Q-learning

Биршерт Алексей

Февраль 2020

Содержание

- Марковский процесс принятия решений
- Равенство Беллмана
- on-policy и off-policy
- Q-learning
- Deep Q-learning
- Experience replay
- вопросы

Марковский процесс принятия решений

Взаимодействие агента и среды

Общее представление

- Агент совершает действие $A_t \in A(s)$
- Агент получает награду $R_{t+1} \in R \subset \mathbb{R}$, новое состояние $S_{t+1} \in S$
- $S_0, A_0, R_1, S_1, \underbrace{A_1, R_2, S_2}_{\text{элемент цикла}}, A_2, R_3, S_3, A_3 \ldots$

Вероятности

- $p(s', r \mid s, a) \doteq Pr\left(S_t = s', R_t = r \mid S_{t-1} = s, A_{t-1} = a\right)$
- $\sum\limits_{s' \in S} \sum\limits_{r \in R} p(s', r \mid s, a) = 1$ для всех $s \in S,\ a \in A(s)$

Марковский процесс принятия решений

Взаимодействие агента и среды

Мем

Марковский процесс принятия решений

Ожидаемая награда агента

- $G_t \doteq R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}, \quad 0 \leq \gamma \leq 1$
- $G_t = R_{t+1} + \gamma G_{t+1}$
- γ определяет насколько важна будущая награда сейчас
- Если $\gamma = 0$, то важна только награда на текущем шагу, если $\gamma = 1$, то все важны равнозначно

Policy функция

$\pi(a \mid s) \doteq Pr(A_t = a \mid S_t = s), \quad a \in A(s), s \in S$

Задаёт вероятностное распределение возможных действий в конкретном состоянии

State-value функция

$$v_\pi(s) \doteq \mathop{\mathbb{E}}_{\pi}[G_t|S_t = s] = \mathop{\mathbb{E}}_{\pi}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1}|S_t = s\right]$$

Задаёт ожидаемую награду, если мы стартуем из состояния s и следуем policy π

Action-value функция

$$q_\pi(s,a) \doteq \mathop{\mathbb{E}}_{\pi}[G_t|S_t = s, A_t = a] = \mathop{\mathbb{E}}_{\pi}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1}|S_t = s, A_t = a\right]$$

Задаёт ожидаемую награду, если стартуем из состояния s и совершаем действие a, и потом следуем policy π

Равенство Беллмана

Связь состояния и последующих

$$
\begin{aligned}
v_\pi(s) &\doteq \mathop{\mathbb{E}}_{\pi}\left[G_t | S_t = s\right] \\
&= \mathop{\mathbb{E}}_{\pi}\left[R_{t+1} + \gamma G_{t+1} | S_t = s\right] \\
&= \sum_a \pi(a|s) \sum_{s'} \sum_r p\left(s', r | s, a\right) \left[r + \gamma \mathbb{E}_\pi \left[G_{t+1} | S_{t+1} = s'\right]\right] \\
&= \sum_a \pi(a|s) \sum_{s', r} p\left(s', r | s, a\right) \left[r + \gamma v_\pi\left(s'\right)\right]
\end{aligned}
$$

Равенство Беллмана

Оптимальность

Оптимальная policy функция

- $\pi \geq \pi' \Leftrightarrow v_\pi(s) \geq v_{\pi'}(s), \quad \forall s \in S$
- Оптимальная π - та, которая лучше или равна всем остальным
- Оптимальной policy функции соответствуют оптимальные state-value и action-value функции
- $v_*(s) \doteq \max\limits_\pi v_\pi(s),\ q_*(s,a) \doteq \max\limits_\pi q_\pi(s,a)$

Оптимальная state-value функция

$$
\begin{aligned}
v_*(s) &= \max_{a \in \mathcal{A}(s)} q_{\pi_*}(s, a) \\
&= \max_a \mathbb{E}_{\pi_*} \left[G_t | S_t = s, A_t = a \right] \\
&= \max_a \mathbb{E}_{\pi_*} \left[R_{t+1} + \gamma G_{t+1} | S_t = s, A_t = a \right] \\
&= \max_a \mathbb{E} \left[R_{t+1} + \gamma v_* \left(S_{t+1} \right) | S_t = s, A_t = a \right] \\
&= \max_a \sum_{s', r} p \left(s', r | s, a \right) \left[r + \gamma v_* \left(s' \right) \right]
\end{aligned}
$$

Ожидаемая награда у состояния при оптимальной policy должна быть равна ожидаемой награде лучшего действия из этого состояния.

Равенство Беллмана

Оптимальность

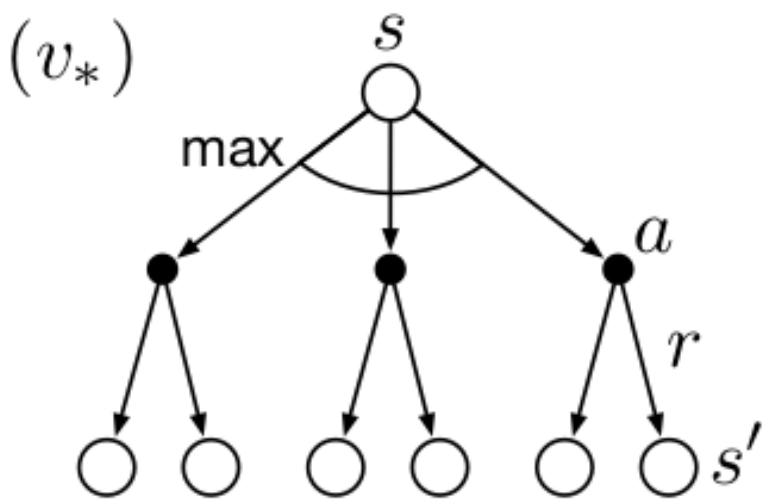
Оптимальная action-value функция

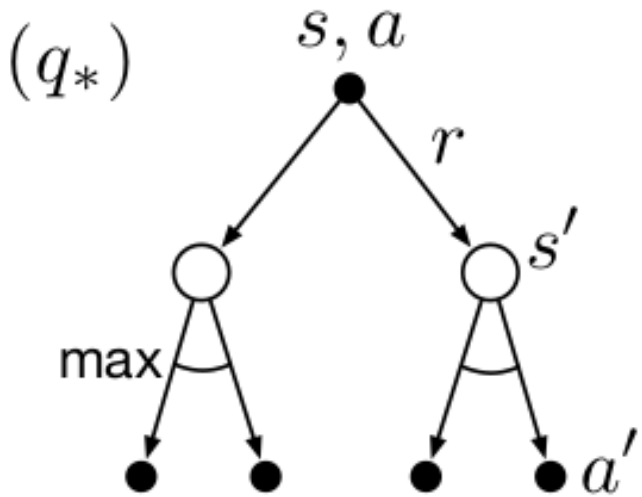
$$q_*(s,a) = \mathbb{E}\left[R_{t+1} + \gamma \max_{a'} q_*\left(S_{t+1}, a'\right) | S_t = s, A_t = a\right]$$

$$= \sum_{s',r} p\left(s', r | s, a\right)\left[r + \gamma \max_{a'} q_*\left(s', a'\right)\right]$$

Равенство Беллмана

Оптимальность





on-policy и off-policy методы

on-policy

Пытаются оценить или оптимизировать policy функцию, в соответствии с которой действуют сами.

off-policy

Используют две различные policy функции - behavior и target policy, первая регулирует их исследование МППР, вторая оптимизируется в процессе исследования.

SARSA, on-policy

$$Q(S_t, A_t) \leftarrow (1-\alpha)Q(S_t, A_t) + \alpha\left[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1})\right]$$

Q-learning, off-policy

$$Q(S_t, A_t) \leftarrow (1-\alpha)Q(S_t, A_t) + \alpha\left[R_{t+1} + \gamma \max_a Q(S_{t+1}, a)\right]$$

Q-learning

Алгоритм, $Q \approx q_*$

Инициализация $Q(\cdot\,,\cdot)$ произвольно

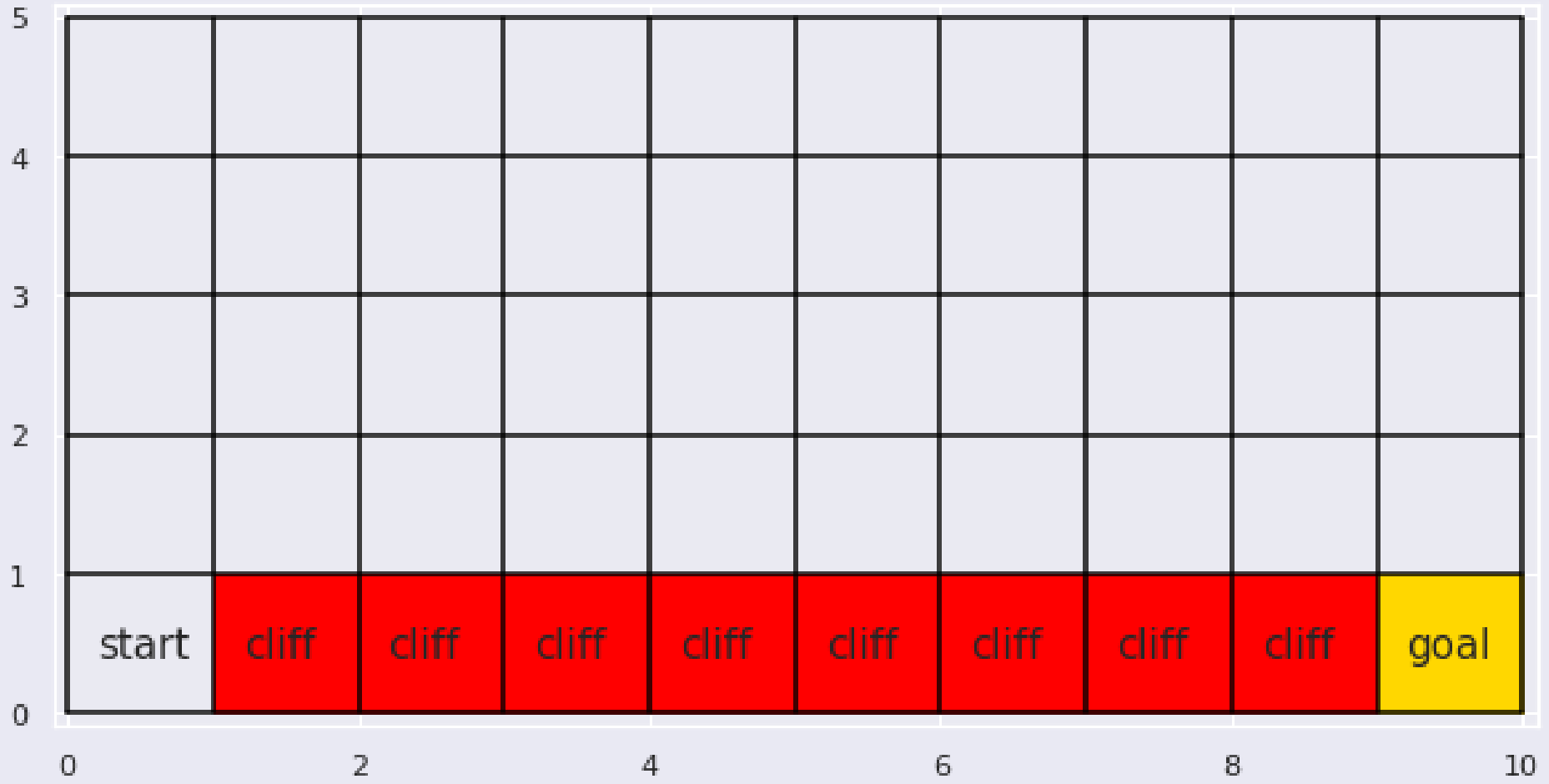
Цикл:

- Инициализировать s
- Цикл:
  - Выбрать шаг a из s с помощью behavior policy ($\varepsilon - greedy$, UCB)
  - Сделать шаг a, получить r и s'
  - $Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha\left[r + \max\limits_{a'}\gamma Q(s',a')\right]$
  - $s \leftarrow s'$

Q-learning

Сравнение с SARSA на основе задачи cliff walking

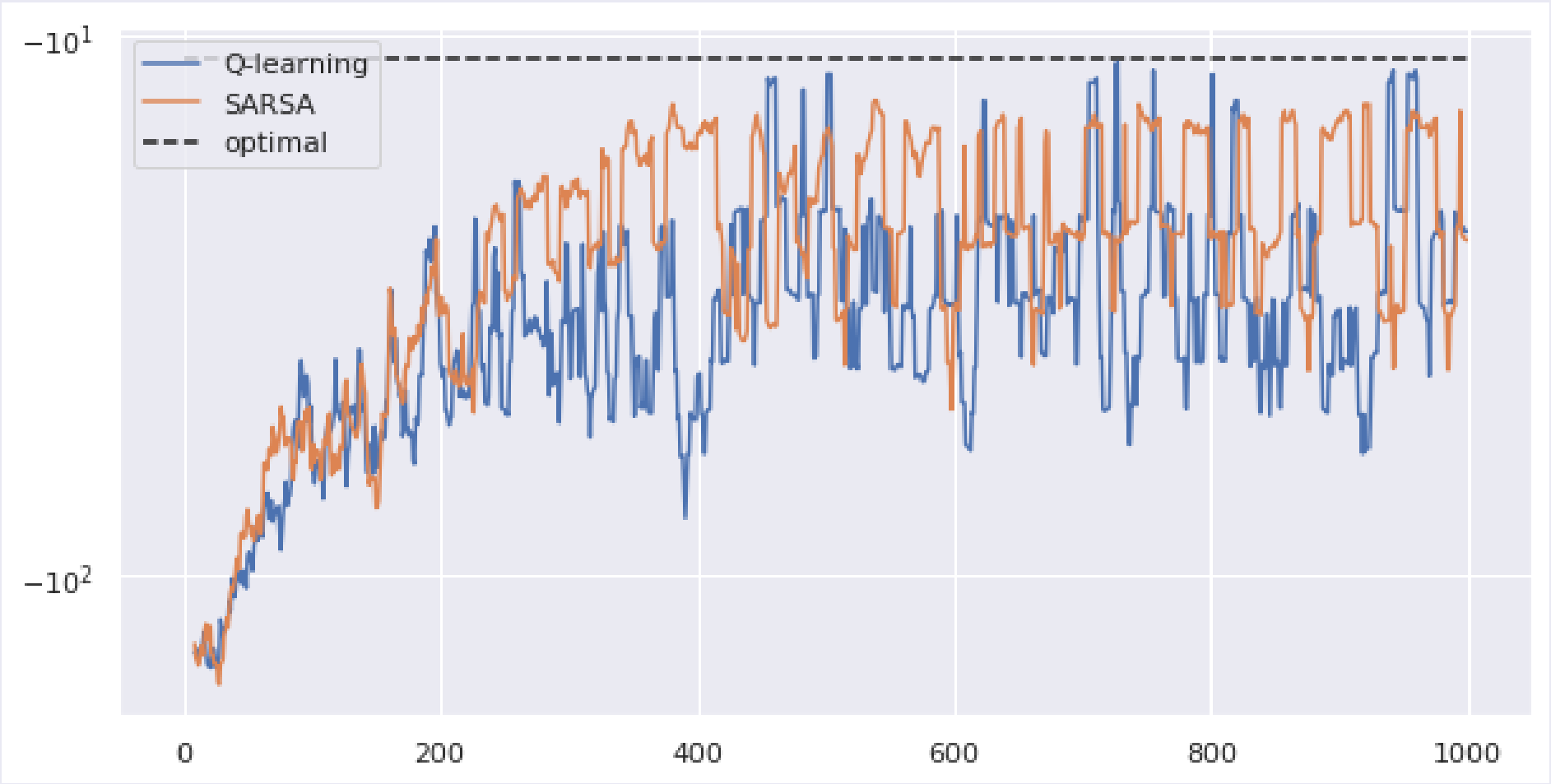
Постановка задачи



Q-learning

Сравнение с SARSA на основе задачи cliff walking

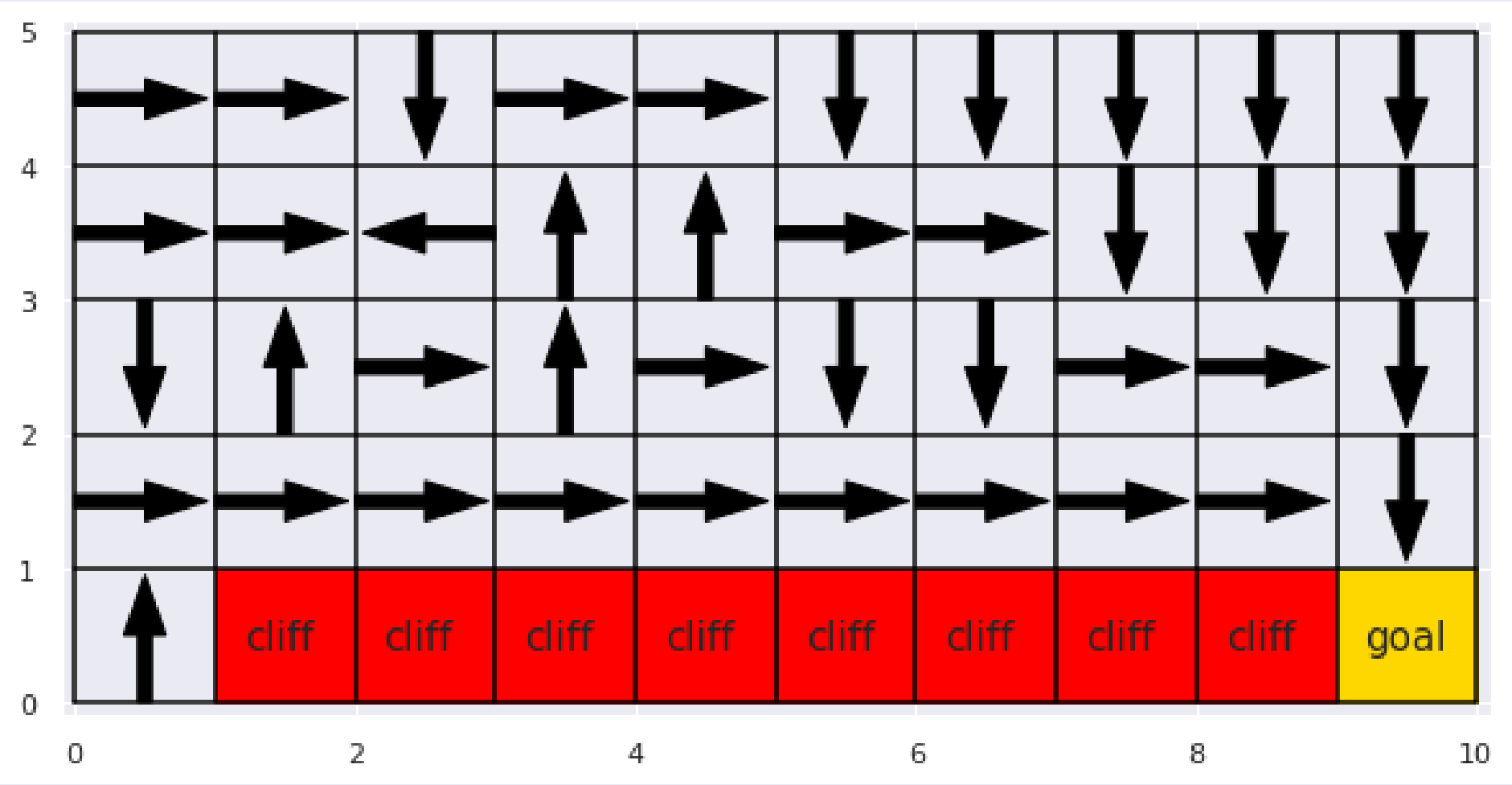
Обучение, $\varepsilon = 0.1$



Q-learning

Сравнение с SARSA на основе задачи cliff walking

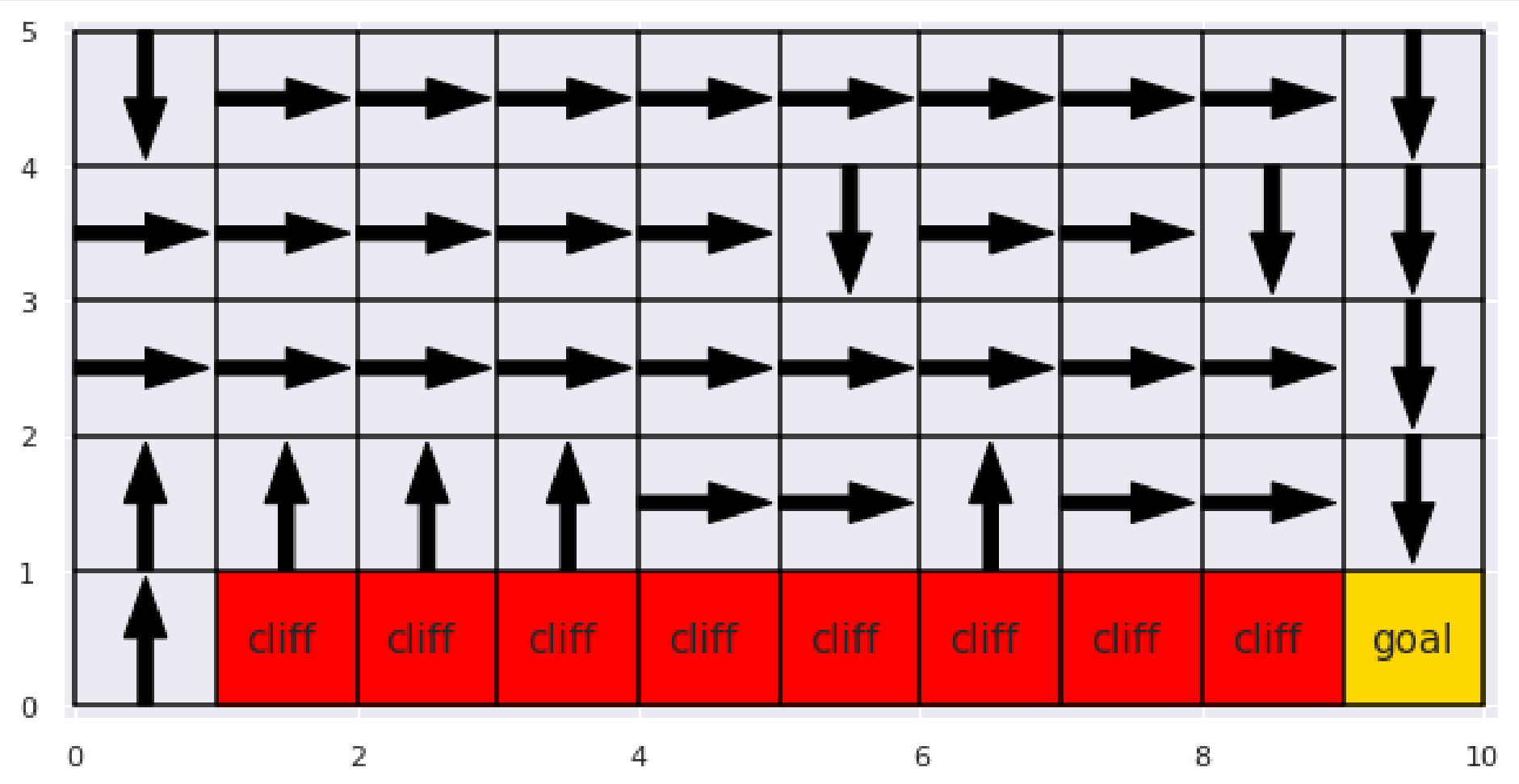
Q-learning, $\varepsilon = 0.1$



Q-learning

Сравнение с SARSA на основе задачи cliff walking

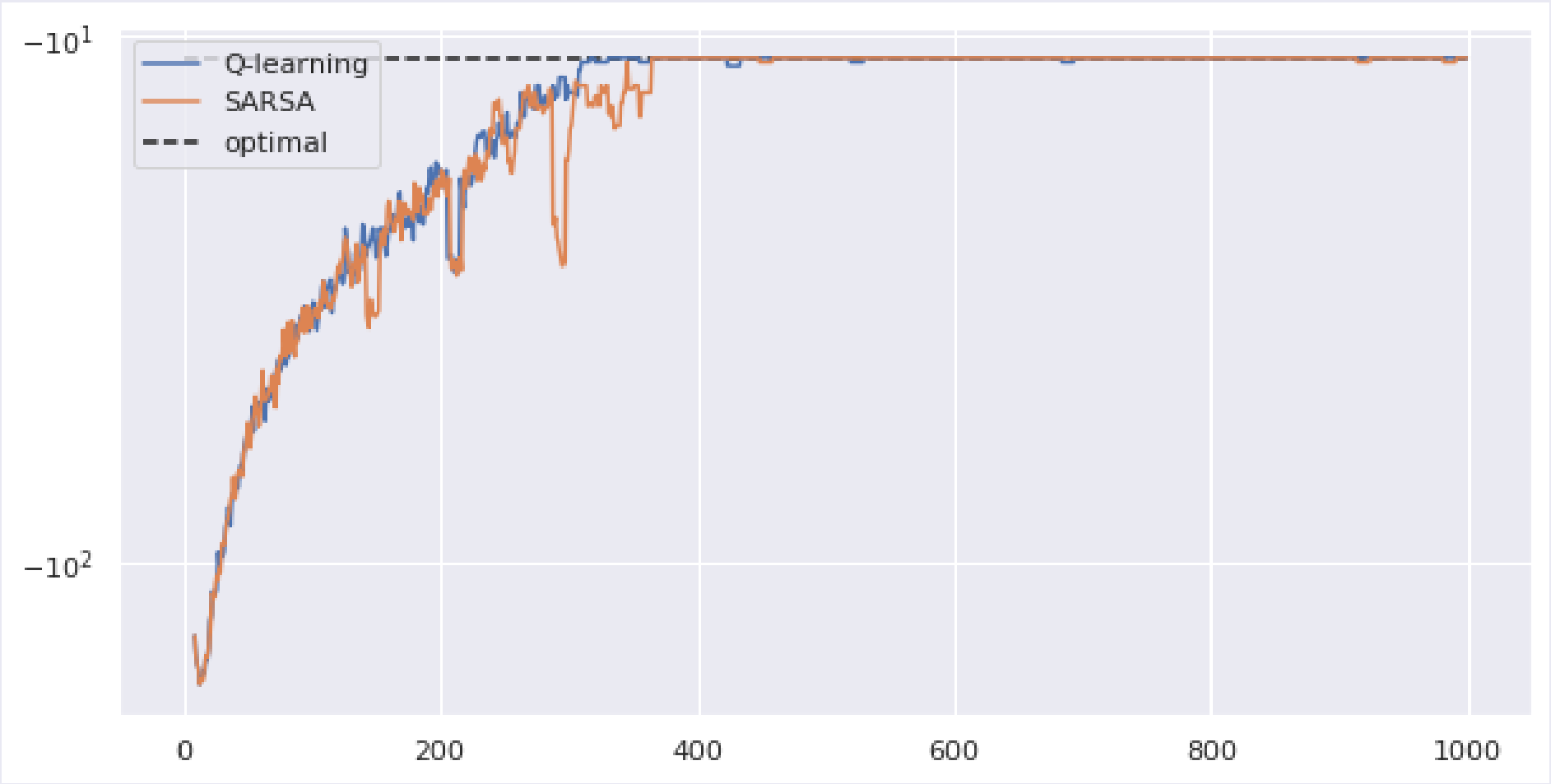
SARSA, $\varepsilon = 0.1$



Q-learning

Сравнение с SARSA на основе задачи cliff walking

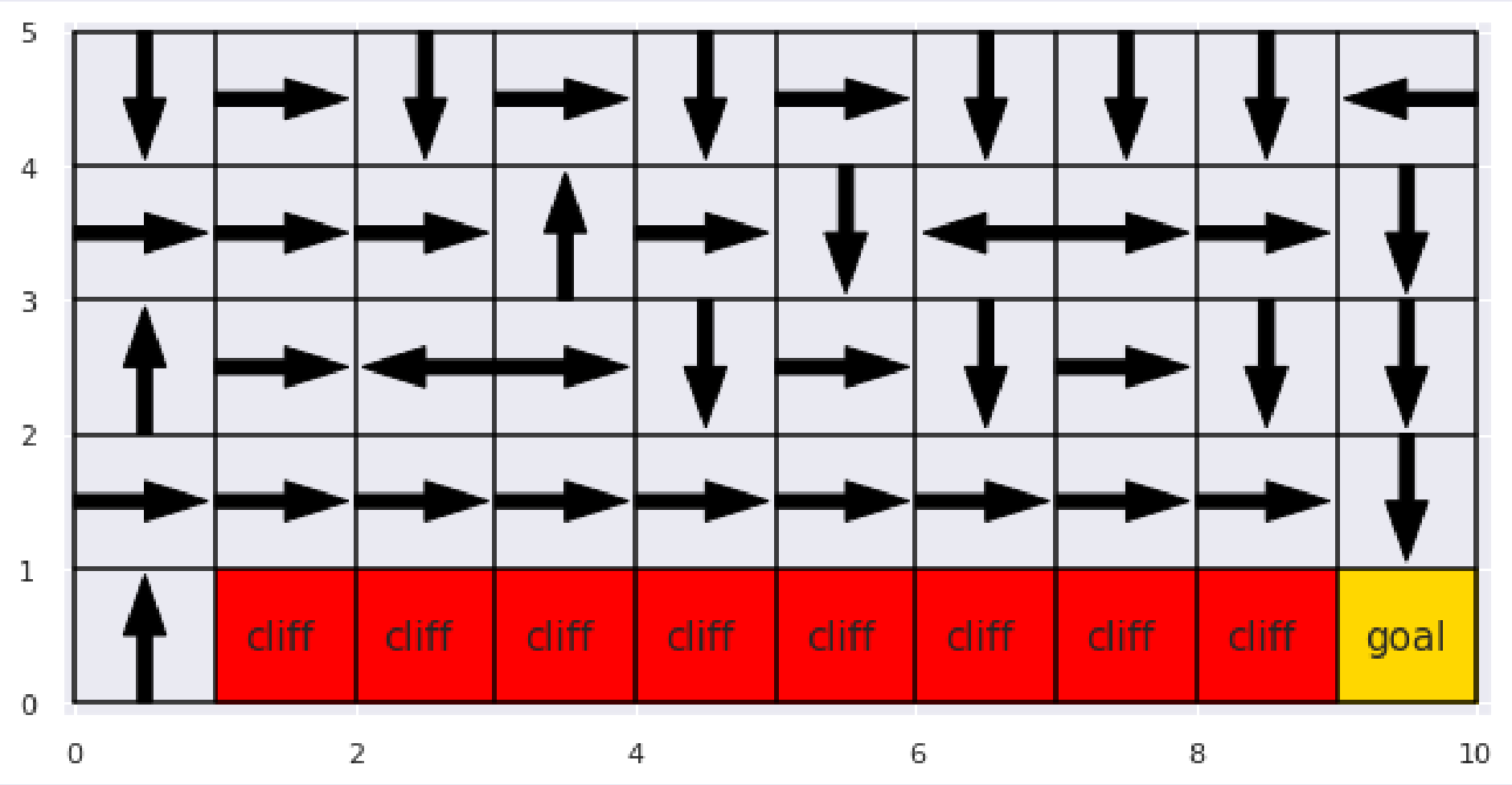
Обучение, $\varepsilon = 0.001$



Q-learning

Сравнение с SARSA на основе задачи cliff walking

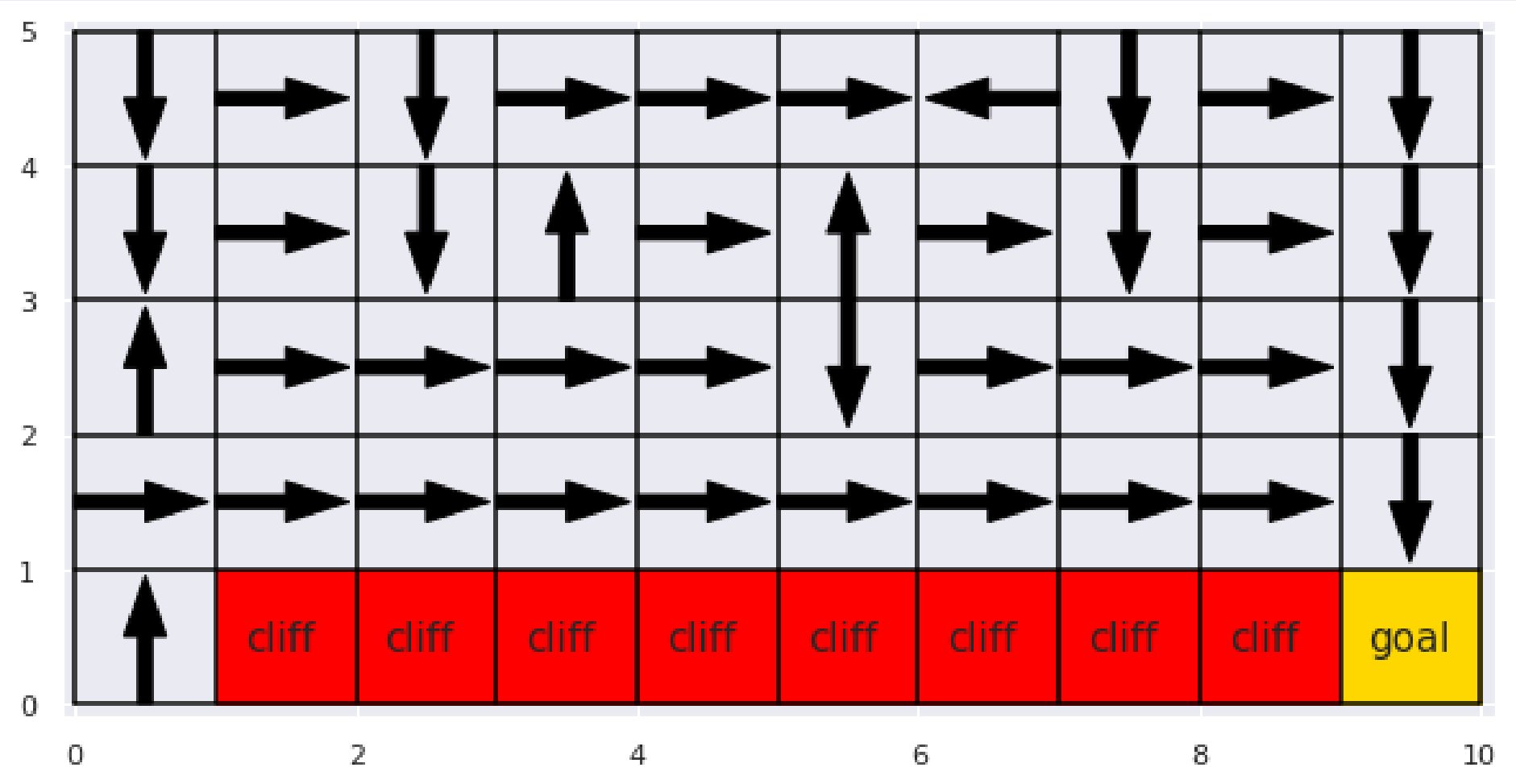
Q-learning, $\varepsilon = 0.001$



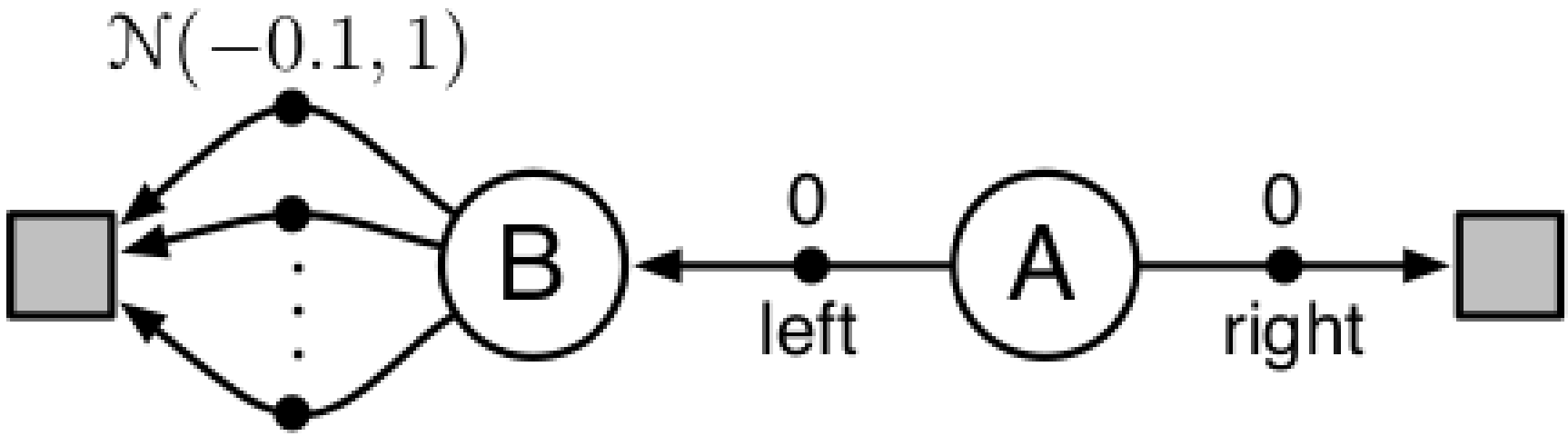
Q-learning

Сравнение с SARSA на основе задачи cliff walking

SARSA, $\varepsilon = 0.001$



Максимизация смещения



Алгоритм, $Q_1 \approx Q_2 \approx q_*$

Инициализация $Q_1(\cdot\,,\cdot)$ и $Q_2(\cdot\,,\cdot)$ произвольно

Цикл:

- Инициализировать s
- Цикл:
  - Выбрать шаг a из s с помощью $\varepsilon - greedy$ из $Q_1 + Q_2$
  - Сделать шаг a, получить r и s'
  - С вероятностью 0.5:

$$Q_1(s,a) \leftarrow (1-\alpha)Q_1(s,a) + \alpha\left[r + \gamma Q_2(s', \arg\max_{a'} Q_1(s',a'))\right]$$

  - Иначе:

$$Q_2(s,a) \leftarrow (1-\alpha)Q_2(s,a) + \alpha\left[r + \gamma Q_1(s', \arg\max_{a'} Q_2(s',a'))\right]$$

  - $s \leftarrow s'$

Deep Q-learning

Интуиция позади использования DQN, $Q(s, a, \theta) \approx Q_*(s, a)$

$Q-$learning имеет проблемы со сходимостью в пространствах больших размерностей и не может обобщить предыдущий опыт для новых ситуаций

$Q\left(S_t, A_t, \theta\right) \leftarrow (1-\alpha) Q\left(S_t, A_t, \theta\right) + \alpha \left[R_{t+1} + \gamma \max\limits_a Q\left(S_{t+1}, a, \theta\right)\right]$, θ - веса нейронной сети

$Q\left(S_t, A_t, \theta\right)$ - предсказание

$R_{t+1} + \gamma \max\limits_a Q\left(S_{t+1}, a, \theta\right)$ - целевая переменная

$R_{t+1} + \gamma \max\limits_a Q\left(S_{t+1}, a, \theta\right) - Q\left(S_t, A_t, \theta\right)$ - ошибка

Deep Q-learning

Интуиция позади использования DQN, $Q(s, a, \theta) \approx Q_*(s, a)$

$$Q\left(S_t, A_t, \theta\right) \leftarrow (1-\alpha) Q\left(S_t, A_t, \theta\right) + \alpha \left[R_{t+1} + \gamma \max_a Q\left(S_{t+1}, a, \theta\right)\right], \ \theta \text{ -}$$

веса нейронной сети

$$L(\theta) = \mathop{\mathbb{E}}_{s,a,r} \left[\left(\mathop{\mathbb{E}}_{s'} \left[r + \gamma \max_{a'} Q\left(s', a', \theta\right) | s, a \right] - Q\left(s, a, \theta\right) \right)^2 \right]$$

$$\Delta_\theta L(\theta) = \mathop{\mathbb{E}}_{s,s',a,r} \left[\left(r + \gamma \max_{a'} Q\left(s', a', \theta\right) - Q\left(s, a, \theta\right) \right) \Delta_\theta Q\left(s, a, \theta\right) \right]$$

Deep Q-learning

Проблемы

Однако такая нейронная сеть может не сойтись:

- Корреляция в последовательности наблюдений
  - Решение - experience replay
- Корреляция между целевой переменной и предсказанием
  - Решение - fixed Q targets

Experience replay

Решение проблемы с корреляцией в последовательности наблюдений

Используется буфер, в котором хранятся последние n наблюдений.

Нейронная сеть обучается на мини-батчах, сэмплируемых из буфера.

Преимущества:

- Снижает влияние отдельных наблюдений на обучение
- Убирает корреляцию в обучающем батче, что позволяет нейросети сходиться лучше
- Повышает скорость обучения
- Позволяет обучать параллельно

Вопросы

- В чем заключается bellman optimality equation?
- В чем заключаются off-policy и on-policy methods? Почему Q-learning считается off-policy method?
- Что такое experience replay?

Список литературы

1. Richard S. Sutton and Andrew G. Barto. Reinforcement Learning: An Introduction. 2018
2. Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg & Demis Hassabis. Human-level control through deep reinforcement learning. 2015