Energy Analyst Programming Exercise – Analytical Tasks

Use of AI Tools: Please ***DO NOT*** use any AI-based tools (e.g., ChatGPT, Gemini, etc.) to generate code that you submit for your response or consult other people directly during the exercise. You may use AI tools to efficiently locate information for solving specific problems, but **do not use them to generate code you submit**. While AI tools can be excellent productivity resources for on-the-job work, this exercise is intended to assess your personal ability to generate code, not to understand how you may use AI tools to assist in this process. **Your compliance with this request is on the "honor system"**, and we ask for your word that you will utilize only approved resources when completing the exercise.

Before beginning, be sure to create a clean new directory that will hold all the output you produce as part of this exercise (CSV files, images, etc.).

Task 1: Read in all the historical data and combine it into a single data structure (data frame, table, etc.).

Locational Nature of Power Prices

Power prices vary by location on the grid due to supply/transmission availability, regional demand, and many other factors.

- **Task 2**: Compute the average price for each settlement point and year-month in the historical dataset (48 year-months: January 2016 through December 2019).
    - Compute these monthly average prices for **all** settlement points (both hubs and load zones) in the historical dataset. Hubs are denoted by the prefix "HB_" and load zones by the prefix "LZ_" in the SettlementPoint name.
    - Do not filter out prices less than or equal to zero in the historical data when computing averages. Negative prices are indeed possible in deregulated power markets.
- **Task 3**: Write the computed monthly average prices to file as a CSV named `AveragePriceByMonth.csv`.
    - The CSV file you write should have four columns with the following names: SettlementPoint, Year, Month, AveragePrice.

Price Volatility

Hourly volatility is defined as the standard deviation in log returns of hourly prices.

- **Task 4**: Compute the hourly price1 volatility for each year and each settlement hub in the historical power price data (hubs are denoted by the prefix "HB_" in the SettlementPoint name).
    - Since the natural logarithm is only defined for positive values, be sure to first filter out any prices that are zero or negative before computing log returns.
    - Do not compute volatilities for the load zones in the data (prefix "LZ_" in the SettlementPoint name) for this question.
- **Task 5**: Write the computed hourly volatilities for each settlement hub and each historical year to file as a CSV named `HourlyVolatilityByYear.csv`.
    - The CSV file you write should have three columns with the following names: SettlementPoint, Year, HourlyVolatility.

    - Column names *are case sensitive*.
- **Task 6**: Determine which settlement hub showed the highest overall hourly volatility for each historical year. Write code to extract the rows of your computed hourly volatility table corresponding to these maxima and write this table to a second output file named `MaxVolatilityByYear.csv`.
    - Your file should contain the same column names as the file you wrote in Task 5.

Data Translation and Formatting

cQuant's price simulation models consume hourly historical spot price data as input. Each price variable represents a distinct settlement point on the grid. Each price variable's historical data is submitted to the model in a separate CSV file that conforms to a specific format. Examples of formatted input historical spot price data files for three hourly price variables are provided in the supplementalMaterials folder within the Git repo.

- **Task 7**: Use the files in the supplementalMaterials folder as examples of the required data format, write code to translate the power price data structure you created in **Task 1** into a format that could be readily consumed by the cQuant price simulation models, and write the data to separate files for each settlement point.
    - The historical data provided to you uses the hour-beginning convention.
    - Each SettlementPoint should have its historical data saved to a separate file. That is, each CSV file you write should have data for one and only one SettlementPoint. There should be 15 files in total; one for each of the settlement points represented in the historical power price data.
    - To keep things tidy, save these files in a subdirectory named `formattedSpotHistory` within your main output directory for the exercise.
    - Use the convention "`spot_<SettlementPointName>.csv`" for the filenames of the data you write to file.

Bonus – Mean Plots

Generate two line plots that display the monthly average prices you computed in Task 2 in chronological order, according to the following specification:

- The first plot should show the monthly average prices for all settlement hubs only.
- The second plot should show the monthly average prices for all load zones only.
- In both plots, all relevant settlement points of a given type should be overlaid on the plot as separate curves, with a different color assigned to each curve. The plot should include a legend that allows the user to decode which color maps to which settlement point.
- Save both plots to file as either PNG or JPEG image files, with the filenames `SettlementHubAveragePriceByMonth.png` and `LoadZoneAveragePriceByMonth.png` (or corresponding filenames for JPEG files).
- **Hint**: To preserve chronological order, you should create a date column that you use for the x-axis dimension in your plots. You may assign each computed monthly average to the first day of the month to which it applies. This way, your plots will have x-axis tick labels of “2016-01-01”, “2016-02-01”, etc.

Bonus – Volatility Plots

Create a plot/plots that compare volatility across settlement hubs by year that you computed in Task 4. Save the plot(s) in your output directory. Use your judgement for how to present the data in the most visually effective manner. You may create as many plots as you feel is appropriate.

Bonus – Hourly Shape Profile Computation

An hourly shape profile is a normalized version of the 24 average hourly prices for a given price location over some period of historical data. It helps illustrate how the price varies, on average, by month of year, day of week, and hour of day.

Compute normalized hourly shape profiles from the historical spot price data for each settlement point by month of year and day of week.

- Each hourly shape profile has 24 values, one for every hour of the day, and there are 12x7=84 hourly shape profiles for each price variable, one for each combination of month of year and day of week.
- The average across the 24 hours in each hourly shape profile must be exactly equal to one.
- Each SettlementPoint should have its hourly shape profiles saved in a separate file. That is, each CSV file you write should have data for one and only one SettlementPoint. There should be 15 files in total; one for each of the settlement points represented in the historical power price data.
- To keep things tidy, save these files in a subdirectory named `hourlyShapeProfiles` within your main output directory for the exercise.
- Use the convention "`profile_<SettlementPointName>.csv`" for the filenames of the data you write to file.

Bonus – Open-Ended Analysis

If you have completed responses to all other tasks, with what time you have left, write code to analyze the data further in some way of your choosing. That is, tell us and/or show us something interesting about the data. This is a chance for you to show off. Plots, models, machine learning, you name it: it's all in bounds. You may provide narrative around your analysis within comments in your code; we will read the comments and the associated code for your analysis upon reviewing your response. Save any relevant output to your output folder.

Again, this question is entirely open-ended; use your intellectual curiosity and analytical prowess to guide you to some interesting data-driven insights.