

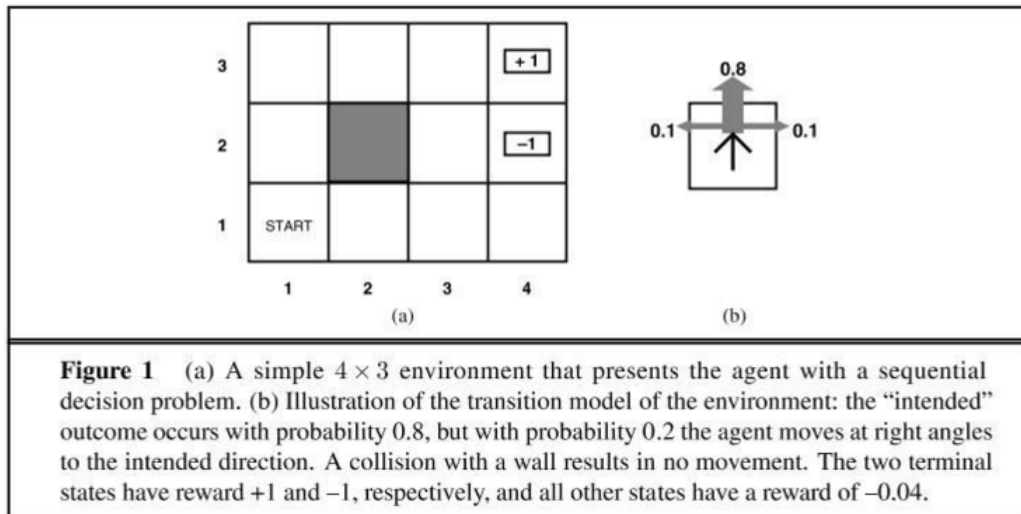
Artificial Intelligence  
 Tutorial 4 on Reinforcement Learning  
 TCS & BIT only

## Introduction

The following multiple choice questions are examples of typical questions one can expect on the AI exam. The questions on the AI exam are also multiple choice, but for this tutorial one has to explain the answers given. Moreover there are also some open questions. After the tutorial the answers to the MC will be available on BB.

## Questions on Reinforcement Learning

1. Consider the following 4x3 environment with reward function as described in the caption.



Assume that the agents applies the Bellman update

$$U_{i+1}(s) \leftarrow R(s) + \gamma \max_a \sum_{s'} P(s' | s, a) U_i(s')$$

with  $\gamma = 1$  and initial values for  $U$  equal to 0 except for (4,2) and (4,3), for which the values are as given: -1 and +1 respectively. What will be the values for  $U$  after:

- (a) One complete update over all states?

- (b) Two complete updates over all states?
- (c) Three complete updates over all states?

Answer: After one complete update the value of  $U(s)$  will be  $-0.04$  expect for the cells from which the agent can reach the goal states in one step, so  $U(s) = -0.04$  for  $s \in \{(1, 1), (1, 2), (1, 3), (2, 1)(2, 3), (3, 1)\}$ . The new value for  $U(3, 3)$  can be computed by taking the max of  $\{0, 0.1, 0.8, 0.1\}$  (the expected  $U$  for the four actions left, up, right,down). Hence the new value for  $U(3, 3)$  equals  $-0.04 + 0.8 = 0.76$ . For  $(3, 2)$  the optimal action is left, bouncing at the wall, because this reduces the probability of ending up in  $(4, 2)$  to 0. Hence  $U(4, 1)=-0.04$ . A similar argument gives that the optimal action for  $U(4, 1)$  is down and new value is equal  $-0.04$

-0.04	-0.04	0.76	+1
-0.04	wall	-0.04	-1
-0.04	-0.04	-0.04	-0.04

After the second sweep all states from which the goal state is not reachable in two steps have utility of  $-0.08$ . The utility for  $(2, 3)$  becomes  $-0.04+0.8*0.76-0.2*0.04 = 0.56$ . The new value for  $U(3, 3)$  is  $-0.04 + 0.8 * 1 + 0.1 * 0.76 + 0.1 * -0.04 = 0.832$ . For  $U(3, 2)$  the utility becomes  $-0.04 + 0.8 * 0.76 + 0.1 * -1 + 0.1 * -0.04 = 0.464$ .  $U(3, 1) = -0.04 + 0.8 * -0.04 + 0.1 * -0.04 + 0.1 * -0.04 = -0.08$  and  $U(4, 1) = -0.04 + 0.8 * -0.04 + 0.1 * -0.04 + 0.1 * -0.04 = -0.08$ .

-0.08	0.56	0.822	+1
-0.08	wall	0.454	-1
-0.08	-0.08	-0.08	-0.08

Values for the third iteration can be computed in a similar way.

2. Consider the RL situation with a state space consisting of  $s$  states and each state the agent can do  $a$  actions. How many policies  $\pi$  are there for this RL problem?
  - (a)  $a^s$
  - (b)  $s^a$
  - (c)  $s \times a$
  - (d)  $s + a$

Answer: (a). In each state the agent can choose among  $a$  actions.

3. Consider the following statements:
  - (i) In order to apply the value iteration algorithm (Bellman update) the agent must know the reward function  $R$  and the successor state (transition) function  $\delta$ .

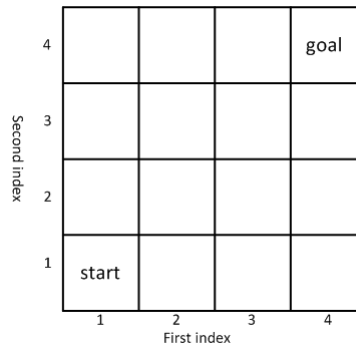
- (ii) In order to apply Q-learning the agent must know the successor state function  $\delta$  but does not need to know the reward function  $R$ .

Which of the following claims is true?

- (a) Only statement (ii) is true
- (b) Only statement (i) is true
- (c) Both statements (i) and (ii) are false.
- (d) Both statements (i) and (ii) are true.

Answer: Only (i) is true.

4. Consider a grid of  $4 \times 4$  with start state in the corner left down and goal state right up.



Assume that the agent can move in four directions *up*, *down*, *left*, *right*. If it bounces to a wall it will stay in the same cell. Moreover the agent cannot leave the goal state, i.e. no action can be executed in the goal state or only a null action (do nothing) with reward 0 can be executed. Assume that every state has a reward of 0 except for the goal state, which has a reward of 10.

Assume that the agent applies the following value iteration algorithm.

- 1 Initialize  $U_{old}$  to 0.
- 2 Repeat
  - 3 For all  $s \in S$ 
    - 4  $U_{new}(s) = \max_a [R(s) + \gamma U_{old}(\delta(s, a))]$
  - 5 End (for all)
  - 6  $U_{old} = U_{new}$
- 7 Until change in  $U_{new}$  is below threshold

Assume that the value for discount factor  $\gamma$  is 0.8. What will after two iterations of the “Repeat” loop (statements [2] – [7]) be the value of  $U_{new}(s)$  for  $s = (3, 3)$ ?

- (a) 0
- (b) 8
- (c) 10
- (d) None of the above

Answer: Initially  $U_{old}(s) = 0$  for all  $s$ . Since  $R(s) = 0$  for all  $s$  except  $R((4, 4)) = 10$ , only  $U_{new}((4, 4))$  will be different from 0 and the value will be  $U_{new}((4, 4)) = 10$ . After the second iteration only  $U_{new}((4, 4)) = 10$  and  $U_{new}(3, 4) = U_{new}(4, 3) = 0.8 * 10 = 8$ , rest zero. After three iterations the value for  $U(3, 3) = 0.8 * 8 = 6.4$ .

5. Consider the following grid world and the given learned values for utility function. The optimal policy (given this utility function) for the top row middle square is moving right (see arrow). This action has a probability of 0.80 of success, and a probability of 0.15 moving down and 0.05 probability of moving left.

0.73	→	0.89
	0.83	

What is the utility value of the top middle square given that a move costs 0.1? Discount factor is 1.

Answer: the utility value for the top middle square is:

$$U = -0.1 + 0.8 * 0.89 + 0.15 * 0.83 + 0.05 * 0.73 = 0.773$$

6. Consider an agent which lives in a deterministic environment and uses it's learned state value function  $U$  for it's policy to select the next action. Assume that an agent is in state  $s_0$  with value  $U(s_0) = 3.0$ . The possible actions, next states, corresponding reward and value function are given by:

action $a$	state $\delta(s, a)$	reward $R(s)$	$U(\delta(s, a))$
$a_1$	$s_1$	-1	5.0
$a_2$	$s_2$	-1	4.0
$a_3$	$s_3$	-1	2.0

Assume that the discount factor is 0.8, what will be the action executed by the agent and what will be the new value  $U(s_0)$  if the agent applies Temporal Difference Learning with learning rate 0.5?

- (a) action  $a_1$  and new  $U(s_0) = 4.0$
- (b) action  $a_2$  and new  $U(s_0) = 2.6$
- (c) action  $a_3$  and new  $U(s_0) = 3.3$

(d) action  $a_1$  and new  $U(s_0) = 3.0$

Answer: The optimal action is the action  $a$  for which  $U(\delta(s, a))$  is maximal. In this case  $a_1$ . The new value for  $U(s_0)$  will be  $U(s_0) + \alpha(R(s_0) + \gamma U(\delta(s_0, a_1)) - U(s_0)) = 3.0 + 0.5(-1 + 0.8 * 5 - 3) = 3.0$ .

