Q-LEARNING

XUESONG LUO
WHAT IS Q-LEARNING

• Q-learning is a model-free reinforcement learning algorithm.
  • Does not require a model of the environment

• For any finite Markov decision process (FMDP), Q-learning finds a policy that is optimal in the sense that it maximizes the expected value of the total reward over any and all successive steps, starting from the current state.
MORE ABOUT Q-LEARNING

• It involves an agent, a set of states $S$, and a set $A$ of actions per state. By performing an action $a \in A$, the agent transitions from state to state. Executing an action in a specific state provides the agent with a reward (a numerical score).

• The goal of the agent is to maximize its total (future) reward. It does this by adding the maximum reward attainable from future states to the reward for achieving its current state, effectively influencing the current action by the potential future reward. This potential reward is a weighted sum of the expected values of the rewards of all future steps starting from the current state.
Q-LEARNING ALGORITHM

\[
Q^{\text{new}}(s_t, a_t) \leftarrow (1 - \alpha) \cdot Q(s_t, a_t) + \alpha \cdot \left( r_t + \gamma \cdot \max_a Q(s_{t+1}, a) \right)
\]

where \( r_t \) is the reward received when moving from the state \( s_t \) to the \( s_{t+1} \), and \( \alpha \) is the learning rate (0<\( \alpha \)<1).

An episode of the algorithm ends when state \( s_{t+1} \) is a final or terminal state. However, Q-learning can also learn in non-episodic tasks. If the discount factor is lower than 1, the action values are finite even if the problem can contain infinite loops.
THE $\gamma$

- WHEN $\gamma = 1$, the future value will not reduce.
- if $\gamma < 1$, the future value will reduce.
- If $\gamma = 0$, there is not future value only focus on current reward.
PSEUDOCODE

```plaintext
Initialize $Q(s, a)$ arbitrarily

Repeat (for each episode):
    Initialize $s$

Repeat (for each step of episode):
    Choose $a$ from $s$ using policy derived from $Q$ (e.g., $\varepsilon$-greedy)
    Take action $a$, observe $r$, $s'$
    $Q(s, a) \leftarrow Q(s, a) + \alpha \left( r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right)$
    $s \leftarrow s'$

until $s$ is terminal
```

Figure 6.12: Q-learning: An off-policy TD control algorithm.