Q-learning: Learn function $Q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$

Require:

States $\mathcal{X} = \{1, \ldots, n_x\}$

Actions $\mathcal{A} = \{1, \ldots, n_a\}$, $A : \mathcal{X} \Rightarrow \mathcal{A}$

Reward function $R : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$

Black-box (probabilistic) transition function $T : \mathcal{X} \times \mathcal{A} \to \mathcal{X}$

Learning rate $\alpha \in [0, 1]$, typically $\alpha = 0.1$

Discounting factor $\gamma \in [0, 1]$

procedure QLearning($\mathcal{X}$, A, R, T, α, γ)

Initialize $Q : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ arbitrarily

while Q is not converged **do**

Start in state $s \in \mathcal{X}$

while s is not terminal **do**

Calculate π according to Q and exploration strategy (e.g. $\pi(x) \leftarrow \arg\max_a Q(x, a)$)

$a \leftarrow \pi(s)$

$r \leftarrow R(s, a)$ ▷ Receive the reward

$s' \leftarrow T(s, a)$ ▷ Receive the new state

$Q(s', a) \leftarrow (1 - \alpha) \cdot Q(s, a) + \alpha \cdot (r + \gamma \cdot \max_{a'} Q(s', a'))$

$s \leftarrow s'$

return Q