

Introduction to Artificial Intelligence

Lab 6: Reinforcement Learning

Daniel Marczak

Summer 2024

1 General Task Description

Create an implementation of the Q-Learning algorithm to solve a toy Reinforcement Learning problem using the Gymnasium library. Here are the environments to use for each lab variant:

- Variant 1: CartPole. Use “CartPole-v1”. Notice that the observation space is continuous and requires quantization to put it in the Q-Table.
- Variant 2: MountainCar. Use “MountainCar-v0”. Notice that the observation space is continuous and requires quantization to put it in the Q-Table.
- Variant 3: CliffWalking. Use “CliffWalking-v0”
- Variant 4: Taxi. Use “Taxi-v3”
- Variant 5: FrozenLake. Use the arguments “FrozenLake-v1” and `map_name="8x8"` for `gym.make()`

Please refer to the Gymnasium documentation for each environment to understand the problems, goals, actions, states, rewards, and termination conditions.

2 Technical Details

- Write your solution using Python.
- Adhere to clean coding standards and provide comments for readability.
- Provide instructions on how to run your code.
- Do not use Machine Learning or optimization libraries. The gymnasium library along with numpy and a visualization library is enough to do this exercise.

3 Additional Information

References

Introduction to Q-Learning and visualization techniques are available at the following links:

- Hugging Face Course on Q-Learning
- Gymnasium API Wrappers

4 Submission

To complete this exercise, students should submit a solution consisting of:

1. Code:

- Initialization of the Q-table, and function for updating it.
- Training loop: initialize hyperparameters for the Q-Learning algorithm. perform iterative training of taking an action, observing the state, and updating the Q-table.
- Log training metrics (e.g. episode reward) to visualize improvements.

2. Report:

- Experiment with different initialization values and compare the training metrics (e.g. total reward, convergence rate, episode length, etc).
- Report interesting findings from your experiments.

Handing-In Guidelines

You should upload the source code of your solution and the final report to GitHub at least 24 hours before the online assessment. The assessments will be held via Teams during your usual lab hours in the week of 6-10 May (because of the May break, you have 3 weeks for the lab this time).

5 Assessment Criteria

You can earn between 0 and 5 points for the lab. The following criteria will be used to evaluate your work:

- Proper implementation of the training procedure and inference: 1 point.
- Final report, including training plots (see description), observations, and conclusions: 2 points.
- Online assessment: 2 points.

In case of any questions, contact me via MS Teams.