state ↑ action ↑ reward $R(s,a,s')$ ↗ discount

Markov Decision Process (MDP)

$$M = (S, A, T, R, \gamma)$$

· total: $\sum_t \gamma^t r_t$
· max: $\frac{r_{max}}{1-\gamma}$

transition $T(s,a,s') = Pr[s'|s,a]$

- Value function $V_t^{\pi}(s)$:

  expected total reward with ① policy π ② begin in s ③ have t steps remaining

  - Finite horizon optimal: $V_t^*(s) = \max_{a\in A} [R(s,a) + \gamma \sum_{s'} P(s'|s,a) V_{t-1}^*(s)]$.
    $V_0^* = 0$
  - Infinite horizon optimal: $V^*(s) = \max_{a \in A} [R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^*(s)]$.

- Policy iteration:

  ① Policy evaluation: $\begin{cases} \text{Value iteration: } V_t^{\pi}(s) \leftarrow R(s,\pi(s)) + \gamma \sum_{s'} P(s'|s,a) V_{t-1}^{\pi}(s) \\ \qquad V_0^{\pi}(s) \leftarrow 0 \\ \text{Solve linear system (Bellman equation)} \end{cases}$

  ② Policy improvement: For state s, $\pi(s) \leftarrow \arg\max_{a\in A} [R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^{\pi}(s')]$.

Reinforcement Learning

Transition T / Reward R unknown!

- Monte-Carlo method:

  ① Monte-Carlo prediction $\begin{cases} \text{Average: average}(G_t^{\pi}(s)) \\ \text{Incremental: } V^{\pi}(S_t) \leftarrow V^{\pi}(S_t) + \alpha [G_t^{\pi} - V^{\pi}(S_t)] \end{cases}$

  first-visit / every-visit

  - Functional approximation: $\theta_i \leftarrow \theta_i + \alpha(u_j(s) - \hat{V}_\theta(s)) \frac{\partial \hat{V}_\theta(s)}{\partial \theta_i}$.

  ② Monte-Carlo control: $\begin{cases} \text{Naïve: } \pi(s) \leftarrow \arg\max_a Q(s,a) \text{ in each episode} \\ \varepsilon\text{-soft: } \pi(a|s) \leftarrow \begin{cases} 1-\varepsilon+\varepsilon/|A| & \text{if } a = a^* \\ \varepsilon/|A(s)| & \text{if } a \neq a^* \end{cases} \end{cases}$

  on-policy ←

- Temporal difference method

  ① TD prediction: $V(S_t) \leftarrow V(S_t) + \alpha [R_t + \gamma V(S_{t+1}) - V(S_t)]$.

  - Functional approximation: $\theta_i \leftarrow \theta_i + \alpha [R_t + \gamma \hat{V}_\theta(s') - \hat{V}_\theta(s)] \frac{\partial \hat{V}_\theta(s)}{\partial \theta_i}$.

  ② TD control $\begin{cases} \text{SARSA: } Q(S,A) \leftarrow Q(S,A) + \alpha [R + \gamma Q(S',A') - Q(S,A)] \\ \text{Q-learning: } Q(S,A) \leftarrow Q(S,A) + \alpha [R + \gamma \max_a Q(S',a) - Q(S,A)] \end{cases}$

  - Functional approximation: $\theta_i \leftarrow \theta_i + \alpha [R + \gamma \max_{a'} \hat{Q}(s',a') - \hat{Q}(s,a)] \frac{\partial \hat{Q}(s,a)}{\partial \theta_i}$.

- Monte-Carlo policy gradient: $\theta_{j+1} \leftarrow \theta_j + \alpha u_j \nabla_\theta \ln \pi_\theta(s, a_j)$

Partially Observable Markov Decision Process (POMDP)

$$M = (S, A, E, T, O, R, \gamma)$$

evidences/observations

observation function $O(s,e) = P(e|s)$

- Belief state $b(s)$
  - Update: $b'(s') = \alpha Pr[e'|s'] \sum_s Pr[s'|s,a] b(s)$.
- Value iteration: $\alpha_p(s) = \sum_{s'} Pr[s'|s,a] [R(s,a,s') + \gamma \sum_{e'} Pr[e'|s'] \alpha_{p,e'}(s')]$