IE 498: Online Learning and Decision Making, Fall 2019 10/15/19

Lecture 15: Introduction to Reinforcement Learning

Lecturer: Tanmay Gangwani *Scribe: Ruoyi Feng*

In this lecture, we discussed about Reinforcement learning (RL) problems. We described the framework of reinforcement learning problems and talked about the mathematical modeling based on Markov decision process (MDP). We also introduced some important mathematical properties of reinforcement learning problem, such as value functions and Bellman equations.

1 The Agent–Environment Interface

The reinforcement learning problem is meant to be a straightforward framing of the problem of learning from interaction to achieve a goal. The learner and decision-maker is called the *agent*. The thing it interacts with, comprising everything outside the agent, is called the *environment*. They interact continually. The agent receives some representation of the environment's *state* and selects *actions*. Then environment responding to those actions and presenting new situations to the agent. The environment also gives rise to *rewards*, special numerical values that the agent tries to maximize over time [1].

Assume 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, $s_t \in \mathcal{S}$, where $\mathcal{S}$ is the set of possible states and on that basis selects an action, $a_t \in \mathcal{A}$, where $\mathcal{A}$ is the set of actions available. It receives a reward $r(s_t, a_t) \in \mathcal{R}$, where $\mathcal{R}$ is the reward distribution, and finds itself in a new state s_{t+1}.

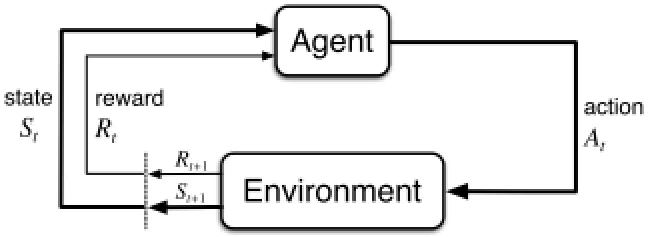

Figure 1: The agent-environtment interface in reinforcement learning

2 Markov Decision Process

2.1 Definition

In previous lectures, we have discussed about multi-armed bandit and contextual bandits problems. In multi-armed bandit problems, the objective is to select the action (arm) to maximize the immediate (one-step) reward. In contextual bandits, the reward distribution depends on the context, along with the action.

RL is a *sequential decision making* framework that differs from bandits by adding a temporal dimension to the problem. The objective is to maximize the sum of rewards, generated by a sequential data-generation process which involves repeated interaction between agent and the environment. The environment is usually modeled as a *Markov Decision Process*, or MDP, where given (s_t, a_t), the next-state distribution is independent of $(s_{<t}, a_{<t})$. This is also known as the transition dynamics $\mathcal{T}(s_{t+1}|s_t, a_t)$.

2.2 Notation

t	t_{th} time step
$\mathcal{S}$	The set of possible states
$\mathcal{A}$	The set of possible actions
s_t	Environment's state at time t, $s_t \in \mathcal{S}$
a_t	Action at time t, $a_t \in \mathcal{A}$
$r(s_t, a_t)$	Reward at time t, $r(s_t, a_t) \in \mathbb{R}$
$\mathcal{T}(s_{t+1}\|s_t, a_t)$	Transition dynamics (Markov property)
$\pi(a\|s)$	Agent's policy distribution
$\mu_0(s)$	Start state distribution
γ	Discount rate, $\gamma \in [0, 1)$

The sequential data-generation process in RL can be described as below:

$$\textit{Start time step}\begin{cases} s_0 \sim \mu_0 \\ a_0 \sim \pi(a_0|s_0) \\ s_1 \sim \mathcal{T}(s_1|s_0, a_0) \end{cases}$$

$$i_{th}\ \textit{time step}\begin{cases} a_i \sim \pi(a_i|s_i) \\ s_{i+1} \sim \mathcal{T}(s_{i+1}|s_i, a_i) \end{cases}$$

$$\vdots$$

The objective in RL is to maximize the expected sum of rewards, where the expectation is taken over the random variables in the data generation process. We can write the objective for both the **finite-horizon case**, where data generation terminates after H steps, and an **infinite-horizon case**, where there is no termination time-period. Equation(1) shows the former, Equation(2) the latter with a discount factor $\gamma \in [0, 1)$. In the following sections, we focus on the situation of infinite horizon.

$$\eta(\pi) = \mathop{\mathbb{E}}_{\mu_0, \mathcal{T}, \pi}(\sum_{t=0}^{H} r(s_t, a_t)) \tag{1}$$

$$\eta(\pi) = \mathop{\mathbb{E}}_{\mu_0, \mathcal{T}, \pi}(\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)) \tag{2}$$

$$\pi^* = \arg\max_{\pi} \eta(\pi) \tag{3}$$

Assume bounded rewards, i.e. $r \in [-R_{max}, R_{max}]$. In finite horizon, $\eta(\pi) \leq H.R_{max}$; in infinite horizon, $\eta(\pi) \leq \frac{R_{max}}{1-\gamma}$. Therefore, the objective functions are bounded.

3 Value Functions

3.1 Definition

$V^\pi(s)$, known as the State-value-function for a policy π, is a function of state only. $Q^\pi(s,a)$, known as the State-action-value-function for a policy π, is a function of state and action. Intuitively, these functions indicate how *good* the state (or state-action pair) is, when using policy π to decide actions. The notion of *good* here is defined in terms of expected discounted cumulative rewards, as detailed below.

3.2 Notation

$Q^\pi(s,a)$	State-action-value function for policy π
$V^\pi(s)$	State-value function for policy π

$$Q^\pi(s,a) = \mathop{\mathbb{E}}_{\mu_0,\mathcal{T},\pi}\left(\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)\right|_{s_0=s,a_0=a} \tag{4}$$

$$V^\pi(s) = \mathop{\mathbb{E}}_{\mu_0,\mathcal{T},\pi}\left(\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)\right|_{s_0=s} \tag{5}$$

Then, the relation between state-action-value function, state-value function and objective function are:

$$\eta(\pi) = \mathop{\mathbb{E}}_{s\sim\mu_0,a\sim\pi(a|s)}(Q^\pi(s,a)) \tag{6}$$

$$V^\pi(s) = \mathop{\mathbb{E}}_{a\sim\pi(a|s)}(Q^\pi(s,a)) \tag{7}$$

4 Occupancy Measure

4.1 Definition

The (discounted) occupancy measure of a policy, denoted by $\rho^\pi(s,a)$, can be understood as the stationary distribution over the $\mathcal{S}\times\mathcal{A}$ space, induced by running policy π in the environment.

4.2 Notation

$\rho^\pi(s,a)$	Occupancy measure for policy π
$\hat{\rho}^\pi(s,a)$	Unnormalized occupancy measure for policy π
$\mathrm{P}(\mathrm{s}_t = s, a_t = a \mid \mu_0, \pi, \mathcal{T})$	Probability of landing in state s and taking action a at time t, when following policy π from an initial state sampled from μ_0, in an environment with transition dynamics $\mathcal{T}$

$$P(s_t = s, a_t = a|\mu_0, \pi, \mathcal{T}) = P(s_t = s|\mu_0, \pi, \mathcal{T}) * \pi(a_t = a|s_t = s) \tag{8}$$

$$\hat{\rho}^\pi(s,a) \overset{\text{def}}{=\!=} \sum_{t=0}^{\infty} \gamma^t P(s_t = s, a_t = a|\mu_0, \pi, \mathcal{T}) \tag{9}$$

$$\rho^\pi(s,a) = \frac{\hat{\rho}^\pi(s,a)}{\sum_{s,a}\hat{\rho}^\pi(s,a)} = (1-\gamma) * \hat{\rho}^\pi(s,a) \tag{10}$$

We can rewrite the RL objective (Equation 2) using the (discounted) occupancy measure:

$$\begin{aligned}
\eta(\pi) &= \mathop{\mathbb{E}}_{\mu_0,\mathcal{T},\pi}(\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)) \\
&= \sum_{t=0}^{\infty}\gamma^t \mathop{\mathbb{E}}_{\mu_0,\mathcal{T},\pi} r(s_t,a_t) \\
&= \sum_{t=0}^{\infty}\gamma^t \mathop{\mathbb{E}}_{s_t\sim P(s_t|\mu_0,\pi,\mathcal{T})}(\mathop{\mathbb{E}}_{a_t\sim\pi(a_t|s_t)} r(s_t,a_t)) \\
&= \sum_{t=0}^{\infty}\gamma^t[\sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} P(s_t = s|\mu_0,\pi,\mathcal{T}) * \pi(a_t = a|s_t = s) * r(s,a))] \\
&= \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}}\sum_{t=0}^{\infty}\gamma^t P(s_t = s|\mu_0,\pi,\mathcal{T}) * \pi(a_t = a|s_t = s) * r(s,a)) \\
&= \mathop{\mathbb{E}}_{s,a\sim\hat{\rho}^\pi(s,a)} r(s,a)
\end{aligned} \tag{11}$$

4.3 Properties

- The occupancy measure $\rho^\pi(s,a)$ has a one-to-one relationship with policy π. This means that if two policies have the same occupancy measures, then these two policies must be the same, and vice-versa.
- **Bellman Flow Constraint:** This is defined as:

  $$\chi(s,a) = \pi(a|s) * [(1-\gamma)\mu_0(s) + \gamma\int \chi(s',a')\mathcal{T}(s|s',a')\mathrm{d}s'\mathrm{d}a']; \qquad \chi(s,a) \geq 0. \tag{12}$$

  It can be shown that occupancy measure $\rho^\pi(s,a)$ is a **unique solution** to Bellman flow constraint.

5 Bellman Equations

Bellman Equations define back-up recursions for the state and state-action value functions.

5.1 Bellman Expectation Equations

$V^\pi(s)$ and $Q^\pi(s,a)$ satisfy the Bellman Expectation Equations.

$$V^{\pi}(s) = \mathop{\mathbb{E}}_{a\sim\pi(a|s),s\sim\mathcal{T}(s'|s,a)}(r(s,a)+\gamma V^{\pi}(s')) \tag{13}$$

$$Q^{\pi}(s,a) = \mathop{\mathbb{E}}_{a'\sim\pi(a'|s'),s\sim\mathcal{T}(s'|s,a)}(r(s,a)+\gamma Q^{\pi}(s',a')) \tag{14}$$

$$\tag{15}$$

5.2 Bellman Optimality Equations

$V^*(s)$ and $Q^*(s,a)$ satisfy the Bellman Optimality Equations.

π^*	The optimal policy, $\pi^* = \arg\max_{\pi}\eta(\pi)$
Q^*	Q function for π^*
V^*	V function for π^*

$$V^*(s) = \max_a [r(s,a) + \gamma \mathop{\mathbb{E}}_{s'\sim\mathcal{T}(s'|s,a)}(V^*(s')] \tag{16}$$

$$Q^*(s,a) = \mathop{\mathbb{E}}_{s\sim\mathcal{T}(s'|s,a)}(r(s,a)+\gamma\max_{a'}Q^*(s',a')) \tag{17}$$

$$\tag{18}$$

The relationship between $V^*(s)$ and $Q^*(s,a)$:

$$V^*(s) = \max_a Q^*(s,a)$$

References

[1] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction.* MIT press, 2018.