Exercise Sheet - RL (Model Free)

Exercise 1. Assume we are an agent in a $3x2$ gridworld, as shown in Figure 1. We start at the bottom left state (1) and finish in the top right state (6). When state 6 is reached, we receive a reward of $+10$ and we return to the start for a new episode. Every time we take an action that does not lead to state 6, the reward is -1.

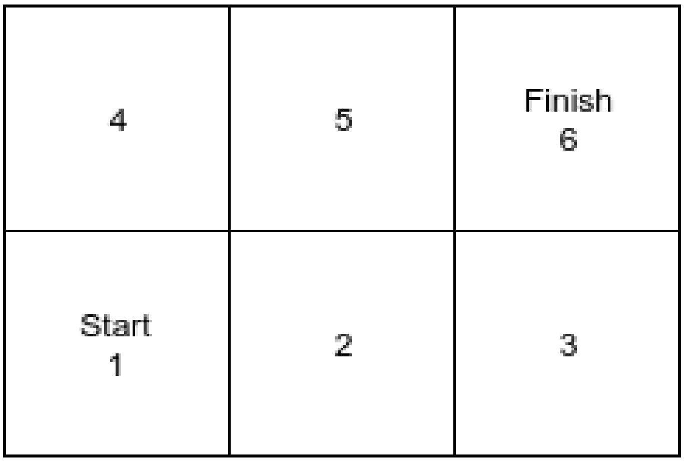

Figure 1: 3x2 gridworld problem.

In each state we have four possible actions: up, down, left and right. For each action we move in the specific direction on the grid. However, there is always a 10% probability that we *slip*, which causes us to actually stay at the same location and not move at all (however, the reward is still -1). Assume that we cannot take actions that bring us outside the grid.

a) Let $P^a_{ss'} = T(s, a, s')$ denote the probability of ending in state s' when taking action a in state s. Give $T(2, right, 3)$, $T(2, right, 2)$ and $T(2, up, 3)$.

Assume our current policy is **random**. We can use Bellman's equation to update the values of each state under the current policy. Initialize all current $V(s)$ to 0. Bellman's equation is given by:

$$V^\pi(s) = \sum_a \pi(s,a) \sum_{s'} P^a_{ss'}[r^a_{ss'} + \gamma V^\pi(s')]$$

b) Take discount parameter $\gamma = 0.5$. Update $V(3)$ once according to Bellman's equation.

John suggests we should not assume a model of the environment. He proposes to use a sampling based approach. In particular, he wants to use Q-learning, which implements the following one step update:

$$Q(s,a) = Q(s,a) + \alpha[r^a_{ss'} + \gamma \max_b Q(s', b) - Q(s,a)]$$

John has already made some steps in this process. He gives you the following table with his current estimates:

Q(1,up)=3	Q(1,down)=.	Q(1,left)=.	Q(1,right)=5
Q(2,up)=5	Q(2,down)=.	Q(2,left)=2	Q(2,right)=6
Q(3,up)=8	Q(3,down)=.	Q(3,left)=3	Q(3,right)=.
Q(4,up)=.	Q(4,down)=2	Q(4,left)=.	Q(4,right)=4
Q(5,up)=.	Q(5,down)=1	Q(5,left)=3	Q(5,right)=7

IMPORTANT!: From now on assume there is no more slipping, i.e. each actions leads deterministically to the next state. So for example, taking action right in state 2 always brings you in state 3.

c) What is the Q-value for state 6, for example: what is $Q(6, down)$?

d) Imagine we start exploitation now, i.e. we take a greedy policy. What policy will the agent follow from the start state. You can indicate the trajectory. Write down the equation you base your greedy choice on.

e) John goes to lunch and asks you to continue his work. He says he stopped in state 4 and uses an ϵ-greedy exploration policy with $\epsilon = 0.20$. He has been drawing random numbers for each step: if the number is smaller than 0.20 he makes an exploring step (excluding the greedy action). Else, he follows the greedy action. The two next numbers are: 0.14 and 0.70. Make the two next updates following Q-learning with $\alpha = 0.1$ and $\gamma = 0.5$. For each step, fill in the form and calculate the update.

s	a	r	s'

$Q(\;,\;) =$

s	a	r	s'

$Q(\;,\;) =$

Solution:

a) 0.9, 0.1, and 0.

b) Since we assumed we cannot take actions that bring us outside the grid, we only have to check for the actions up and left.

For a random policy $\pi(3, up) = 0.5$ and $\pi(3, left) = 0.5$. The tricky aspect is the slipping which we have to take into account.

$$\begin{aligned} V(3) &= 0.5(0.9(-1 + 0.5 \times 0)) + 0.1(-1 + 0.5 \times 0) \\ &+ 0.5(0.9(10 + 0.5 \times 0) + 0.1(-1 + 0.5 \times 0) \\ &= -0.5 + 4.45 = 3.95 \end{aligned}$$

c) State 6 is terminal (no outgoing links), so its value is by definition 0 for each action.

d) Equation for the greedy policy:

$$\pi(s) = \max_a Q(s, a)$$

This results in the trajectory: state 1 - state 2 - state 3 - state 6. Or in the actions: right - right - up. (Both are correct).

Note that we officially write $\pi(s, a)$ for the policy, and it returns a probability distribution of actions (i.e. $\sum_a \pi(s, a) = 1$). However, $\pi(s)$ is shorthand for a deterministic policy, and it returns the action to which it assigns probability 1.

e) First step explore, so take action down from state 4 to state 1. Second step exploit, so take action right from state 1 to state 2. Plugging in from the Q-table into the equation gives:

s	a	r	s'
4	down	-1	1

$Q(4, down) = 2 + 0.1(-1 + 0.5 \times 5 - 2) = 2 - 0.05 = 1.95$

s	a	r	s'
1	right	-1	2

$Q(1, right) = 5 + 0.1(-1 + 0.5 \times 6 - 5) = 4.7$

▲