ROB311 - TD2 - Apprentissage pour la Robotique

Araujo Belén, Victor
De Carvalho Ferreira Sula, Julia

September 2019

1 Reinforcement Learning- Markov Decision Process

The reinforcement Learning, in short, is an algorithm aims to chose a action to maximize the reward in a particular scenario. In figure 1, it's a simple diagram explaining the main influences in the algorithm, which perceiving from the environment a state and a reward, chooses an action as to maximise its rewards.

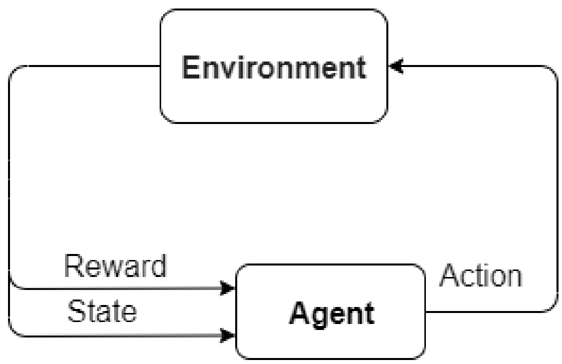

Figure 1: Reinforcement Learning Diagram

For example, there could be four states and three action as describe by figure 2. Each state would have a respective reward, the reward for getting to S_3 would be 10, to S_2 , 1 and for the rest of the state zero.

Then,a transition matrix would have to be defined , as to shown , the probability of getting to a state to another considering a particular action.

The transition matrix considered in this case are:

$$T(S, a_0, S*) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1-x & 0 & x \\ 1-y & 0 & 0 & y \\ 1 & 0 & 0 & 0 \end{bmatrix} \tag{1}$$

$$T(S, a_1, S*) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{2}$$

$$T(S, a_2, S*) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{3}$$

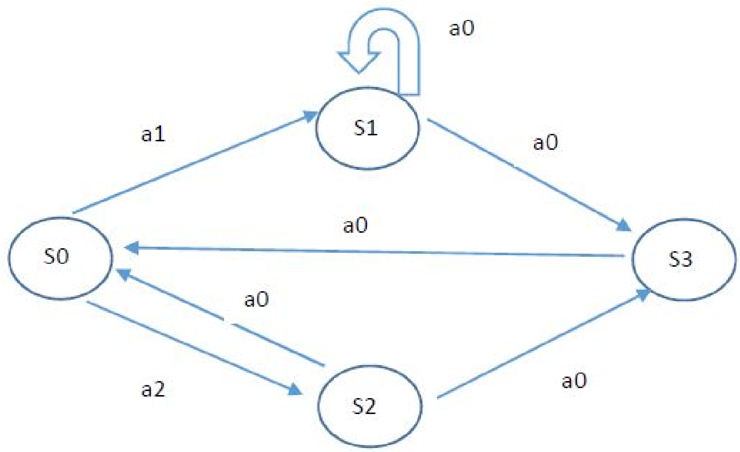

Figure 2: Transition diagram

These information allows us to apply the reinforcement learning- MDP to found out the best course of action , that is the best policy, as it will be develop through this TP.

1.1 Enumerate all the possible policies

Policy can be defined as a possible system‘s strategy for a given environment. There are four states: $(S0, S2, S1, S2)$, and three possible actions per each one: (a_0, a_1, a_2). However, as shown in the figure 2, every state can't perform all the actions, and in consequence, there are 5 policies listed in the table 1.1 and 7 paths which are listed in the table 1.1:

State	action
S_0	a_1
	a_2
S_1	a_0
S_2	a_0
S_3	a_0

Table 1: Possible policies

State	action	*state
S_0	a_1	S_1
S_0	a_2	S_2
S_1	a_0	S_1
S_1	a_0	S_3
S_2	a_0	S_0
S_2	a_0	S_3
S_3	a_0	S_0

Table 2: Possible paths

1.2 Write the equation for each optimal value function for each state

Resolving the equation 4 for each one of the for states, that is $V^*(s_0), V^*(s_1), V^*(s_2), v^*(s_3)$. The following utility equations are found. It is clear that for the states 1-3 the optimum policy will always be *action*0, in the other hand for the state 0, the optimum policy is not clear.

The optimum policy a_O for states 1-3 is easily found due to the fact that it is the only action with a probability not null and because there are no negative rewards or a negative initialization of the utilities values. Actually, all utilities will always be positives, in this case, as x and y are < 1,$V*(S)_o = 0$ and $\gamma \geq 0$.

$$V^*(S) = R(S) + max_a \gamma \sum_{S'} T(S, a, S')V^*(S') \tag{4}$$

1.2.1 State S_0

$$\begin{aligned} V^*(S_0) &= R(S_0) + max\gamma \left[\sum T(S_0, a_1, S_1)V^*(S_1), \sum T(S_0, a_2, S_2)V^*(S_2)\right] \\ &= 0 + max\ \gamma\,(V^*(S_1), V^*(S_2)) \end{aligned} \tag{5}$$

1.2.2 State S_1

$$V^*(S_1) = 0 + \gamma((1-x)V^*(S_1) + xV^*(S_3)) \tag{6}$$

1.2.3 State S_2

$$V^*(S_2) = 1 + \gamma\,((1-y)V^*(S_0) + yV^*(S_3)) \tag{7}$$

1.2.4 State S_3

$$V^*(S_3) = 10 + \gamma V^*(S_0) \tag{8}$$

1.3 Is there exist a value for x, that for all $\gamma \in [0, 1)$ and $y \in [0, 1], \pi^*(S_0) = a_2$. Justify your answer

$$\pi^*(S_0) = argmax_a \gamma \sum_{S'} T(S, a, S')V^*(S') \tag{9}$$

From (4), the optimal policy of $\pi^*(S_0)$ can be written as following:

$$\pi^*(S_0) = argmax\gamma\,[V^*(S_1), V^*(S_2)] \tag{10}$$

For a_2 to be the optimum policy $\pi^*(S_0)$=a_2, the inequality defined by 11 must be true and γ must not be 0.

$$V^*(S_2) > V^*(S_1) \tag{11}$$

Considering (5),(6),(7) and (8), there is a dependency on the value of γ, therefore, the inequality will be evaluated in the possible range of γ.

1.3.1 $\gamma = 0$

If $\gamma = 0$, both (7) and (6) are zero. Therefore we cannot guarantee which action will be taken.

1.3.2 $0 \leq \gamma < 1$

Now, if γ varies between 0 and <1, x might have an influence in the terms of (6) and (7).

Considering simply the case $x = 0$, fulfills the condition: a_2 is optimal is possible, as (6) becomes:

$$\begin{aligned} V^*(S_1) &= 0 + \gamma(1-0)V^*(S_1) + 0\gamma V^*(S_3) \\ V^*(S_1) &= 0 + \gamma V^*(S_1) \end{aligned} \tag{12}$$

And developing (7), it is clear that this equation is at least bigger or equal to one, as the utility values are positives.

$$\begin{aligned} &V^*(S_2) = 1 + \gamma\left((1-y)V^*(S_0) + yV^*(S_3)\right) \\ &V^*(S_2) = 1 + \alpha \\ &Where \\ &\qquad \alpha = \gamma\left((1-y)V^*(S_0) + yV^*(S_3)\right) \\ &\ And\ \alpha \geq 0 \end{aligned} \tag{13}$$

Therefore, the equation (11) is also verified when $0 \leq \gamma < 1$ if x is (0), consequently, there is a value of x that for all $y \in [0,1]$ and all $\gamma \in [0,1]$, a_2 is the optimal policy.

1.4 Is there exist a value for y, that all $x > 0$, and $y \in [0,1], \pi^*(S_0) = a_1$. Justify your answer

If $\pi^*(S_0) = a_1$, then :

$$V^*(S_1) > V^*(S_2) \tag{14}$$

However for this case , it's easier to analyse directly the equation considering all γ range, as it's known that for $\gamma = 0$ no optimum solution can be found.

1.4.1 $0 < \gamma < 1$

Firstly, as stated above from (7), $V^*(S_2) \geq 1$, hence to comply with the inequality $V^*(S_1) > V^*(S_2)$, $V^*(S_1) > V^*(S_1) \geq 1$.

Developing (6) $V^*(S_1)$ is:

$$\begin{aligned} 0 + \gamma(1-x)V^*(S_1) + x\gamma V^*(S_3) &= V^*(S_1) \\ V^*(S_1) - \gamma(1-x)V^*(S_1) &= x\gamma V^*(S_3) \\ V^*(S_1)(1-\gamma(1-x)) &= x\gamma V^*(S_3) \\ V^*(S_1) &= \frac{x\gamma V^*(S_3)}{1-\gamma(1-x)} \end{aligned} \tag{15}$$

Therefore, to comply with 14:

$$V^*(S_1) = \frac{x\gamma V^*(S_3)}{1-\gamma(1-x)} \geq 1 \tag{16}$$

Nevertheless, y has no influence in the outcome of this inequality, consequently, there is no possible value of y that will make a_1 the optimum policy.

1.5 Using $x = y = 0.25$ and $\gamma = 0.9$, calculate the π^* and V^* for all states. Implement value iteration

Algorithm 1: OPTIMUM VALUE FUNCTION

n^0 states,γ,$V(s)$, $T(S,a,S')$, R
begin
  for *error* $> \epsilon$ **do**
    for *every state* **do**
      for *every action* **do**
        $\pi^*(s) \leftarrow \gamma \sum T(S,a,S')V(s)V(s)^*$ $V(s) \leftarrow R + \gamma \sum T(S,a,S')V(s)V(s)^*$
      end
      $\pi^*(s) = argmax(\pi^*(s))$
      $V(s)^* = maxV(s)^*$
    end
  end
end
return $\pi^*(s), V(s)^*$

The algorithm implemented to calculate the policy and the utility is as above.The optimum policy obtained is $\pi^*(s_0), \pi^*(s_1), \pi^*(s_2), \pi^*(s_3) = [a_1, a_0, a_0, a_0]$.

Conclusion

The reinforcement learning -MDP is really useful if there are not any good models to represent the environment , for example if there are not label database, as it does not depend on previously known information.

Therefore, it is a sequential algorithm, for each input there is a output , which represent a new iteration, unlike other supervised learning algorithm where there is a only one input-output iteration.

Nevertheless, reinforcement algorithm can be very high cost in large and complex environment as there are many states and possible actions, hence, there is a need to explore the scenario before exploiting it and converging to the optimum solution.