Basic RL.3

Judy Tutorial

So far, we know

$$\boldsymbol{\pi}(\mathbf{a}|\boldsymbol{s}), \boldsymbol{\pi}^*(\boldsymbol{a}|\boldsymbol{s})$$

$$\boldsymbol{V}_{\pi}(\boldsymbol{s}), \boldsymbol{Q}_{\pi}(\boldsymbol{s}, \boldsymbol{a}), \boldsymbol{V}^*(\boldsymbol{s}), \boldsymbol{Q}^*(\boldsymbol{s}, \boldsymbol{a})$$

But, how do we learn?

Recall Discounted Return

$$R_t \triangleq r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} \dots$$

$$= r_t + \gamma(r_{t+1} + \gamma^1 r_{t+2} + \gamma^2 r_{t+3} \dots)$$

$$= r_t + \gamma R_{t+1}$$

Recall $V_\pi(s)$

$$
\begin{aligned}
V_\pi(s) &\triangleq \mathbb{E}_\pi[R_t | s_t = s] \\
&= \mathbb{E}_\pi[r_t + \gamma R_{t+1} | s_t = s] \\
&= \sum_a \pi(a|s) \sum_{s'} p(s'|s,a) \left[r_t + \gamma \mathbb{E}_\pi[R_{t+1} | s_{t+1} = s'] \right] \\
&= \sum_a \pi(a|s) \sum_{s'} p(s'|s,a) \left[r_t + \gamma V_\pi(s') \right] \\
&= \underline{\mathbb{E}_\pi[r_t + \gamma V_\pi(s') | s_t = s]} \qquad \forall s \in \mathcal{S}
\end{aligned}
$$

Bellman Equation

Recall $Q_\pi(s,a)$

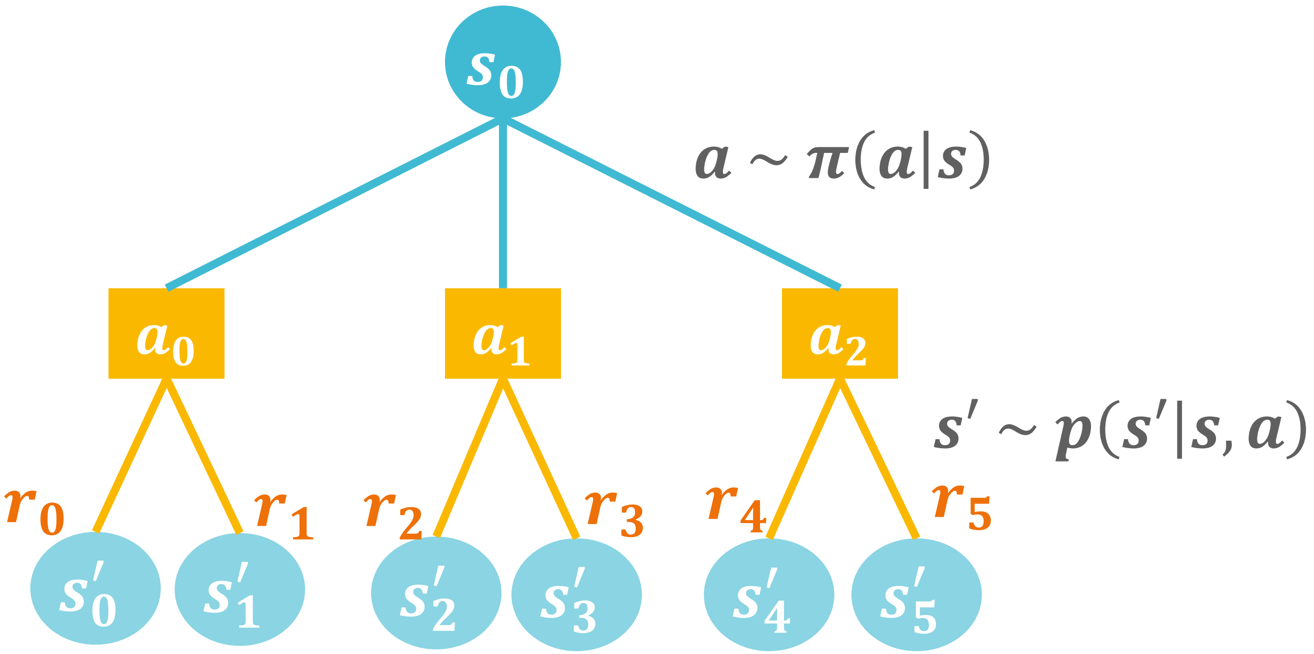
$$
\begin{aligned}
&Q_\pi(s,a)\\
&\quad \triangleq \mathbb{E}_\pi[R_t|s_t=s,a_t=a]\\
&\quad = \mathbb{E}_\pi[r_t+\gamma V_\pi(s')|s_t=s,a_t=a]\\
&\quad = \sum_{s'} p(s'|s,a)\left[r_t+\gamma \mathbb{E}_{a'\sim\pi}[Q_\pi(s',a')|s_{t+1}=s',a_{t+1}=a']\right]\\
&\quad = \sum_{s'} p(s'|s,a)\cdot\\
&\qquad \left[r_t+\gamma \sum_{a'} \pi(a'|s')\, Q_\pi(s',a')|s_{t+1}=s',a_{t+1}=a'\right]
\end{aligned}
$$

$\forall s\in\mathcal{S}, a\in\mathcal{A}$

Bellman Equation

Diagram

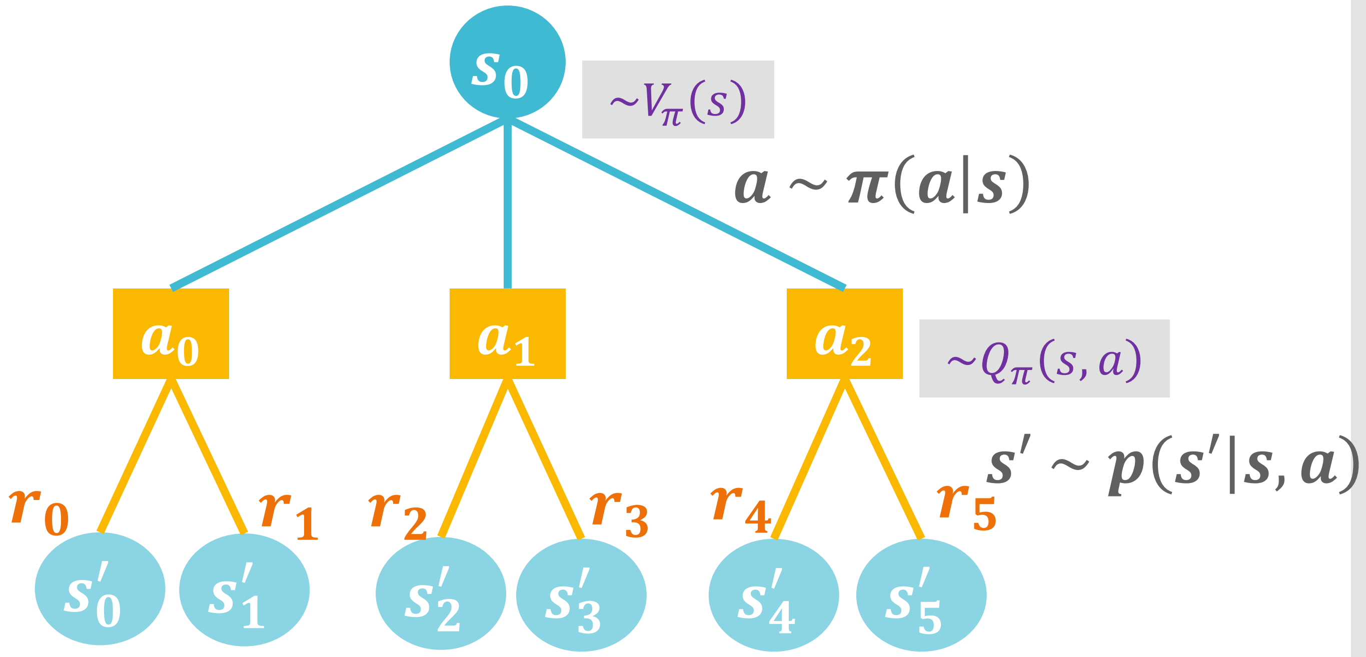
Given one transition: (s, a, r, s')



$$
\begin{aligned}
V_\pi(s) &= \mathbb{E}_\pi[r_t + \gamma V_\pi(s') | s_t = s] \\
&= \sum_a \pi(a|s) \sum_{s'} p(s'|s,a) \, [r + \gamma V_\pi(s')]
\end{aligned}
$$

Diagram

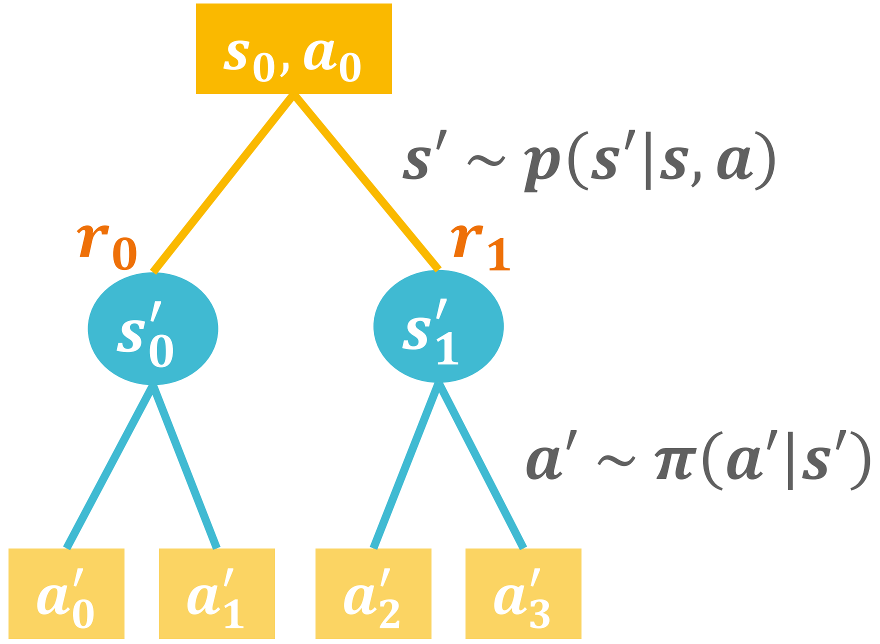
Given one transition: (s, a, r, s')



$$\begin{aligned} V_\pi(s) &= \mathbb{E}_\pi[Q_\pi(s,a)] \\ &= \sum_a \pi(a|s) Q_\pi(s,a) \end{aligned}$$

Diagram

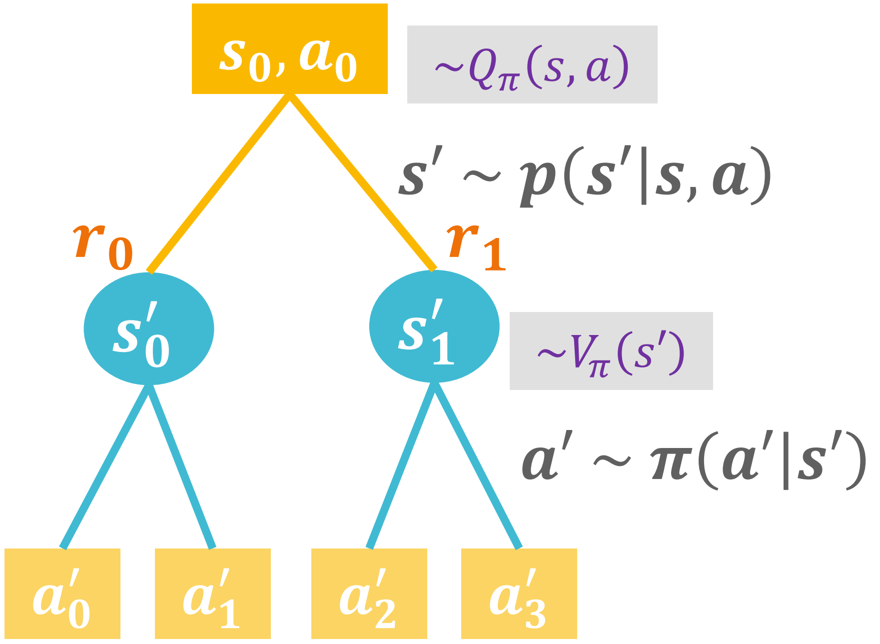
Given one transition: (s, a, r, s', a')



$$Q_\pi(s,a) = \sum_{s'} p(s'|s,a) \cdot$$
$$\left[r_t + \gamma \sum_{a'} \pi(a'|s')\, Q_\pi(s',a') | s_{t+1} = s', a_{t+1} = a'\right]$$

Diagram

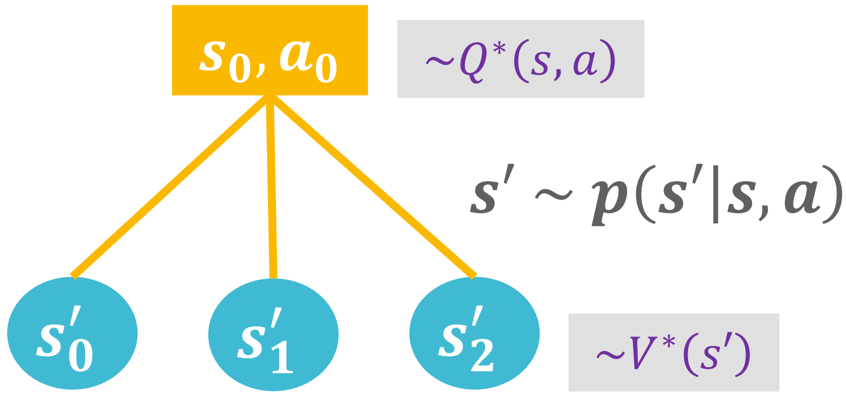
Given one transition: (s, a, r, s', a')



$$Q_\pi(s,a) = \mathbb{E}[r_t + \gamma V_\pi(s_{t+1}) | s_t = s, a_t = a]$$
$$= \sum_{s'} p(s'|s,a)\,[r + \gamma V_\pi(s')]$$

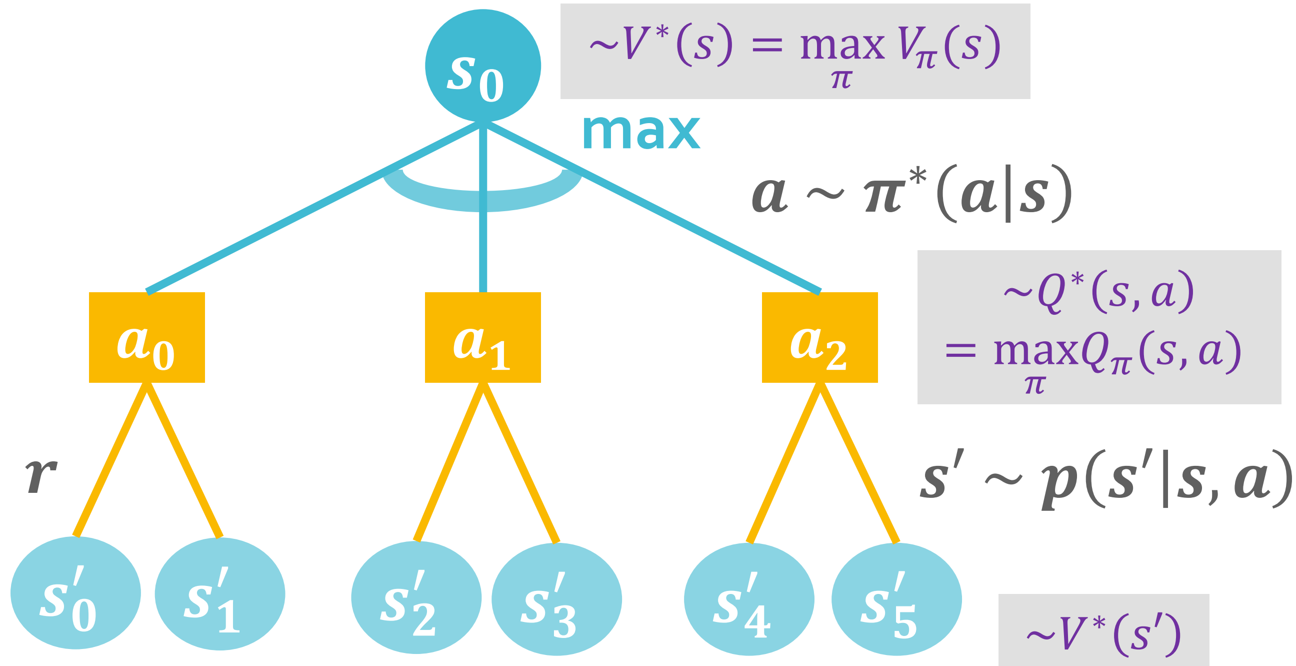
Recall
$\mathrm{V}^*(s)$
$Q^*(s,a)$

$$V^*(s) \triangleq \max_\pi V_\pi(s), \text{ for all } s \in \mathcal{S}$$
$$Q^*(s,a) \triangleq \max_\pi Q_\pi(s,a), \text{ for all } s \in \mathcal{S}, a \in \mathcal{A}$$



$$Q^*(s,a) = \mathbb{E}[r_t + \gamma V^*(s_{t+1}) | s_t = s, a_t = a]$$
$$= \sum_{s'} p(s'|s,a)\,[r + \gamma V^*(s')]$$

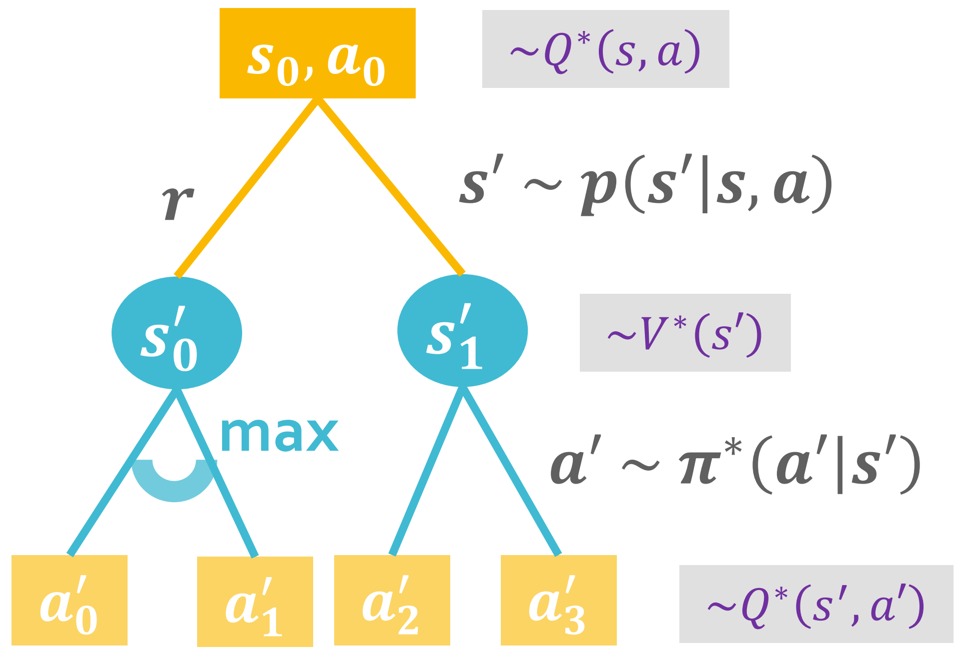
Diagram



$$
\begin{aligned}
V^*(s) &= \max_{a \in \mathcal{A}} Q_{\pi^*}(s,a) \\
&= \max_{a} \mathbb{E}[r + \gamma V^*(s')]
\end{aligned}
$$

Bellman Optimality Equation

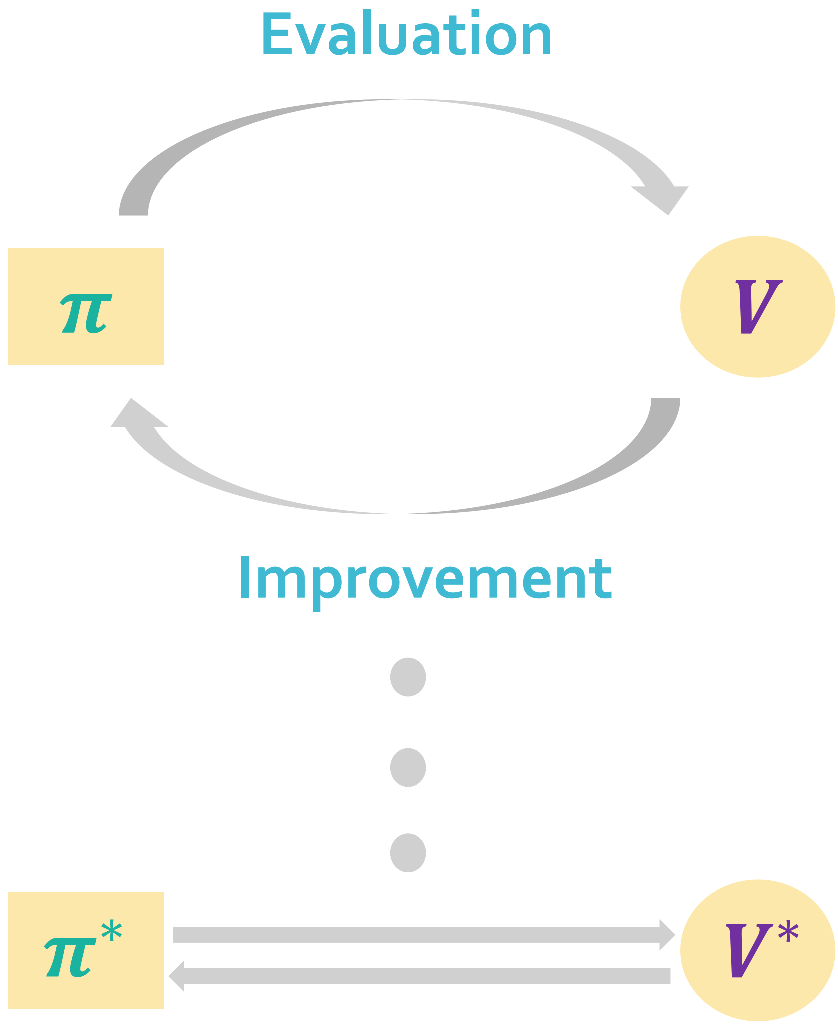
Diagram



$$
\begin{aligned}
Q^*(s,a) &= \mathbb{E}\left[r_t + \gamma \max_{a'} Q^*(s_{t+1}, a') | s_t = s, a_t = a\right] \\
&= \underline{\mathbb{E}\left[r + \gamma \max_{a'} Q^*(s', a')\right]}
\end{aligned}
$$

Bellman Optimality Equation

Generalized Policy Iteration



Policy Evaluation

Calculate $V(s)$ or $Q(s, a)$ functions for certain π

- $V_{k+1}(s) = \mathbb{E}_{\pi}[r + \gamma V_k(s')], k \longrightarrow \infty$

Or

- $V_{k+1}(s) = \max_{a} \mathbb{E}_{\pi}[r + \gamma V_k(s')], k \longrightarrow \infty$

Policy Improvement

Improve π with respect to $V(s)$ or $Q(s,a)$ by making it greedy

- $\pi'(s) \triangleq \text{argmax}_a Q_\pi(s,a)$
- $= \text{argmax}_a \mathbb{E}[r + V_\pi(s')]$

Now we can talk about

Q-learning!

Q-learning

[Watkins, 1989]

Recall Bellman Optimality Equation

$$Q^*(s,a) = \mathbb{E}\left[r + \gamma \max_{a'} Q^*(s',a')\right]$$

Target

$$Q(s,a) \leftarrow Q(s,a) + \alpha\left[\underbrace{r + \gamma \max_{a'} Q(s',a') - Q(s,a)}_{\text{Temporal Difference error } \delta}\right]$$

Step size or
Learning rate

Learning Flow

Initialization can severely change the learning performance

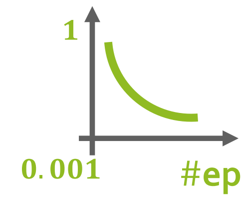
Hyperparameters (need to tune a bit)

- Initialize Q with size $|\mathcal{S}|\times|\mathcal{A}|$, specify α, γ, ϵ

- For each episode:
- s = env.reset ( ) Initialization of the env
- For each step:
- $a \sim \epsilon$-greedy (s, ϵ)
- $s', r, done$ = env.step (a)

Agent takes an action
Env gives feedbacks

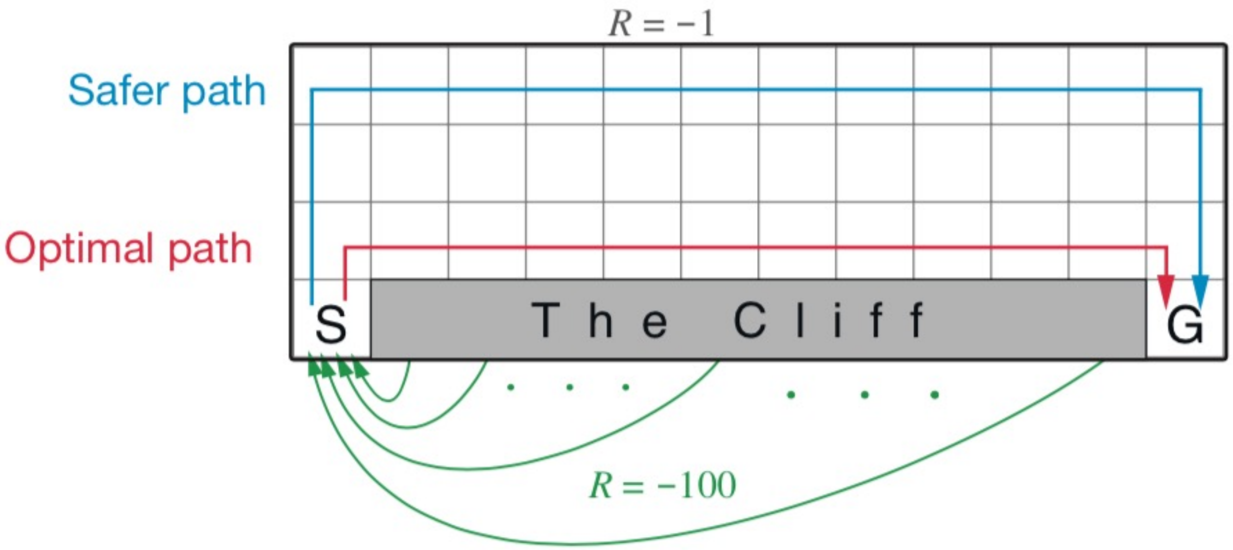
a probability $\epsilon \in [0, 1]$
With ϵ, random action
With $1-\epsilon$, $\mathrm{argmax}_a Q(s, a)$



- $$Q(s,a) \leftarrow Q(s,a) + \alpha\left[r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right]$$
- $s = s'$
- until $done$

Update Q based on s, a, s', r

Cliff Walking



- What's the state space?
- What's the action space?
- What's the size of Q ?
- How to represent initial state and terminal state?

- $\mathcal{S}$: 4 x 12
- $\mathcal{A}$: $\uparrow, \downarrow, \leftarrow, \rightarrow$
- s_0: (3,0)
- s_f: (3,11)

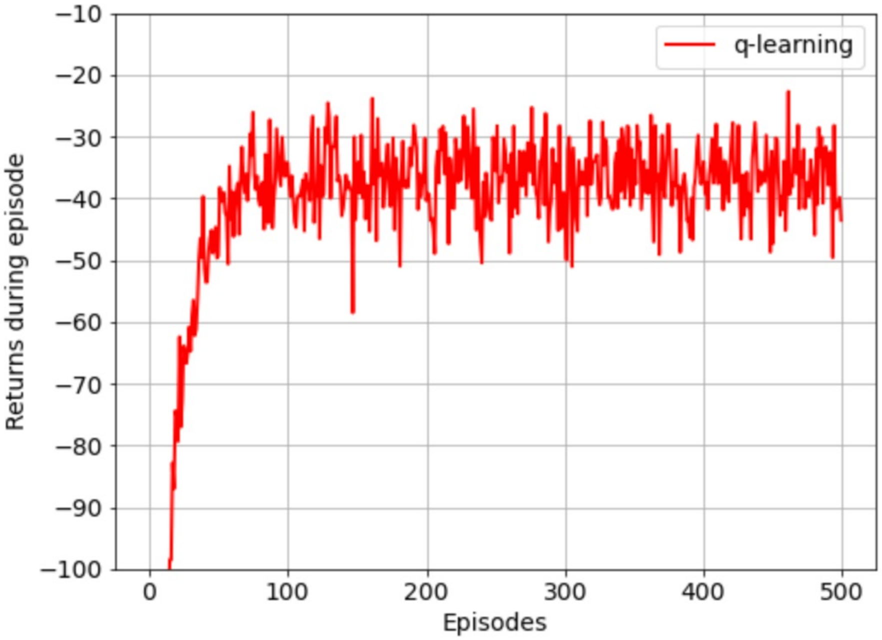
Cliff Walking in python

```
Q=np.zeros((n_rows,n_cols,n_a))

stpCnt=0

for ep in range(n_eps):

    r_sum,done=0,False
    s=START

    for stp in range(n_stps):

        a=e_greedy(eps,Q[s[0],s[1]])
        s_,r,done=step(s,a)
        delta=r+gm*np.max(Q[s_[0],s_[1]])-Q[s[0],s[1],a]
        Q[s[0],s[1],a]+=lr*delta

        s=s_
        r_sum+=r
        stpCnt+=1

        if done:
            break
```

Q-learning result

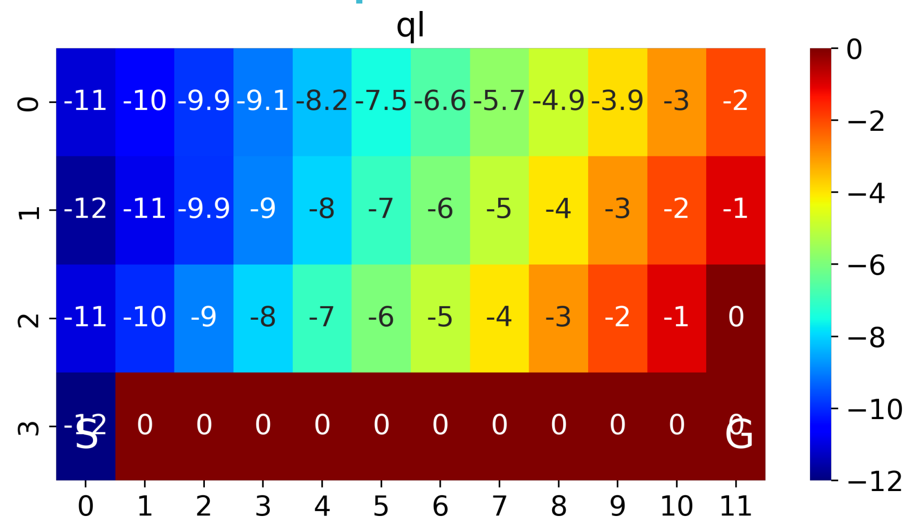
Learning curve

$\alpha = 0.5$
$\gamma = 1$
$\epsilon = 0.1$

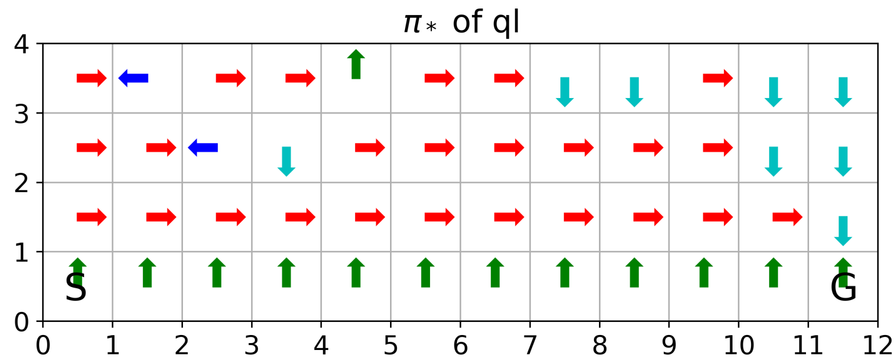


Q-learning result

Heatmap of learned V



Optimal Policy



Questions

- What are the policy evaluation and the policy improvement procedures in Q-learning?
- Which factor makes the Q-learning agent always choose the ‘dangerous’/’optimal’ path?
- What will happen if we anneal ϵ?

Reference & Code

- Sutton and Barto 2nd Edition, Example 6.6
- https://ha5ha6.github.io/judy_blog/td/#cliff-walking