Reinforcement Learning Practical, Assignment 2

Matthia Sabatelli, Nicole Orzan

December 6, 2021

1 Policy Evaluation (2 points)

After years and years of Reinforcement Learning research, Matthia has finally managed to train a robot to bring him coffee. The robot's policy is still far from optimal, but he feels like it is a good start as it allows him to avoid going back and forth his office and the Albert Heijn each time he feels sleepy. However, he is not entirely satisfied with the robot's policy and therefore wishes to improve it. He recalls from back in the days when he was a student that before improving an agent's policy, it is usually good practice to evaluate it. He, therefore, draws the following Markov Decision Process (see Fig. 1) representing the states that are being visited by the robot together with the transition probabilities and rewards that are obtained when it moves from a certain state to another. Assuming that the robot starts in state ①, and that the terminal state is given by state ⑤, what are the values of each state? Note that before learning $V(s) = 0$ for all $s \in \mathcal{S}$ and that throughout learning $\gamma = 0.5$.

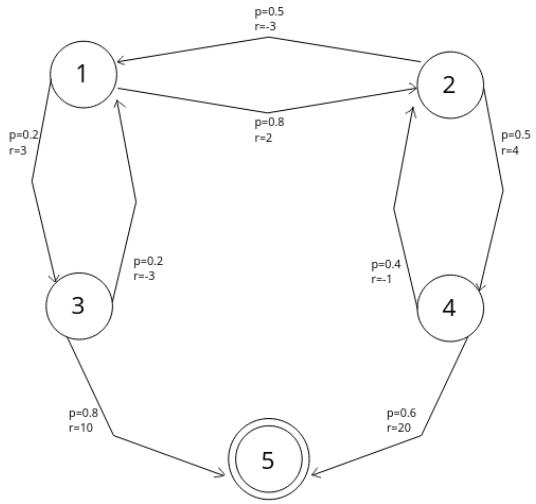

Figure 1: The Markov Decision Process representing the policy that Matthia's robot is following to get him coffee.

2 Temporal Difference Learning (4 points)

The team of the Hanzehogeschool has challenged the soccer team of the University of Groningen in playing a soccer match. As Matthia and Nicole are both Italian and given that Italy managed to win the previous European Cup, the university's dean had the wonderful idea of appointing them as trainers of the team since he was convinced both of them were soccer experts. However, this is nothing further from the truth, and after the first half, the team of the Hanzehogeschool is already leading $1 - 0$.

Matthia and Nicole, therefore, decide to use all of their experience in Reinforcement Learning to prevent the team of the University of Groningen from experiencing such a shameful loss. They start with modeling the ongoing game as a Markov Decision Process with two different states: the "Play" state and the terminal "Goal" state [1]. The reward signal that is returned after each $\langle s, a, s_{t+1} \rangle$ transition is summarized by Table 1.

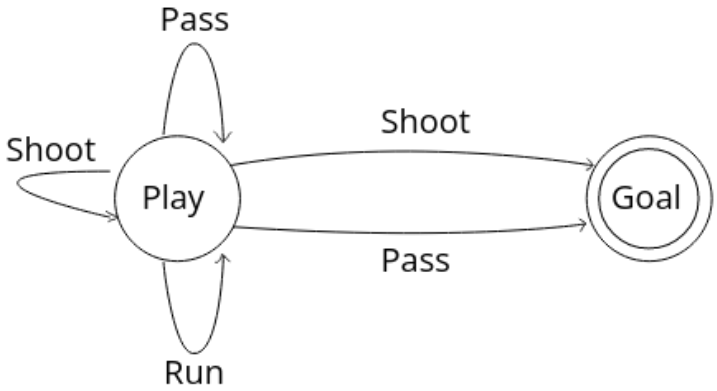

Figure 2: The Markov Decision Process representing the soccer game between the University of Groningen and the Hanzehogeschool.

s	a	s_{t+1}	$\Re(s, a, s_{t+1})$
Play	Run	Play	2
Play	Pass	Play	3
Play	Pass	Goal	8
Play	Shoot	Play	0
Play	Shoot	Goal	20

Table 1: The rewards returned by $\Re$ after taking action a in state s and moving to s_{t+1}.

1. Matthia and Nicole are really interested in knowing the value of the "play" state, as they think it is one of the most important states of the Markov Decision Process represented in Fig. 1. However, they are so focused on the game that they cannot spend any time doing calculations. Fortunately, they have Henry which in tape review has the chance of observing the following trajectories: ⟨play,run,play⟩ and ⟨play,shoot,goal⟩. Use them in order to update the value of the "play" state by using temporal difference learning. The learning rate $\alpha = 0.5$, the discount factor $\gamma = 0.9$ and $V(s) = 0$ for all $s \in \mathcal{S}$.

2. Despite the previous calculations, the RUG team is still losing the game. Nicole thinks that the reason for this is that she and Matthia have only focused on learning the state-value function $V^{\pi}(s)$, and not the state-action value function $Q^{\pi}(s, a)$ which she remembers to be much more informative. Therefore she suggests to use Q-Learning to improve the team's strategy based on the following three episodes: ⟨play,run,play⟩, ⟨play,pass,play⟩ and ⟨play,pass,goal⟩. Help her and Henry with the calculations. The learning rate $\alpha = 0.5$, the discount factor $\gamma = 0.9$ and $Q(s, a) = 0$ for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$.

[1] Note that differently from the previous exercise, in this case, while the connectivity across states is known, the transition probabilities are not.

3. Thanks to the previous calculations, the RUG team scored the equalizer and therefore managed to level the score. However, Matthia and Nicole really want to win this game, so they decide to surprise the team of the Hanzehogeschool by improving the RUG's Q-function through the use of an *on-policy* learning algorithm. However, they do not remember a single update rule and therefore ask Henry for help. Join forces with Henry and improve the team's policy by using an *on-policy* learning algorithm of your choice. Based on Table 1 create two new episodes suitable for an *on-policy* learning algorithm. The learning rate $\alpha = 0.5$ and the discount factor $\gamma = 0.9$. *Hint*: there might be something missing in the Table 1 ...

3 Function Approximators (4 points)

The strategy of improving the team's policy first through Q-Learning and then through *on-policy* learning paid off! In fact, the team of the University of Groningen managed to win the game $1-2$. The dean is so happy that he decided to give Matthia and Nicole a permanent contract, however, not as Reinforcement Lecturers but rather as official soccer trainers. Their goal is to prepare the university's team for next year's national university league and to possibly win it. While they are happy with how they managed to defeat the team of the Hanzehogeschool, they also know that in order to succeed at the national level, they will have to come up with a smarter strategy. At the first training camp, Nicole proposes to use Q-Learning in combination with a function approximator.

During training she decides to re-watch the game against the Hanzehogeschool and re-observes the following trajectory: $\langle play, run, play \rangle$ and now she wants to learn an approximation of the state-action value function with a linear model $Q^\pi(s,a) \approx \hat{Q}^\pi(s,a;\mathbf{w})$. The model comes in the following form:

$$\hat{Q}^\pi(s,a;\mathbf{w}) = w_1 + w_2 f(s,a),$$

where

$$f(s,a) = \begin{cases} 1 & \text{a} = play \\ -1 & \text{a} = run \end{cases} .$$

Starting from initial weights $w_1 = 0$ and $w_2 = 0$ update both weights after having observed the aforementioned trajectory. $\alpha = 0.5$ and $\gamma = 1$.