Artificial Intelligence 2020/2021

Exercise Sheet 6: Reinforcement Learning

6.1 Software/Library Installation

Open AI Gym is a toolkit for developing and comparing reinforcement learning (RL) algorithms. The gym library is a collection of test problems with a shared interface — environments — that you can use to work out your RL algorithms.

To work with Open AI Gym, you need to install the corresponding package (e.g., with "*pip install gym*"). You will also need *numpy*. Read further information on how to install *gym*: https://gym.openai.com/docs/

After installing *gym*, please open the example Notebook available at Moodle.

6.2 Frozen Lake Environment

The FrozenLake 4x4 environment is part of the gym library. The agent controls the movement of a character in a grid world. Some tiles of the grid are walkable, and others lead to the agent falling into the water. Additionally, the movement direction of the agent is uncertain and only partially depends on the chosen direction. The agent is rewarded for finding a walkable path to a goal tile.

In this Notebook, we will implement an agent that plays FrozenLake. As mentioned above, the goal of this game is to go from the starting state (S) to the goal state (G) by walking only on frozen tiles (F) and avoid holes (H). However, the ice is slippery, so you will not always move in the direction you intend (in other words, we are in a stochastic environment). For the purpose of this exercise sheet, however, we will consider a deterministic variant of this environment.

a) Unzip the file with the example notebook available at Moodle and open the notebook.

b) Import the dependencies: numpy, OpenAI Gym, and random.

c) Create a deterministic Frozen Lake environment, from the gym library, and render it on the screen.

d) Create a Q-table, considering the action space and the state space for this environment, and initialize its values with zeros.

e) Set hyperparameter variables for the Q-learning algorithm: the total number of episodes to run and the maximum number of steps per episode; the learning rate and the discount factor; and the exploration parameter epsilon, together with its ranges and decay rate.

f) Implement Q-learning and use it to train the agent. Print some information as learning takes place.

g) Print the optimal action for each state, according to the obtained Q-table values.

h) Show the outcome of following a greedy policy.