

Dynamic Programming (of Model based)

• Optimal Policy

- policy π a distribution over action given states.

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

stationary. time-independent.

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

$s \xrightarrow{a} s' \quad s \xrightarrow{a} a \rightarrow s'$

$$R_s^{\pi} = \sum_{a \in A} \pi(a|s) R_s^a \quad \text{예상 reward}$$

- Optimal Value function

state-value $V_{*}(s) = \max_{\pi} V_{\pi}(s) \quad \leftarrow \pi \geq \pi' \text{ if } V_{\pi}(s) \geq V_{\pi'}(s), \forall s$

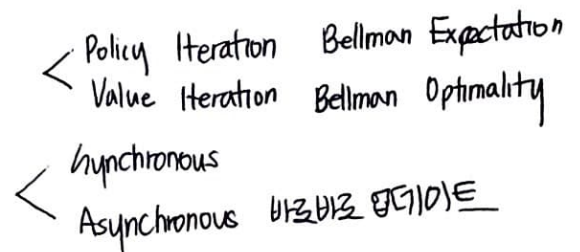
action-value $q_{*}(s, a) = \max_{\pi} q_{\pi}(s, a).$

- Optimal policy can be found by maximising over $q_{*}(s, a)$

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

• DP in MDP

- 대략구조



- Policy Evaluation

evaluate a given policy π

$$V_1 \rightarrow V_2 \rightarrow V_3 \rightarrow \dots \rightarrow V_{\pi}$$

$$\begin{cases} V_{k+1}(s) = \sum_{a \in A} \pi(a|s) \left(R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a V_k(s') \right) \\ V^{k+1} = R^{\pi} + \gamma P^{\pi} V^k \end{cases}$$

처음에는 random policy 배정, k (회수)가 증가할수록 optimal policy로.

Control

Policy Improvement : action 고르기

↳ Greedy Policy Improvement

$$\pi' = \arg \max_{a \in A} q_{\pi}(h, a)$$

앞에 $V_{k+1}(s)$ 를 통해 state value 구하기 완료

현재 state \rightarrow 다음 state
|
action 고르기

$$q_{\pi}(h, a) = R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a V_{\pi}(s')$$

확률에 의존

Value Iteration

evaluation 이 모든 이동가능한 state에 대한 state value 구하기 완료

이 중 max를 취하여 greedy하기 value function을 구하자

$$V_{k+1}(s) = \max_{a \in A} \left(R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a V_k(s') \right)$$

deterministic