Technical University of Denmark

46765 - Machine Learning for Energy Systems
Assignment 2
Deadline: December 3rd, 2025 (11:59 pm)

Instructions: This assignment evaluates the topics covered in **Lectures 6 - 10** as well as programming and writing skills. It should be carried out in groups. Each group should provide a single submission, including the following:

- A concise project report (maximum 10 pages), detailing the mathematical models developed and presenting and analysing the main results. An appendix can be included (outside of page limit).
- A participation table accompanying the report, detailing the participation of each group member.
- A working and well-documented code in the programming language of your choice. Include also a README in your repository to explain the file structure you have and how to run the code.
- Additional relevant files and data.

This assignment will count towards 33% of the final grade. The assessment will be based on the grading guide (rubric table) provided in the repository. Make sure to look at it when writing the report.

Objectives: In this assignment, you will explore machine learning methods for decision-making in the real-time electricity market. The focus is on controlling a battery energy storage system (BESS) to maximize profit while respecting operational limits. You will implement supervised learning (classification using logistic regression) and reinforcement learning (value or policy iteration).

Assumptions: For simplicity, let us assume instead of day-ahead and intra-day markets, we have a continuous market which is cleared on hourly basis in a continuous manner, and you are able to bid in real-time, i.e. the electricity price of the previous hour is known but not of current and future hours. For the hourly electricity prices, you can utilize the price dataset attached to the assignment and pick your desired price area. Note that similar to other assignments, you can partition the data into a training dataset for the learning process and a testing dataset for evaluation purposes. The battery possesses the ability to take three actions: charge, discharge, or remain inactive during each time period.

Battery characteristics: Let us consider a battery with a total capacity of 400 MWh, featuring charging and discharging capabilities of 100 MW per hour. Consequently, when the battery is at a state of charge (SOC) of 0, it can be fully charged within a span of 4 hours. It is essential to note that at an SOC of 0, the battery cannot discharge, and at an SOC of 400 MWh, it cannot be charged.

Expected Learning Outcomes: You will demonstrate your ability to:

- Solve an optimization problem for operation of a BESS,

- Generate training labels and build a logistic-regression classifier for battery actions,
- Formulate the control task as a Markov Decision Process (MDP),
- Implement value iteration or policy iteration algorithms,
- Evaluate and compare the performance of supervised and reinforcement learning approaches,
- Your critical analysis of the results generated.

Model 1: Logistic Regression

In this model, you will formulate and solve an integer optimization problem to determine the optimal hourly actions of a BESS over a given training horizon. The resulting optimal actions (*charge*, *discharge*, *idle*) will then serve as labels for a logistic regression classifier. Afterwards, you will train a classifier and use it to predict the optimal action for control of a BESS. You have learned about logistic regression **in Lecture 6 and Exercise 7**. The framework of Model 1 is shown in Figure 1.

Figure 1: Classifier architecture for action selection: features in, discrete action out.

Step 1: Optimization-based Action Labels

In this step, you will first determine optimal discrete actions (*charge*, *discharge*, *idle*) over a training horizon by solving an integer optimization problem using historical prices. **In Exercise 2**, you learned how to implement and solve an optimization problem using Gurobipy. The optimal actions serve as labels for a logistic-regression classifier that maps current state information and the electricity price at the previous hour (two input features) to an action via probability thresholds [1].

Consider a time horizon $t = 1, \ldots, T_{\text{train}}$ with hourly electricity prices λ_t (€/MWh). The BESS has a maximum energy capacity $E_{\max} = 400$ MWh and a maximum charge/discharge power $P_{\max} = 100$ MW. The initial SOC is 0 ($E_1 = 0$ MWh).

The optimization problem is defined as follows:

$$\max_{\substack{E_t,\\ a_t^{\text{ch}}, a_t^{\text{di}}, a_t^{\text{id}}}} \quad \sum_{t=1}^{T_{\text{train}}} \lambda_t \, P_{\max} \left(a_t^{\text{di}} - a_t^{\text{ch}}\right) \tag{1}$$

$$\text{s.t.} \quad E_{t+1} = E_t + P_{\max} a_t^{\text{ch}} - P_{\max} a_t^{\text{di}}, \qquad \forall t = 1, \ldots, T_{\text{train}} - 1, \tag{2}$$

$$0 \le E_t \le E_{\max}, \qquad \forall t, \tag{3}$$

$$a_t^{\text{ch}} + a_t^{\text{di}} + a_t^{\text{id}} = 1, \qquad \forall t, \tag{4}$$

$$a_t^{\text{ch}}, a_t^{\text{di}}, a_t^{\text{id}} \in \{0, 1\}, \qquad \forall t, \tag{5}$$

$$E_1 = 0, \tag{6}$$

The objective (1) maximizes the total profit from buying and selling energy. Constraint (2) governs SOC dynamics, while (3) ensures SOC remains within feasible limits. Constraints (4)–(5) enforce that exactly one action is taken at each time step and Constraint (6) defines the initial SOC.

Action labels. From the optimal solution, define per-hour labels:

$$\text{label}_t \;=\; \begin{cases} \text{charge}, & a_t^{\text{ch}} = 1, \\ \text{discharge}, & a_t^{\text{di}} = 1, \\ \text{idle}, & a_t^{\text{id}} = 1. \end{cases}$$

Step 2: Training a Logistic Regression with a Probability Threshold

Construct supervised samples using features $x_t = [\,\text{SOC}_t,\ \lambda_{t-1}\,]$ where $\text{SOC}_t = E_t/E_{\max}$ and the label label_t as defined above. Train a logistic regression classifier to estimate the probability of the *charge* action:

$$P_t(\text{charge} \mid \text{SOC}_t, \lambda_{t-1}).$$

Step 3: Evaluation of the Trained Classifier

Once you have trained the classifier, you can estimate the probability that the charge action is the optimal action and based on that, decide the action $\hat{a}_t$ using a fixed threshold $\hat{P}$:

$$\hat{a}_t \;=\; \begin{cases} \text{charge}, & P_t > \hat{P}, \\ \text{discharge}, & P_t < 1 - \hat{P}, \\ \text{idle}, & 1 - \hat{P} \le P_t \le \hat{P}. \end{cases}$$

where $\hat{P}$ is a hyperparameter that can be tuned by validation (using a validation dataset). Note that you can use the concept of confidence θ and confidence threshold $\hat{\theta}$ as you learned in Exercise 7. Evaluate the classifier on a held-out test set by computing the cumulative profit obtained when applying $\hat{a}_t$ over the test horizon. Consider the initial SOC equal to zero for testing similar to training.

Note 1: If the estimated optimal action $\hat{a}_t$ is not feasible, you must project it back to the feasible set.

Note 2: To tune hyperparameters, you can use the cumulative profit as a metric in the validation process.

Model 2: Reinforcement Learning (Lectures 8-10)

Step 1: Problem Formulation as MDP

Define the problem within the framework of a MDP problem, specifying key components such as the state space, action space, and reward function. Similar to Model 1, consider discrete actions. For simplicity, assume a discrete state space for this step, even if some state variables were initially continuous. Therefore, apply discretization, and then estimate state transition probabilities. Similar to Model 1, consider the electricity price of the previous hour is known but not of current and future hours. The framework of Model 2 is shown in Figure 2.

Step 2: Implementation of Value Iteration or Policy Iteration Algorithms

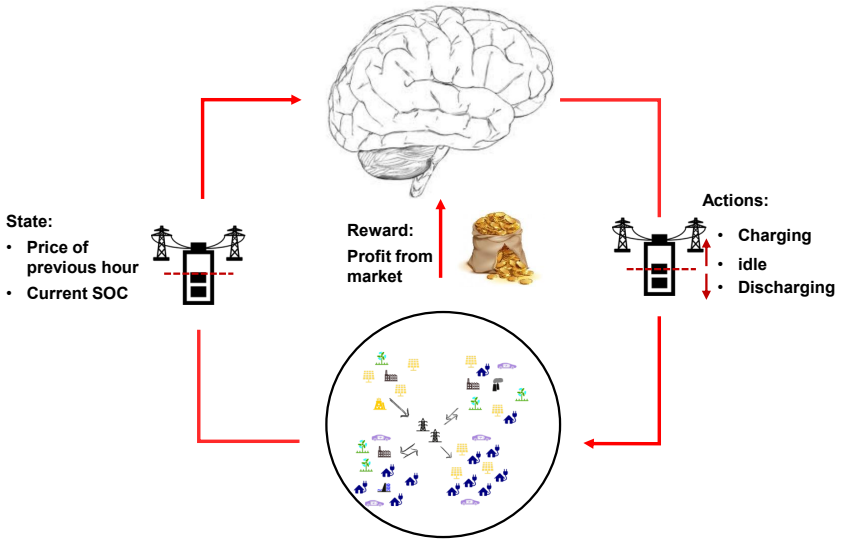

Figure 2: BESS state–action–reward structure used for reinforcement learning

Having acquired a model for the MDP, use the value iteration or policy iteration algorithm to solve the MDP with the obtained transition probabilities. Obtain the optimal policy and its corresponding value obtained from the algorithm.

Step 3: Evaluation of the Obtained Policy

Evaluate the optimal policy you obtained from the previous step on a held-out test set by computing the cumulative profit obtained when applying $\hat{a}_t$ over the test horizon. Consider the initial SOC equal to zero for testing, similar to training (similar to step 3 in Model 2).

Step 4: Comparison of Models 1 and 2

Compare the both models by calculating the ratio of the cumulative profits to the optimal cumulative profit. You can calculate the optimal cumulative profit for the testing set as in Step 1 of Model 1 before.

References

[1] Exercise notes, Exercise 7: https://learn.inside.dtu.dk/d2l/le/lessons/271337/topics/1086504

[2] Chapter V, Lecture notes: https://learn.inside.dtu.dk/d2l/le/lessons/271337/topics/1076448