6-11

For example, assume we use ϵ**-greedy** method for $\pi_{behavior}$. Q-Learning updates Q-value greedily with regard to Q-value. In other words, Q-Learning uses **greedy** method for π_{target}. Thus, Q-Learning is an off-policy method because it uses **both behavior policy and target policy.**

6-12

In many cases, Q-Learning and Sarsa perform the same way. However, when $S = S'$(present-state and next-state are the same) two method operate differently. Let's consider each flow frow timestep t to timestep $t+1$. (S_t is alerady determined and $S_t = S_{t+1}$)

Sarsa

Select $A_t = \arg\max_a Q(S_t, a)$

Update $Q(S_{t-1}, A_{t-1})$

Take action A_t

Observe R_{t+1}, S_{t+1}

Select

$$\begin{aligned} A_{t+1} &= \arg\max_a Q(S_{t+1}, a) \\ &= \arg\max_a Q(S_t, a) \\ &= A_t \end{aligned}$$

Sequence:

$$\begin{aligned} &S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1} \\ &= S_t, A_t, R_{t+1}, S_t, A_t \end{aligned}$$

Q-Learning

Select $A_t = \arg\max_a Q(S_t, a)$

Update $Q(S_t, A_t)(\rightarrow Q'(S_t, A_t))$

Take action A_t

Observe R_{t+1}, S_{t+1}

Select

$$\begin{aligned} A_{t+1} &= \arg\max_a Q'(S_{t+1}, a) \\ &= \arg\max_a Q'(S_t, a) \\ &\neq A_t \end{aligned}$$

Sequence:

$$\begin{aligned} &S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1} \\ &= S_t, A_t, R_{t+1}, S_t, A_{t+1} \end{aligned}$$