46040 - Introduction to energy analytics F25

Assignment 2: Forecasting Electricity Prices

Deadline: Tuesday, April 22, 2025 at 23.59 hrs.

The goal of Assignment 2 is to become familiar with statistical, machine learning, and optimization tools for extracting value out of data. Our specific goal is to design a forecasting tool for electricity prices. If you are able to forecast prices, then you can combine this with your optimization tool from Assignment 1 to determine the appropriate day-ahead bidding strategy for your battery.

- Who are you? A battery owner.
- Where are you? Somewhere in DK2 zone of Denmark.

You will use the electricity spot prices from 2019 until 2024 and a set of provided exogenous variables. These can be found in the following files:

- Elspotprices2nd.csv
- ProdConData.csv

Electricity spot prices are provided in DKK/MWh. Work only with UTC time to avoid any problems with gaps or extra hours due to the change of summer and winter time. You can use the following code to load your data:

```
import os
import pandas as pd

file_P = os.path.join(os.getcwd(),'Elspotprices2nd.csv')
df_prices = pd.read_csv(file_P)
df_prices["HourUTC"] = pd.to_datetime(df_prices["HourUTC"])

file_P = os.path.join(os.getcwd(),'ProdConData.csv')
df_data = pd.read_csv(file_P)
df_data["HourUTC"] = pd.to_datetime(df_data["HourUTC"])
```

Note that there may be NaN values in the ProdConData csv because some of the quantities were not measured from the start of the dataset but started to be recorded at a later stage.

Logistics for the submission of your report:

1. Maximum length of the report: **8 pages, excluding the frontpage**
2. Upload one report per group as a .pdf in Assignment 2 at DTU Learn.
3. Please **mention how you shared the workload** either on the frontpage or at the end. This is necessary by the DTU guidelines.
4. You must also **provide running-code**, i.e., the user should be able to press run in your **Main.py** file and obtain answers to all the following questions. Please use short comments in the code to indicate what each part of the code does.
5. Upload all your code files in a single zip file.
6. You need to upload your report and code on DTU Learn the latest by **Tuesday, April 22, 2025, at 23:59 hrs**.

Expectations and assessment:

- Please use short and clear comments in the code to indicate what each part of the code does. We should be able to follow your thought process and understand what you are doing.
- Do not use any absolute paths. One should be able to run **Main.py** and obtain all answers without any code adjustments.
- Do not use printscreens but actual figures/tables.
- Your figures must be clear and easy to read/interpret.
- Provide short but clear answers to the questions, justified by your data analysis.
- The grade does not depend on the forecasting performance you achieve (i.e., achieving the lowest RMSE). It depends more on the overall thinking and process you followed.
- The first three tasks account for 20% each and the fourth 30% of the grade. Overall quality and presentation of your work accounts for 10%.

Problem description

- Training and Testing Datasets:
  - **Training Dataset: 1/1/19 until 31/8/24**. You can use the whole or only part of the training dataset. Explore different options and see what gives you a good performance. Do not judge by the performance on the testing dataset, because then you simply tweak your model for behaving well on something you are not supposed to know!
  - **Testing Dataset: 1 September 2024 until 30 September 2024 (30 days).**
- **Day-ahead prediction:** You need to make 30 predictions, each containing 24 values. You start (Day 1) with predicting the 24 prices for the 1st of September 2024. Then you know the real values for Day 1, you include them as input and you predict Day 2, i.e., the 2nd of September. Do that for every day until the end of the month (30 x 24 values).
- **Exogenous Variables:** You can use any variables from df_data as exogenous input. Note that in this assignment you will treat the real values of the exogenous variables as the forecasted ones, since we assume we can perfectly forecast those. Besides those variables, you can also use any calendar features or transformations you think help you achieve better performance.

Task 1: Develop an ARIMA model to predict electricity prices. **In both sub-tasks report the RMSE values you achieve with your models: ARIMA, ARIMA + exogenous, persistence.** Be careful to avoid data leakage! The chosen model should be optimized on data other than the testing dataset! Otherwise, you end up fine-tuning your model to perform well specifically on that data!

1.1 Use NO exogenous variables in your model and make day-ahead prediction for your testing dataset. You can use a seasonal ARIMA or FourierFeaturizer and any data transformation you want in your model, but no exogenous features from df_data. Establish a suitable persistence forecast and report the RMSE values in both cases (your model and persistence). Briefly discuss your results.

1.2 Add any exogenous variables you want (maximum 3) and repeat the process (choose/optimize your model and evaluate it for the day-ahead prediction). What exogenous variables helped you improve the prediction and how did you choose the specific ones? Report the RMSE value and compare your results with those from task 1.1 and briefly discuss them.

Task 2: Develop a long short-term memory (LSTM) network to predict the electricity prices. **In both sub-tasks report the RMSE values you achieve with your models: LSTM, LSTM + exogenous, persistence.** Be careful to avoid data leakage! The chosen model should be optimized on data other than the testing dataset! Otherwise, you end up fine-tuning your model to perform well specifically on that data!

2.1 Use NO exogenous variables from df_data in your model and make day-ahead prediction for your testing dataset. Report the RMSE values of your model and persistence. Briefly discuss your results.

2.2 Pick a maximum of 3 exogenous variables and perform the day-ahead prediction for the training dataset. Report the RMSE value you achieve with your model and compare your results with those from task 2.1 and briefly discuss them. What exogenous variables helped you improve the prediction and how did you choose the specific ones?

Task 3: Compare the performance of your ARIMA and LSTM models. What do you observe? In your comparison, please consider the following points:

- How do the two models compare in accuracy when you perform the day-ahead prediction? Do you observe a difference? What are the reasons for the differences in performance you observe in each case? Compare both models performance with the persistence performance as well.
- How much time did it take you to train the models? Is computation time comparable? If not, what are the differences that led to a different computation time?
- How many parameters did you have to tune in each model? Which model did you find easier to train?
- Any other point you consider relevant for the comparison.
- Recommendations: Assume you own an engineering company that develops software solutions for energy forecasting. A small energy trading company comes to you and asks you to develop a tool for them that can predict electricity prices. Considering your assessment, what kind of model would you advise your client to buy/develop for day-ahead predictions and why?

Task 4: Combine the forecasting tool that predicts **for the next 24 hours** you developed with the optimization tool for the optimal bidding strategy for batteries you developed in Assignment 1, and calculate the total profits for the whole month of September 2024.

Consider the following for your battery system.

- The battery's state of charge can vary between 10% and 100%.
- The battery has a 50% SOC at the start of your simulation (on September 1st).

- The battery has a power capacity of 1 MW and an energy capacity of 2 MWh.
- Your charging and discharging efficiency is equal to 95%, i.e., $\eta_\text{c} = \eta_\text{d} = 0.95$.
- Each day you optimize the schedule of the battery between 00:00 and for the next 24 hours.
- You buy and sell energy at the spot price (do not consider any taxes or tariffs).
- You can decide what your state of charge constraint at the end of each daily optimization is. Remember that the final SOC of day N is the starting SOC of day N+1! Mind that on the first day the SOC is set to 50% (initial SOC on September 1) and you should choose the same strategy for the whole period.
- The choice of the final SOC constraint must be justified using your training dataset and not your testing dataset! You should design a strategy based on available data and not tailor your strategy based on the testing dataset!

4.1 Calculate the profits you achieve during the testing dataset by using the bidding model from Assignment 1 and perfect knowledge of prices (you performed this task in Assignment 1). Next, use the forecasting model you developed and has the highest accuracy (ARIMA or LSTM) for day-ahead prediction, and use the forecasted prices instead of the actual ones to bid. Compare the profits you achieve with your forecasting model with the ones you achieve with perfect price knowledge. What do you observe?

Note: To calculate profits when you use forecasted prices, you need to do the following. First, establish a schedule by optimizing your battery based on your **forecasted** prices. Once you have a schedule, use the **actual** prices to calculate your profit from that day.

4.2 Use the persistence model for day-ahead predictions you established in task 1.1 to forecast electricity prices, and use those to calculate your profits from optimizing the battery. Compare the profits you obtain from persistence forecasting with the profits you calculated in task 4.1 (perfect knowledge and forecasts via ARIMA or LSTM). Discuss the results and what is the added value of your forecaster, compared to simply using persistence.

4.3 Can you identify the key characteristics you need to predict well in order to deliver a good profit-maximizing bidding strategy for a given day?

Lessons Learnt: During this assignment, there were definitely several issues that came up while trying to answer the questions or until you got your code running correctly. In no more than half a page, please list 2-3 main points that you think you should remember for the next time you design and code a forecaster and a bidding strategy. Please list at least one issue that had to do with coding, i.e., what should you remember to do in some specific way, or avoid, next time you code for a similar task? And please list at least one main takeaway from the design of the optimal bidding with forecasting and how it affected the results you obtained.