{ws} \; daily \; observations)$$

$$weight_i(t) \; is \; the \; weight \; of \; asset \; i \; (in \; the \; portfolio) \; at \; time \; t.$$

In question2.zip you will find a CSV file named *dr.csv*, and a python script named question2.py.

dr.csv contains for 4 assets their daily returns, their weights, and also the portfolio index performance.

For example, this would be the plot the function will create for DR for a rolling_window_size of 200 days for the provided dr.csv CSV file $DR(t, 200)$.



Please implement the methods *rolling_dr_ratio* and *plot_dr_ratio* in *question2.py* so it will create a plot of the rolling DR ratio.