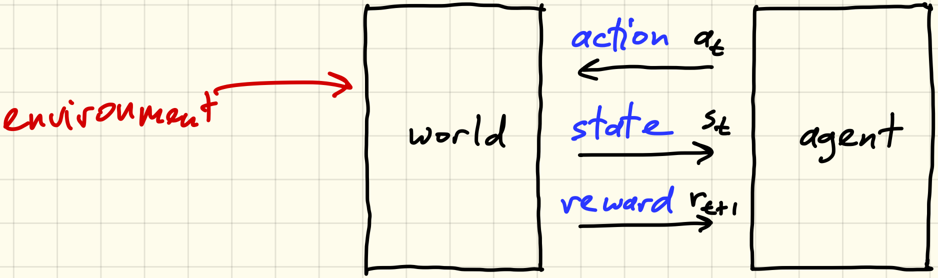
MDP: Markov Decision Process (S&B Ch 3)

- stochastic, discrete state, discrete action, state feedback



t = time

3 system spaces: state $s_t \in S$, action $a_t \in A$, reward $r_t \in \mathbb{R}$

3 system functions: model $p(s_{t+1} | s_t, a_t)$ = probability of s_{t+1} given s_t, a_t

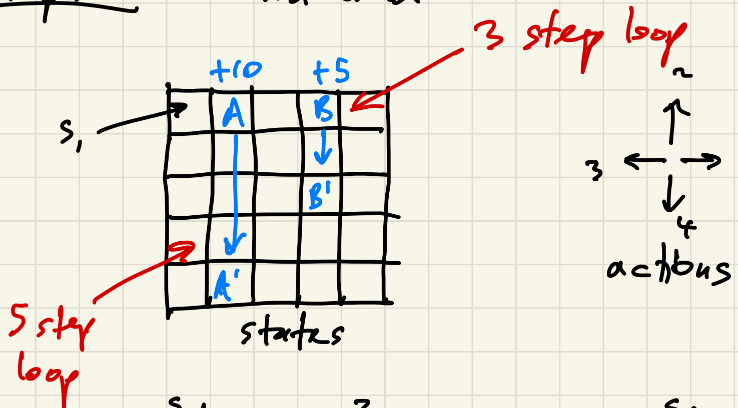
policy $\pi(a_t | s_t)$ = probability of a_t given s_t

reward $r_{t+1} = R(s_t, a_t)$ ⟵ not stochastic (if so, take $E[\,]$)

model-based RL: algorithms can directly call $p()$

model-free RL: only have access to $p()$ via trajectories

Example : Gridworld



$$\begin{array}{cccc} & s_{t+1} & s_t & a_t \\ p = & 25 \times & 25 \times & 4 \end{array}$$

$$R = 25 \times 4$$

$$s_{t+1} \sim p(\cdot \mid s_t, a_t)$$

$$p[:, s_t, a_t] = \underbrace{[0.1 \ 0.3 \ \dots]}_{25}$$

$s \in \{1, \dots, 25\}$

$a \in \{1, \dots, 4\}$

$$V = \underset{1}{\left[\quad\right]_{25}} \qquad Q = \underset{4}{\left[\qquad\right]_{25}} \qquad \pi = \underset{25}{\left[\qquad\right]_{4}}$$

Policy evaluation (S&B 4.1): prediction $\quad \pi \longrightarrow V \quad$ (know p)

$$V_\pi(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}\left[R(s,a) + \gamma\, \mathbb{E}_{s'\sim p(\cdot|s,a)}\left[V_\pi(s')\right]\right]$$

$$V_\pi(s) - \gamma \sum_{s'} \underbrace{\left(\sum_a \pi(a|s)\, p(s'|s,a)\right)}_{A(s,s') \,\leftarrow\, |S|\times|S|} V_\pi(s') = \underbrace{\sum_a \pi(a|s) R(s,a)}_{b(s) \,\leftarrow\, |S|\times 1}$$

model-based

$$(I - \gamma A)\, V = b$$

Iterative solution:

$$v_{k+1}(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}\left[R(s,a) + \gamma\, \mathbb{E}_{s'\sim p(\cdot|a,s)}\left[V_k(s')\right]\right] \quad \forall s$$

Policy improvement (S & B 4.2) $\pi \to \pi'$

Policy improvement theorem Given π, π' deterministic policies

If $Q_\pi(s, \pi'(s)) \geq V_\pi(s) \quad \forall s$ one step improvement

then $V_{\pi'}(s) \geq V_\pi(s) \quad \forall s$ $\Rightarrow$ all-steps improvement

Greedy policy: $\pi'(s) = \arg\max_a Q_\pi(s,a)$

Policy iteration (S & B 4.3)

$$\pi_0 \xrightarrow{E} V_{\pi_0} \xrightarrow{I} \pi_1 \xrightarrow{E} V_{\pi_1} \xrightarrow{I} \pi_2 \to \dots \to \pi^* \to V^*$$

Value iteration (S & B 4.4)

$$V_{k+1}(s) = \max_a \left(R(s,a) + \gamma \, E_{s' \sim p(\cdot|s,a)} [V_k(s')] \right) \quad \forall s$$

Model-free methods $\longrightarrow$ only access trajectories

Temporal-difference TD(0) (S&B 6.1) $\pi \longrightarrow V$

$$(s_0, a_0, r_1, s_1, a_1, r_2, s_2, a_2, \ldots\ldots) \longleftarrow \text{trajectory}$$

$$(s_t, a_t, r_{t+1}, s_{t+1}) \longleftarrow \text{piece of trajectory}$$

$$V_\pi(s_t) = E_{a_t \sim \pi(\cdot|s_t)} E_{s_{t+1} \sim p(\cdot|s_t, a_t)} \left[r_{t+1} + \gamma V_\pi(s_{t+1}) \right]$$

model-based method, $E \rightarrow \sum_{a_t}, \sum_{s_{t+1}}$

model-free method, sample a_t, s_{t+1}

$\cancel{V_\pi(s_t) = r_{t+1} + \gamma V_\pi(s_{t+1})}$ ~~where~~ $\cancel{(s_t, a_t, r_{t+1}, s_{t+1})}$ ~~is a sample~~

doesn't work, because one sample is not enough

$$V_\pi(s_t) \sim \underbrace{r_{t+1} + \gamma V_\pi(s_{t+1})}_{\text{target}}$$

want to average these

— use incremental average

$$V_\pi(s_t) = (1-\alpha)\, V_\pi(s_t) + \alpha \underbrace{\left(r_{t+1} + \gamma V_\pi(s_{t+1})\right)}_{\text{target}}$$

$$\alpha = \text{learning rate} \in (0,1)$$

$$V_\pi(s_t) = V_\pi(s_t) + \alpha \underbrace{\left(r_{t+1} + \gamma V_\pi(s_{t+1}) - V_\pi(s_t)\right)}_{\delta_t \,=\, \text{incement}}$$

where $a_t \sim \pi(\cdot \mid s_t)$ and $s_{t+1} \sim p(\cdot \mid s_t, a_t)$

sampled from

Q: how do we improve our policy?

Without access to p, we can't convert $V \to \pi$

But, we can convert $Q \to \pi$ $\quad \pi(s) = a = \arg\max_{a'} Q(s, a')$

without knowing p

We are going to work with Q, not V.

SARSA (S&B 6.4)

$Q \to$ better Q

$\cancel{\pi \to Q}$

Use traj segments

$$\begin{matrix} S & A & R & S & A \\ (s_t, & a_t, & r_{t+1}, & s_{t+1}, & a_{t+1}) \end{matrix}$$

$$Q(s_t, a_t) = r_{t+1} + \gamma E_{s_{t+1} \sim p(\cdot | s_t, a_t)} \left[E_{a_{t+1} \sim \pi(\cdot | s_{t+1})} [Q(s_{t+1}, a_{t+1})] \right]$$

simplified: $Q(s_t, a_t) = r_{t+1} + \gamma E[Q(s_{t+1}, a_{t+1})]$

$$= E[\underbrace{r_{t+1} + \gamma Q(s_{t+1}, a_{t+1})}_{\text{target}}]$$

"on-policy" learning

targets should be sampled from E distribution

$$Q(s_t, a_t) \leftarrow (1-\alpha) Q(s_t, a_t) + \alpha (\underbrace{r_{t+1} + \gamma Q(s_{t+1}, a_{t+1})}_{\text{target}})$$

$$= Q(s_t, a_t) + \alpha (\underbrace{r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)}_{\delta_t = \text{increment}})$$

Given $(s_t, a_t, r_{t+1}, s_{t+1}, a_{t+1})$

$(s_0, a_0, r_1, s_1, a_1)$

$(17, 3, 2, 2, 1)$

$Q = [\;\cdot\;]$ 25 × 4

$0 \to 0.2$

Update $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \delta_t$

$$\delta_t = r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)$$

Choose actions:

greedy: $a_t = \arg\max_a Q(s_t, a)$ ← best choice based on current Q function (no exploration) (all exploitation)

ε-greedy: $a_t = \begin{cases} \text{random } a & \text{with probability } \varepsilon \\ \arg\max_a Q(s_t, a) & \text{otherwise} \end{cases}$

exploration/exploitation tradeoff param.

ε-greedy is hopefully close to optimal when Q is close to optimal

Thm Sarsa converges to the optimal Q if all (s,a) pairs are visited infinitely often and the policy converges to greedy. (e.g. $\varepsilon = \frac{1}{t}$)

Sarsa will converge to the Q for whichever policy we use.

If we use ε-greedy $\longrightarrow$ converge to a good Q

This is "on-policy" learning, where we need to choose good actions.

This means that aggressive exploration hurts learning.

$\longrightarrow$ instead, we can use Q-learning, which doesn't have this problem (Q-learning is off-policy)

Q-learning (S&B 6.5)

Use $(s_t, a_t, r_{t+1}, s_{t+1})$

Won't assume that are chosen from Q

$$Q(s_t, a_t) = r_{t+1} + E_{s_{t+1} \sim p(\cdot \mid s_t, a_t)}\left[E_{a_{t+1} \sim \pi(\cdot \mid s_{t+1})} \left[Q(s_{t+1}, a_{t+1}) \right] \right]$$

next time.