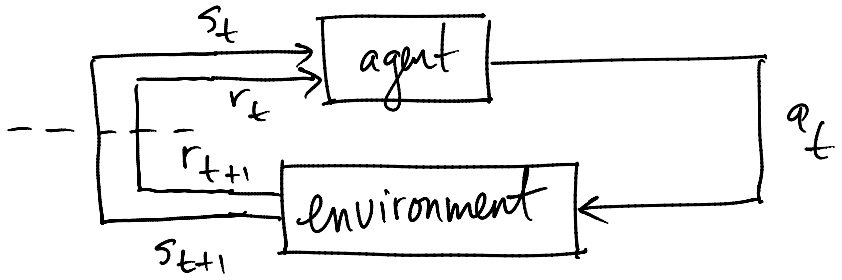
MARKOV DECISION PROCESSES

Definition

Recall the RL setup:



A Markov decision process satisfies

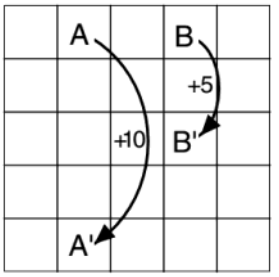
$$\Pr\{s_{t+1} = s', r_{t+1} = r \mid s_t, a_t, \ldots, s_0, a_0\}$$

$$= \Pr\{s_{t+1} = s', r_{t+1} = r \mid s_t, a_t\}$$

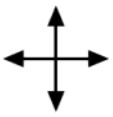
That is, the state and reward at time $t+1$ depend only on the state and action at time t, and not on the more distant past. Many RL problems can be cast (at least approximately) as MDPs by defining the state space appropriately.

Example Gridworld

From Sutton and Barto:



(a)



3.3	8.8	4.4	5.3	1.5
1.5	3.0	2.3	1.9	0.5
0.1	0.7	0.7	0.4	-0.4
-1.0	-0.4	-0.4	-0.6	-1.2
-1.9	-1.3	-1.2	-1.4	-2.0

(b)

$\gamma = 0.9$

Figure 3.5 Grid example: (a) exceptional reward dynamics; (b) state-value function for the equiprobable random policy.

The agent has four available actions. An action that would take the agent of the board keeps the agent in the same square and leads to a reward of -1. All other moves have a reward of 0 except for moves onto either of the two special states, whose rewards are shown.

This is an MDP, with state space

$$S = \{1, 2, \ldots, 25\}$$

and action space

$$A = \{N, S, E, W\}.$$

By definition of conditional probability,

$$\Pr\{s_{t+1}=s',\ r_{t+1}=r \mid s_t=s,\ a_t=a\}$$

$$=\Pr\{s_{t+1}=s' \mid s_t=s,\ a_t=a\}$$

$$\times\ \Pr\{r_{t+1}=r \mid s_{t+1}=s',\ s_t=s,\ a_t=a\}.$$

Therefore it is common to summarize an MDP by the

transition probabilities

$$P^a_{ss'} = \Pr\{s_{t+1}=s' \mid s_t=s,\ a_t=a\}$$

and

$$R^a_{ss'} = \mathbb{E}\{r_{t+1} \mid s_t=s,\ a_t=a,\ s_{t+1}=s'\}.$$

The Bellman Equations

Let $\pi(s,a)$ be a policy. Recall

$$V^\pi(s) = \mathbb{E}_\pi\{R_t \mid s_t=s\}$$

$$Q^\pi(s,a) = \mathbb{E}_\pi\{R_t \mid s_t=s,\ a_t=a\}$$

the state value function and state-action value function.

Unless otherwise stated, we will assume a finite MDP, i.e., $|S| < \infty$, $|A| < \infty$, S = state space, A = action space.

Observe

$$V^\pi(s) = \mathbb{E}\{R_t \mid s_t = s\}$$

$$= \sum_a \pi(s,a)\, \underbrace{\mathbb{E}_\pi\{R_t \mid s_t = s,\ a_t = a\}}_{\text{Note: this is just } Q^\pi(s,a)}$$

$$= \sum_a \pi(s,a) \sum_{s'} P^a_{ss'}\, \mathbb{E}_\pi\{R_t \mid s_t = s,\ a_t = a,\ s_{t+1} = s'\}$$

$$= \sum_a \pi(s,a) \sum_{s'} P^a_{ss'}\, \mathbb{E}_\pi\{r_{t+1} + \gamma R_{t+1} \mid s_t = s,\ a_t = a,\ s_{t+1} = s'\}$$

[because $R_t = \sum_{k \geq 0} \gamma^k r_{t+k+1} = r_{t+1} + \gamma R_{t+1}$]

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

This is a linear system of equations for V^π. Therefore, given any policy π and knowledge of the MDP dynamics $(P^a_{ss'}, R^a_{ss'})$, we can determine the state value function

by solving a linear system of equations. It can be shown that there is a unique V^π solving the above equations.

Similarly, for the state-action value function,

$$Q^\pi(s,a) = \mathbb{E}_\pi\{ R_t \mid s_t = s,\ a_t = s \}$$

$$= \sum_{s'} P^a_{ss'} \mathbb{E}_\pi\{ r_{t+1} + \gamma R_{t+1} \mid s_t = s,\ a_t = a,\ s_{t+1} = s' \}$$

$$= \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma V^\pi(s') \right]$$

$$= \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma \sum_a \pi(s',a)\, Q(s',a) \right]$$

Unfortunately, for many problems, the state space S is too large to solve the Bellman equations directly. E.g., for backgammon, $|S| \approx 10^{20}$. Nonetheless, these equations will later form the basis of efficient algorithms.

Bellman Optimality Equations

We say $\pi \geq \pi'$ iff $V^\pi(s) \geq V^{\pi'}(s)\ \forall s$.

We say π^* is optimal iff $\pi^* \geq \pi\ \forall \pi$.

We say π^* is optimal iff $\pi^* \geq \pi \quad \forall \pi$.

It can be shown that there is always an optimal policy, although it may not be unique. Even if there are multiple optimal policies, they all have the same

optimal state value function

$$V^*(s) = \max_\pi V^\pi(s), \qquad s \in \mathcal{S}$$

and optimal state-action value function

$$Q^*(s,a) = \max_\pi Q^\pi(s,a) \qquad s \in \mathcal{S}, \; a \in \mathcal{A}.$$

These satisfy the Bellman optimality equations:

$$V^*(s) = V^{\pi^*}(s)$$

$$= \max_a Q^{\pi^*}(s,a)$$

as seen previously $\longrightarrow$

$$= \max_a \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma V^{\pi^*}(s') \right]$$

$$= \max_a \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma V^*(s') \right]$$

and

$$Q^*(s,a) = \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma V^*(s') \right]$$

$$Q^*(s,a) = \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma V^*(s') \right]$$

↖ from previous equations

$$= \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma \max_{a'} Q^*(s',a') \right].$$

Note that these are nonlinear equations. Also notice that they do not involve π^*.

Once V^* or Q^* is known, an optimal policy can be determined. If Q^* is known,

$$\pi^*(s) = \text{any solution of } \arg\max_a Q^*(s,a).$$

If V^* is known, then

$$\pi^*(s) = \text{any solution of } \arg\max_a \sum_{s'} P^a_{ss'} \left[R^a_{ss'} + \gamma V^*(s') \right].$$

Many RL algorithms can be understood as (approximately) solving the Bellman optimality equations.