

Model-free reinforcement learning

Christophe Eloy

Outline

- Summary on MDPs and Bellman equations
- Model-free prediction
 - Monte Carlo
 - Temporal difference
- Model-free control
 - SARSA
 - Q-learning

Markov Decision Process

- Finite set of states \mathcal{S}
- Finite set of actions \mathcal{A}
- State transition matrix

$$\mathcal{P}_{ss'}^a = \mathbb{P} [S_{t+1} = s' | S_t = s, A_t = a]$$

- Reward function

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

- Discount factor: $\gamma \in [0,1]$

Policy and value functions

- Stochastic policy

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

- Return: $G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots$
- State-value function

$$v_{\pi}(s) = \mathbb{E}_{\pi} [G_t | S_t = s]$$

- Action-value function (Q-function)

$$q_{\pi}(s, a) = \mathbb{E}_{\pi} [G_t | S_t = s, A_t = a]$$

Bellman equations

- Bellman equation for state-value and action-value functions

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

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

- Bellman optimality equations ($v_*(s) = \max_a q_*(s, a)$)

$$v_*(s) = \max_a \left[\mathcal{R}_s^a + \gamma \sum_{s'} \mathcal{P}_{ss'}^a v_*(s') \right]$$

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

Model-free prediction

- Dynamic programming
- Iterative procedure to approach state-value function, given a known policy
- Monte-Carlo algorithm (high variance, no bias)

$$V(S_t) \leftarrow V(S_t) + \alpha (G_t - V(S_t))$$

- Temporal difference algorithm (lower variance, some bias)

$$V(S_t) \leftarrow V(S_t) + \alpha (R_{t+1} + \gamma V(S_{t+1}) - V(S_t))$$

Theorem: MC and TD algorithm converges towards $v_\pi(s)$

Model-free control

- Iterative procedure to approach Q-function, given sequences S, A, R, S', A'

- Monte Carlo algorithm (on-policy)

$$Q(S, A) \leftarrow Q(S, A) + \frac{1}{N(S, A)} (G - Q(S, A))$$

- SARSA algorithm (on-policy)

$$Q(S, A) \leftarrow Q(S, A) + \alpha (R + \gamma Q(S', A') - Q(S, A))$$

- Q-learning algorithm (off-policy)

$$Q(S, A) \leftarrow Q(S, A) + \alpha \left(R + \gamma \max_{a'} Q(S', a') - Q(S, A) \right)$$

Theorem: These algorithms converge towards q^*

DP, MC and TD

- Dynamic programming, Monte Carlo, temporal difference

- State-value function

$$V(S) \leftarrow \mathbb{E} [R + \gamma V(S') | S] \quad \text{DP (iterative policy evaluation)}$$

$$V(S) \leftarrow V(S) + \alpha (G - V(S)) \quad \text{MC}$$

$$V(S) \leftarrow V(S) + \alpha (R + \gamma V(S') - V(S)) \quad \text{TD}$$

- Q-function

$$Q(S, A) \leftarrow \mathbb{E} [R + \gamma Q(S', A') | S, A] \quad \text{DP}$$

$$Q(S, A) \leftarrow Q(S, A) + \alpha (G - Q(S, A)) \quad \text{MC}$$

$$Q(S, A) \leftarrow Q(S, A) + \alpha (R + \gamma Q(S', A') - Q(S, A)) \quad \text{TD (SARSA algorithm)}$$