Chapter 3

Thursday, January 3, 2019 1:15 PM

Finite Markov Decision Processes (Finite MDP):
The Finite MDP problem involves both evaluative feedback and also an associative aspect (choosing different actions in different situations). They are a classical formalization of sequential decision making, where actions influence not only the immediate rewards but also subsequent situations, or states and through those future rewards also. So they involve delayed reward and the need to tradeoff between immediate and delayed reward.
In the bandit problem only the value of each action ($q_*(a)$) is estimated but in MDPs, the value $q_*(s,a)$ of each action is each state 's' has to be estimated or the value $v_*(s)$ of each state given optimal actions selection have to be estimated. These state dependent quantities are essential to accurately assign credit for long-term consequences to individual action selections.

3.1. The Agent-Environment Interface

MDPs are a straight forward way to frame the problem of learning from interaction to achieve a goal.

Agent:
The learner and decision maker.

Environment:
The thing the agent interacts with, it comprises everything outside the agent.

Interactions:
The agent and environment interacts continually. The agent selects actions and the environment responds to those actions and present new situations to the agent. The environment also provides the rewards, which are nothing but the numerical values that the agent has to maximize over time through its choice of actions.

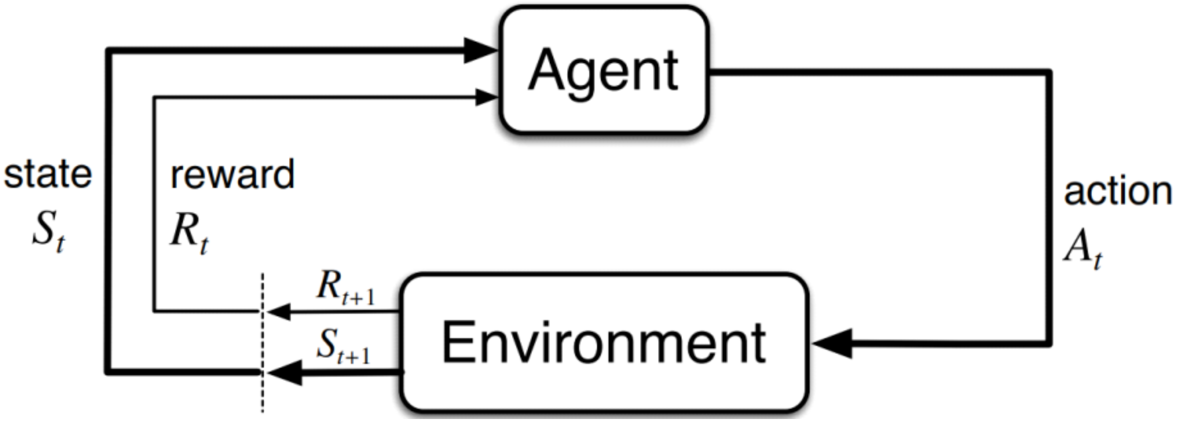

Figure Description: The agent-environment interaction in a Markov Decision Process

The agent and environment interact at each of a sequence of discrete time steps, t = 0, 1, 2, 3, ... At each time step t, the agent receives some representation of the environment's state, $S_t \in S$, and on that basis selects an action, $A_t \in A(s)$. One time step later, in part as a consequence of its action, the agent receives a numerical reward, $R_{t+1} \in R$, and finds itself in a new state, S_{t+1}.
The MDP and agent together thereby give rise to a sequence or *trajectory* that begins like this:

$$S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, R_3, \ldots$$

In finite MDP, the sets of states, actions and rewards (S, A, and R) have a finite number of elements. The random variables R_t and S_t have well defined discrete probability distributions which depend only on the previous state and the action. For particular values of the random variables, s' ∈ S and r ∈ R, there is a probability of those values occurring at time t, given particular values of the preceding state and action:

$$p(s', r \mid s, a) \doteq \Pr\{S_t = s', R_t = r \mid S_{t-1} = s, A_{t-1} = a\}$$

This function 'p' defines the dynamics of MDP. 'p' specifies a probability distribution for each choice of s and a, that is, that

$$\sum_{s' \in \mathcal{S}} \sum_{r \in \mathcal{R}} p(s', r \mid s, a) = 1, \text{ for all } s \in \mathcal{S}, a \in \mathcal{A}(s)$$

The probabilities given by 'p' completely characterize the environment's dynamics. So the probability of each possible value for S_t and R_t depend only on the immediate preceding state and action, S_{t-1} and A_{t-1} not on the earlier states and actions.

This is a restriction to the states and this restriction implies that state must include information about all aspects of the past agent-environment interaction that make a difference for the future. It is called as the "Markov Property".

The four argument dynamics function 'p' can be used to compute anything about the environment.
The state transition probabilities can be found out using the following relation:

$$p(s' \mid s, a) \doteq \Pr\{S_t = s' \mid S_{t-1} = s, A_{t-1} = a\} = \sum_{r \in \mathcal{R}} p(s', r \mid s, a)$$

which becomes a three argument function.

$$p : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \rightarrow [0, 1])$$

The expected rewards can also be computed for the state action pairs using the following equation and this is a two argument function.

$$r(s, a) \doteq \mathbb{E}[R_t \mid S_{t-1} = s, A_{t-1} = a] = \sum_{r \in \mathcal{R}} r \sum_{s' \in \mathcal{S}} p(s', r \mid s, a)$$

The two - argument function:

$$r : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$$

The expected rewards for the state-action-next-state triples as a three-argument function.

$$r(s, a, s') \doteq \mathbb{E}[R_t \mid S_{t-1} = s, A_{t-1} = a, S_t = s'] = \sum_{r \in \mathcal{R}} r \frac{p(s', r \mid s, a)}{p(s' \mid s, a)}$$

The three argument function:

$$r : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$$

The MDP framework is abstract and flexible. They can be applied to different problems in different ways.

Example Considerations:

Example 1: (Time steps)

The time steps need not to refer to the fixed intervals of real time. They can refer to arbitrary successive stages of decision making and acting.

Example 2: (Actions)

The actions can be low-level controls, such as voltages applied to motors of robot arm or high-level decisions such as whether or not to have lunch or to go to graduate school.

Example 3: (States)

States can be completely determined by low-level sensations, such as direct sensor readings, or they can be more high-level and abstract, such as symbolic description of objects in a room. Some of what makes up a state can be based on memory of past sensations or even be entirely mental or subjective.

Broader Considerations:

An agent can be in the state of not being sure where an object is or of having just been surprised in some clearly defined sense.

Similarly some actions might be totally mental or computational. Some actions might control what an agent chooses to think about or where it focuses its attention.

In general, actions can be any decisions that have to be learned how to make, and the states can be anything to know which might be useful in making them.

Boundary between the agent and the environment:

Boundary between the agent and the environment is not necessarily to be same as the physical boundary of a robot's or animal's body.

For example, the motors and mechanical linkages of a robot and its sensing hardware should be considered as part of the environment rather than parts of the agent. If the MDP is applied to a human or animal muscles, skeleton, and sensory organs should be considered part of the environment.

Rewards may be calculated inside the physical bodies of natural and artificial learning systems, but should be considered external to the agent.

So the general rule is, anything that cannot be changes arbitrarily by the agent is considered as outside of it and belongs to the part of its environment.

It is not necessary to consider everything in the environment is unknown to the agent. The agent often knows quite a bit about how the rewards are computed as a function of its actions and the states in which they are taken. But the reward computation is considered as external to the agent, because it defines the task facing the agent and beyond the agent's ability to change arbitrarily.

For example, in a complicated robot, many different agents may be operating at once, each with its own boundary. One agent may take high-level decisions which form part of the states faced by a lower-agent that implements the high-level decisions.

Agent - Environment boundary is selected once particular states, actions and rewards were selected and also specific decision making task of interest is identified.

The MDP framework suggests that any problem of goal-directed learning can be reduced to three signals passing back and forth between an agent and its environment:

1. One signal to represent the choices made by the agent (actions)
2. One signal to represent the basis on which choices are made (state)
3. One signal to define the agent's goal (reward)

The particular states and actions vary greatly from task to task and how they are represented can strongly affect the performance. Such representational choices are more art than science.