Assignment 2

The goal of this assignment is to <u>implement DQN and DDQN models</u> using PyTorch, <u>train</u> RL agents on multiple <u>classical environments</u>, and use <u>Weights & Biases</u> to track the performed experiments.

Summary

1. Sign-up for an academic account on Weights & Biases
2. Install the libraries needed
   - Python, Gymnasium, PyTorch, Wandb
3. Test running the wandb "Quickstart" demo with your API key
4. Implement DQN and DDQN Models
5. Create Gymnasium environment run script
6. Train DQN/DDQN in the following <u>classical environments</u>
   - CartPole-v1
   - Acrobot-v1
   - MountainCar-v0
   - Pendulum-v1
7. Run trained agents 100 tests per environment and track the test episode duration
8. Use different <u>hyperparameter values</u> and find best setup
   - Discount Factor
   - Epsilon Decay Rate
   - NN Learning Rate
   - Replay Memory Size
   - Learning Batch Size
9. Use "RecordVideo" Wrapper to record the RL agent in-action
10. Answer the following questions

Questions

For each of the classical environments:
1. What is the difference between DQN and DDQN in terms of training time and performance?
2. How stable are the trained agents? Show with test episode duration figures.
3. What is the effect of each hyperparameter value on the RL training and performance?
4. Explain from your point of view how well suited DQN/DDQN is to solve this problem.

Deliverables

Use this report template to deliver the following requirements
(https://www.overleaf.com/read/dsmwrcyzyyjs#ef6078)

1. GitHub repository with Python codes (Gym Environment, DQN/DDQN model)
2. The recorded video of the trained agent in action
3. The Experiment charts generated by Weights and Biases
4. Report with the outcome summary and answers to the questions asked

Due date

13 Nov 2025

Helping Materials

- https://docs.pytorch.org/tutorials/intermediate/reinforcement_q_learning.html
- https://gymnasium.farama.org/index.html
- https://gymnasium.farama.org/introduction/record_agent/#integration-with-experiment-tracking
- https://wandb.ai/site/experiment-tracking/
- https://wandb.ai/site/articles/what-is-mlops/
- https://docs.wandb.ai/guides/track/