Reinforcement Learning: An Introduction

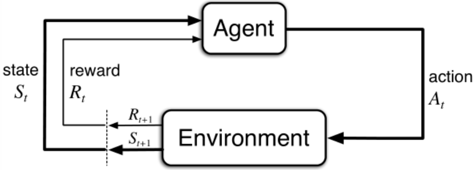
Chapter 3 - Finite Markov Decision Processes

Reporter: Chenyu Xue

July 28, 2020

Outline

1. Markov Processes
2. Markov Reward Processes
3. Markov Decision Processes
4. Summary



- Agent and Environment interact at discrete time steps: $t = 0, 1, 2, ...$
- Then they together give rise to a trajectory like this: $S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, R_3, ...$
- Generally, MDP can be described formally with 4 componets: $\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}$.
- In a finite MDP, $\mathcal{S}, \mathcal{A}, \mathcal{R}$ all have a finite number of elements.

Markov Property

- The history of the states: $H_t = \{S_1, S_2, ..., S_t\}$.

Definition

A State S_t is Markov if and only if:

$$\mathbb{P}(S_{t+1}|S_t) = \mathbb{P}(S_{t+1}|H_t)$$

- The current state already captures the information of the past states.
- "The future is independent of the past given the present."

State Transition Probability

Definition

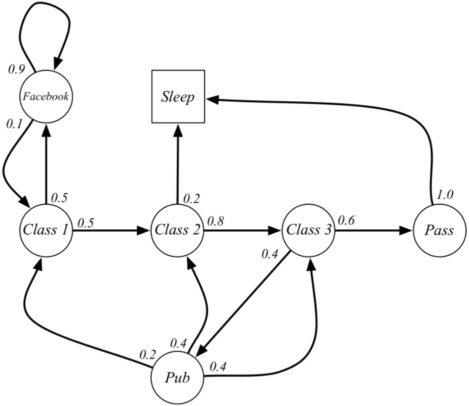
For a Markov state s and successor state s', the state transition probability is given by:

$$\mathcal{P}_{ss'} = \mathbb{P}\left(S_{t+1} = s' | S_t = s\right)$$

- Then we can formulate the state transition probability into a state transition matrix $\mathcal{P}$:

$$\mathcal{P} = \begin{pmatrix} \mathbb{P}\left(s_1|s_1\right) & \mathbb{P}\left(s_2|s_1\right) & \cdots & \mathbb{P}\left(s_N|s_1\right) \\ \mathbb{P}\left(s_1|s_2\right) & \mathbb{P}\left(s_2|s_2\right) & \cdots & \mathbb{P}\left(s_N|s_2\right) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{P}\left(s_1|s_N\right) & \mathbb{P}\left(s_2|s_N\right) & \cdots & \mathbb{P}\left(s_N|s_N\right) \end{pmatrix}$$

- Obviously, each row of the matrix sums to 1.



- Sample episodes starting from $S_1 = \text{C1}$:

- C1 C2 C3 Pass Sleep
- C1 FB FB C1 C2 Sleep
- C1 C2 C3 Pub C2 C3 Pass Sleep

Markov Reward Processes

- Markov Reward Process is a Markov Process + Reward.

Definition

A Markov Reward Process is a tuple $\langle \mathcal{S}, \mathcal{P}, \mathcal{R}, \gamma \rangle$:

- $\mathcal{S}$ is a (finite) set of states;
- $\mathcal{P}$ is a state transition probability matrix,

$$\mathcal{P}_{ss'} = \mathbb{P}\left(S_{t+1} = s' | S_t = s\right)$$

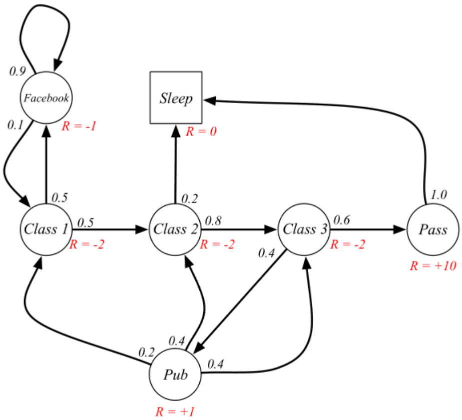
- $\mathcal{R}$ is a reward function,

$$\mathcal{R}_s = \mathbb{E}\left[R_{t+1} | S_t = s\right]$$

- γ is a discount factor, $\gamma \in [0, 1]$.

- If there are finite number of states, $\mathcal{R}$ can be a vector.

Example of MRP



- So that, we can represent $\mathcal{R}$ as:

$$\mathcal{R} = \begin{pmatrix} \mathcal{R}_{C1} \\ \mathcal{R}_{C2} \\ \mathcal{R}_{C3} \\ \mathcal{R}_{Pass} \\ \mathcal{R}_{Pub} \\ \mathcal{R}_{FB} \\ \mathcal{R}_{Sleep} \end{pmatrix} = \begin{pmatrix} -2 \\ -2 \\ -2 \\ +10 \\ +1 \\ -1 \\ 0 \end{pmatrix}$$

- Horizon
  - number of the maximum time steps in each episode;
  - can be infinite, otherwise called finite Markov (Reward) Process.
- Return
  - discounted sum of rewards from time step t to horizon:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots + \gamma^T R_{t+T+1}$$

  - The discount factor γ determines the present value of future rewards.

WHY discount factor γ

- Avoids infinity as a reward;
- If the reward is financial, immediate rewards may earn more interest than delayed rewards;
- Animal/human behavior shows preference for immediate reward;
- $\gamma \to 0$ leads to "myopic" evaluation;
- $\gamma \to 1$ leads to "farsighted" evaluation.

Value Function in MRP

Definition

The state value function $v(s)$ of an MRP is the expected return starting from state s:

$$\begin{aligned} v(s) &= \mathbb{E}\left[G_t | S_t = s\right] \\ &= \mathbb{E}\left[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots + \gamma^T R_{t+T+1} | S_t = s\right] \end{aligned}$$

Example of MRP

- The reward of Student MRP can be represented like that:

$$\mathcal{R}^\top = \begin{pmatrix} \mathcal{R}_{C1} & \mathcal{R}_{C2} & \mathcal{R}_{C3} & \mathcal{R}_{Pass} & \mathcal{R}_{Pub} & \mathcal{R}_{FB} & \mathcal{R}_{Sleep} \end{pmatrix}$$
$$= \begin{pmatrix} -2 & -2 & -2 & +10 & +1 & -1 & 0 \end{pmatrix}$$

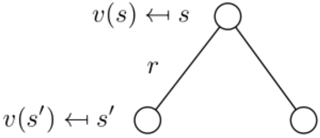
- Sample returns starting from $S_1 = \text{C1}$ with $\gamma = \frac{1}{2}$
  - C1 C2 C3 Pass Sleep ...
    - $v(C1) = -2 - 2 \times \frac{1}{2} - 2 \times \frac{1}{4} + 10 \times \frac{1}{8} + 0 = -2.25$
  - C1 FB FB C1 C2 Sleep ...
    - $v(C1) = -2 - 1 \times \frac{1}{2} - 1 \times \frac{1}{4} - 2 \times \frac{1}{8} - 2 \times \frac{1}{16} + 0 = -3.125$
  - C1 C2 C3 Pub C2 C3 Pass Sleep ...
    - $v(C1) = -2 - 2 \times \frac{1}{2} - 2 \times \frac{1}{4} + 1 \times \frac{1}{8} - 2 \times \frac{1}{16} \cdots = -3.41$
- How to compute the value function? For example, the value function of state C1 as $v(C1)$.

Bellman Equation for MRP(1)

- For any state s, the following equation holds between the value functions of s and its possible successor states:

$$v(s) = \mathcal{R}_s + \gamma \sum_{s'} \mathcal{P}_{ss'} v(s')$$

- The above equation is the Bellman Equation for MRP.



Bellman Equation for MRP(2)

- Now we try to derive the Bellman equation for $v(s)$.

$$\begin{aligned} v(s) &= \mathbb{E}\left[G_t | S_t = s\right] \\ &= \mathbb{E}\left[R_{t+1} + \gamma G_{t+1} | S_t = s\right] \\ &= \mathbb{E}\left[R_{t+1} | S_t = s\right] + \gamma \mathbb{E}\left[G_{t+1} | S_t = s\right] \\ &= \mathcal{R}_s + \gamma \mathbb{E}\left[G_{t+1} | S_t = s\right] \end{aligned}$$

Bellman Equation for MRP(3)

- Then we focus on the term $\gamma \mathbb{E}\left[G_{t+1} | S_t = s\right]$

$$
\begin{aligned}
\gamma \mathbb{E}\left[G_{t+1} | S_t = s\right] &= \gamma \sum_{g_{t+1}} g_{t+1} \cdot p(G_{t+1} = g_{t+1} | S_t = s) \\
&= \gamma \sum_{g_{t+1}} g_{t+1} \cdot \left(\sum_{s'} p\left(G_{t+1} = g_{t+1}, S_{t+1} = s' | S_t = s\right) \right) \\
&= \gamma \sum_{g_{t+1}} g_{t+1} \cdot \left(\sum_{s'} p\left(G_{t+1} = g_{t+1} | S_{t+1} = s', S_t = s\right) p\left(S_{t+1} = s' | S_t = s\right) \right) \\
&= \gamma \sum_{s'} \left(\sum_{g_{t+1}} g_{t+1} \cdot \left(G_{t+1} = g_{t+1} | S_{t+1} = s', S_t = s\right) \right) \mathcal{P}_{ss'} \\
&= \gamma \sum_{s'} \mathcal{P}_{ss'} v(s')
\end{aligned}
$$

- Therefore, we have $v(s) = \mathcal{R}_s + \gamma \sum_{s'} \mathcal{P}_{ss'} v(s')$.

- Moreover, we can express the Bellman Equation using the matrices:

$$V = \mathcal{R} + \gamma \mathcal{P} V$$

$$\begin{bmatrix} v(s_1) \\ v(s_2) \\ \vdots \\ v(s_N) \end{bmatrix} = \begin{bmatrix} R(s_1) \\ R(s_2) \\ \vdots \\ R(s_N) \end{bmatrix} + \gamma \begin{bmatrix} \mathbb{P}(s_1|s_1) & \mathbb{P}(s_2|s_1) & \cdots & \mathbb{P}(s_N|s_1) \\ \mathbb{P}(s_1|s_2) & \mathbb{P}(s_2|s_2) & \cdots & \mathbb{P}(s_N|s_2) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{P}(s_1|s_N) & \mathbb{P}(s_2|s_N) & \cdots & \mathbb{P}(s_N|s_N) \end{bmatrix} \begin{bmatrix} v(s_1) \\ v(s_2) \\ \vdots \\ v(s_N) \end{bmatrix}$$

- It can be solved directly: $V = (I - \gamma \mathcal{P})^{-1} \mathcal{R}$;
- Only possible for small MRPs.

Markov Decision Processes

- Markov Decision Process is a Markov Reward Process + Decisions.

Definition

A Markov Decision Process is a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$:

- $\mathcal{S}$ is a (finite) set of states;
- $\mathcal{A}$ is a (finite) set of actions;
- $\mathcal{P}$ is a state transition probability matrix:

$$\mathcal{P}_{ss'}^{a} = \mathbb{P}\left(S_{t+1} = s' | S_t = s, A_t = a\right)$$

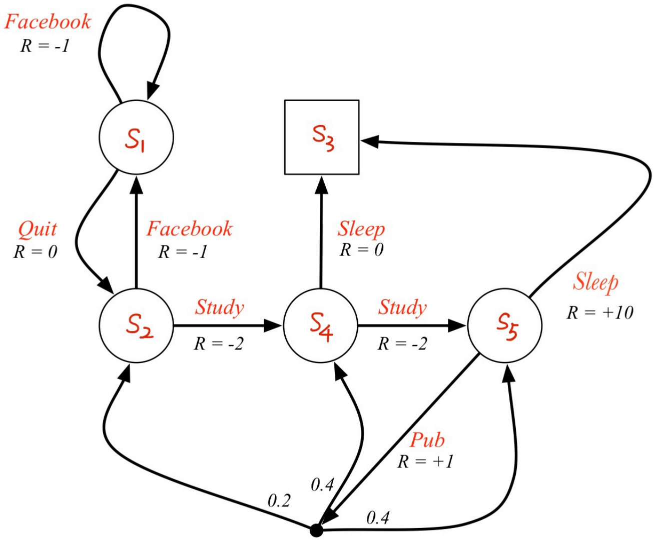
- $\mathcal{R}$ is a reward function:

$$\mathcal{R}_{s}^{a} = \mathbb{E}\left[R_{t+1} | S_t = s, A_t = a\right]$$

$$\mathcal{R}_{ss'}^{a} = \mathbb{E}\left[R_{t+1} | S_t = s, A_t = a, S_{t+1} = s'\right]$$

- γ is a discount factor, $\gamma \in [0, 1]$.

Example of MDP



Policy in MDP(1)

Definition

A policy π is a distribution over actions given states:

$$\pi(a|s) = \mathbb{P}\left[A_t = a | S_t = s\right]$$

- A Policy fully defines the behavior of an agent, can be deterministic or stochastic;
- Policies are stationary, i.e., $A_t \sim \pi(a|s), \forall t > 0$.

Policy in MDP(2)

- Given an MDP $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ and a policy π, we have:

$$\mathcal{P}^{\pi}_{ss'} = \sum_{a \in \mathcal{A}(s)} \pi(a|s) \mathcal{P}^{a}_{ss'}$$

$$\mathcal{R}^{\pi}_{s} = \sum_{a \in \mathcal{A}(s)} \pi(a|s) \mathcal{R}^{a}_{s}$$

- The state sequence $S_1, S_2, \ldots$ is a Markov Process $\langle \mathcal{S}, \mathcal{P}^{\pi} \rangle$;
- The state and reward sequence $S_1, R_2, S_2, R_3 \ldots$ is a Markov Reward Process $\langle \mathcal{S}, \mathcal{P}^{\pi}, \mathcal{R}^{\pi}, \gamma \rangle$.

Value Function in MDP - State Value Function

Definition

The state value function $v_\pi(s)$, is the expected return starting from state s and following policy π:

$$v_\pi(s) = \mathbb{E}_\pi\left[G_t | S_t = s\right] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s\right]$$

- The state value function specifies the goodness of a state;

State	Value
State1	0.3
State2	0.9

Definition

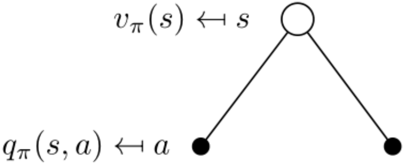
The action value function $q_\pi(s, a)$, is the expected return starting from state s, taking action a, and then following policy π:

$$q_\pi(s, a) = \mathbb{E}\left[G_t | S_t = s, A_t = a\right] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s, A_t = a\right]$$

- The action value function specifies the goodness of an action in a state;

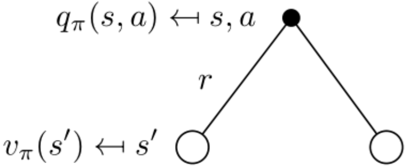
State	Action	Value
State1	Action1	0.03
State1	Action2	0.02
State2	Action1	0.5
State2	Action2	0.9

Relationships between $v_\pi(s)$ and $q_\pi(s, a)$(1)



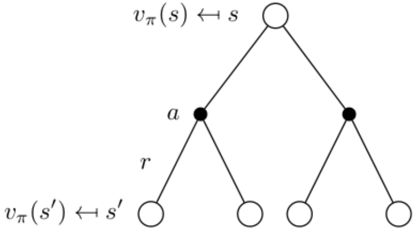
$$v_\pi(s) = \sum_{a \in \mathcal{A}(s)} \pi(a|s) q_\pi(s, a)$$

Relationships between $v_\pi(s)$ and $q_\pi(s,a)$(2)

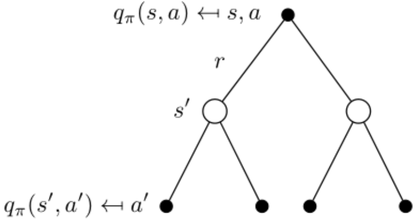


$$
\begin{aligned}
q_\pi(s,a) &= \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a \left(\mathcal{R}_{ss'}^a + \gamma v_\pi(s') \right) \\
&= \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a v_\pi(s')
\end{aligned}
$$

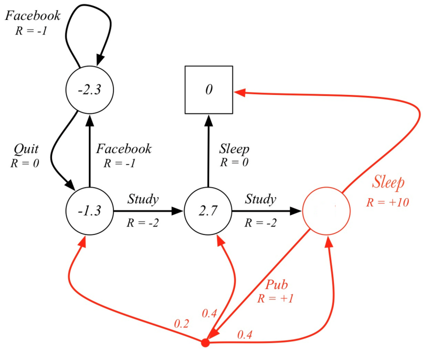
Bellman Equation for $v_\pi(s)$



$$v_\pi(s) = \sum_{a \in \mathcal{A}(s)} \pi(a|s) \left(\mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a v_\pi(s') \right)$$



$$q_\pi(s,a) = \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a \sum_{a' \in \mathcal{A}(s')} \pi(a'|s') q_\pi(s', a')$$



- Assume that $\gamma = 1$, and the agent selects actions with equal probability in each state;
- We have the Bellman Equation for $v_\pi(s)$:

$$v_\pi(s) = \sum_{a \in \mathcal{A}(s)} \pi(a|s) \left(\mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a v_\pi(s') \right)$$

- Similarily, we can express the Bellman Equation using the matrices:

$$V_\pi = \mathcal{R}^\pi + \gamma \mathcal{P}^\pi V_\pi$$

$$\begin{bmatrix} v_\pi(s_1) \\ v_\pi(s_2) \\ \vdots \\ v_\pi(s_N) \end{bmatrix} = \begin{bmatrix} R^\pi_{s_1} \\ R^\pi_{s_2} \\ \vdots \\ R^\pi_{s_N} \end{bmatrix} + \gamma \begin{bmatrix} \mathbb{P}^\pi(s_1|s_1) & \mathbb{P}^\pi(s_2|s_1) & \cdots & \mathbb{P}^\pi(s_N|s_1) \\ \mathbb{P}^\pi(s_1|s_2) & \mathbb{P}^\pi(s_2|s_2) & \cdots & \mathbb{P}^\pi(s_N|s_2) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbb{P}^\pi(s_1|s_N) & \mathbb{P}^\pi(s_2|s_N) & \cdots & \mathbb{P}^\pi(s_N|s_N) \end{bmatrix} \begin{bmatrix} v_\pi(s_1) \\ v_\pi(s_2) \\ \vdots \\ v_\pi(s_N) \end{bmatrix}$$

- It can be solved directly: $V_\pi = (I - \gamma \mathcal{P}^\pi)^{-1} \mathcal{R}^\pi$.

Optimal Policy

- Value functions define a partial ordering over policies:

$$\pi \geq \pi' \text{ if and only if } \forall s \in \mathcal{S}, v_\pi(s) \geq v_{\pi'}(s)$$

Theorem

For any finite Markov Decision Process:

- *There are always one or more policies that are better than or equal to all other policies. These are the Optimal Policies, denoted as π^*:*

$$\pi^* \geq \pi, \forall \pi$$

- *Optimal Policies share the same Optimal State-Value Function:*

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

- *Optimal Policies also share the same Optimal Action-Value Function:*

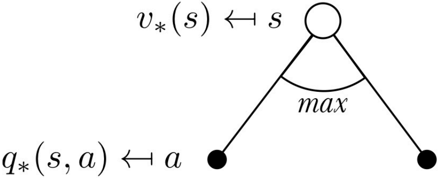
$$q_{\pi^*}(s,a) = q_*(s,a) = \max_\pi q_\pi(s,a), \forall s \in \mathcal{S}, a \in \mathcal{A}(s)$$

Finding an optimal policy

An optimal policy can be found by maximizing over $q_*(s, a)$:

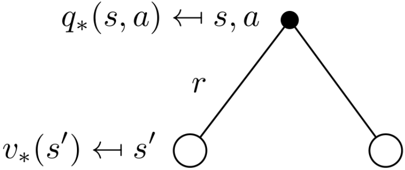
$$\pi_*(a|s) = \begin{cases} 1 & \text{if } a = \underset{a \in \mathcal{A}(s)}{argmax}\, q_*(s, a) \\ 0 & \text{otherwise} \end{cases}$$

- There is always a deterministic optimal policy for any MDP;
- If we know $q_*(s, a)$, we immediately have the optimal policy.



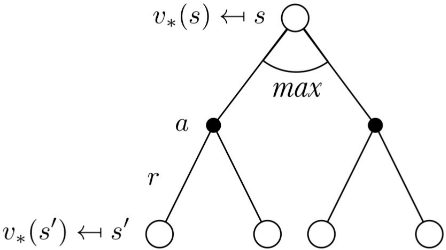
$$v_*(s) = \max_{a \in \mathcal{A}(s)} q_*(s, a)$$

Relationships between $v_*(s)$ and $q_*(s, a)$(2)



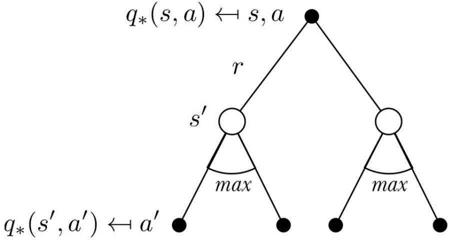
$$q_*(s, a) = \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a v_*(s')$$

Bellman Optimality Equation for $v_*(s)$



$$v_*(s) = \max_{a \in \mathcal{A}(s)} \left(\mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a v_*(s') \right)$$

Bellman Optimality Equation for $q_*(s, a)$



$$q_*(s, a) = \mathcal{R}_s^a + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'}^a \max_{a' \in \mathcal{A}(s')} q_*(s', a')$$

Solving the Bellman Optimality Equation

- Finding an optimal policy by solving the Bellman Optimality Equation relies on at least three assumptions:
  - accurate knowledge of environment dynamics or state transition probability matrix;
  - enough computational resources;
  - the Markov Property.
- We usually have to settle for approximations.

Summary

- Markov Property & Markov Process;
- Markov Reward Process;
  - Horizon & Reward & Return;
  - Value Function;
  - Bellman Equation for MRPs;
- Markov Decision Process;
  - Policy;
  - State Value Function & Action Value Function;
  - Bellman Equation for MDPs;
  - Optimal Policy & Optimal State Value Function & Optimal Action Value Function;
  - Bellman Optimality Equation.

More...

- Reinforcement Learning in OR/OM;
  - Assortment Optimization;
  - Combinatorial Optimization, e.g. TSP;
  - Policy Gradient in Non-Convex Optimization;
  - ...