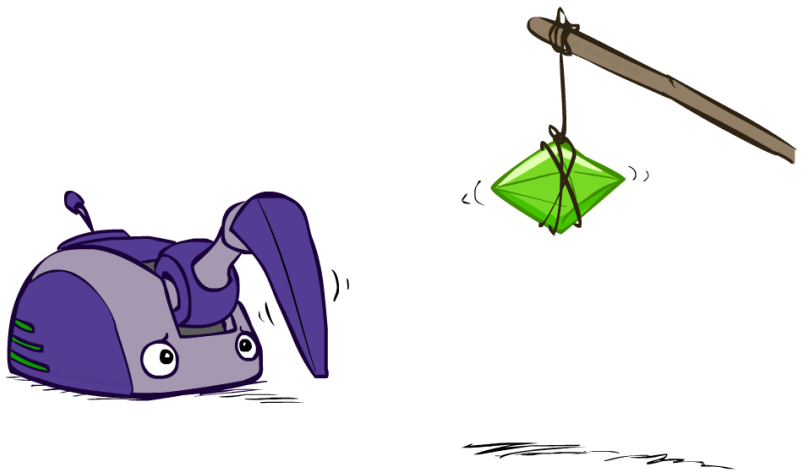
یادگیری تقویتی
(Reinforcement Learning)



مثال: سن مورد انتظار

هدف: محاسبه‌ی سن مورد انتظار دانشجویان کلاس.

P(A) شناخته شده

$$E[A] = \sum_a P(a) \cdot a \qquad = 0.35 \times 20 + \ldots$$

بدون داشتن P(A) باید نمونه برداری انجام شود $[a_1, a_2, \ldots a_N]$

P(A) ناشناخته: "مستقل از مدل"

$$E[A] \approx \frac{1}{N} \sum_i a_i$$

چرا این روش جواب می‌دهد؟ چون نمونه‌ها با فراوانی (تعداد تکرار) درست ظاهر می‌شوند.

P(A) ناشناخته: "مبتنی بر مدل"

$$\hat{P}(a) = \frac{\text{num}(a)}{N}$$

$$E[A] \approx \sum_a \hat{P}(a) \cdot a$$

چرا این روش جواب می‌دهد؟ چون در نهایت مدل درست یاد گرفته می‌شود.

یادگیری تقویتی -- مرور کلی

❑ یادگیری تقویتی منفعل (Passive RL) :

یاد گرفتن از تجربه‌ها (عامل با یک سیاست از پیش‌مشخص‌شده حرکت می‌کند و هدف آن ارزیابی ارزش حالت‌ها است، نه تصمیم‌گیری)

1. **یادگیری مبتنی بر مدل :**
   1. عامل ابتدا مدل MDP (یعنی T و R) را بر پایه‌ی تجربه‌های خود تخمین می‌زند. سپس با استفاده از این مدل تقریبی، ارزش حالت‌ها را محاسبه می‌کند.
2. **یادگیری مستقل از مدل:**

عامل فقط بر پایه‌ی تجربه، بدون دانستن مدل، ارزش حالت‌ها را محاسبه می‌کند.(مستقیماً ارزش هر حالت را یاد می‌گیرد)

Value Learning

1. **ارزیابی مستقیم:** محاسبه‌ی میانگین مجموع پاداش‌ها در اپیزودها
2. **یادگیری تفاضل زمانی:** به‌روزرسانی تدریجی ارزش هر حالت با استفاده از ارزش حالت بعدی

❑ یادگیری تقویتی فعال (Active RL):

عامل خودش باید تصمیم بگیرد و تجربه جمع‌آوری کند (عامل دیگر فقط یک سیاست را دنبال نمی‌کند، بلکه باید خودش سیاست بهینه را کشف کند. بنابراین، علاوه بر یادگیری ارزش‌ها، باید انتخاب عمل نیز انجام دهد.)

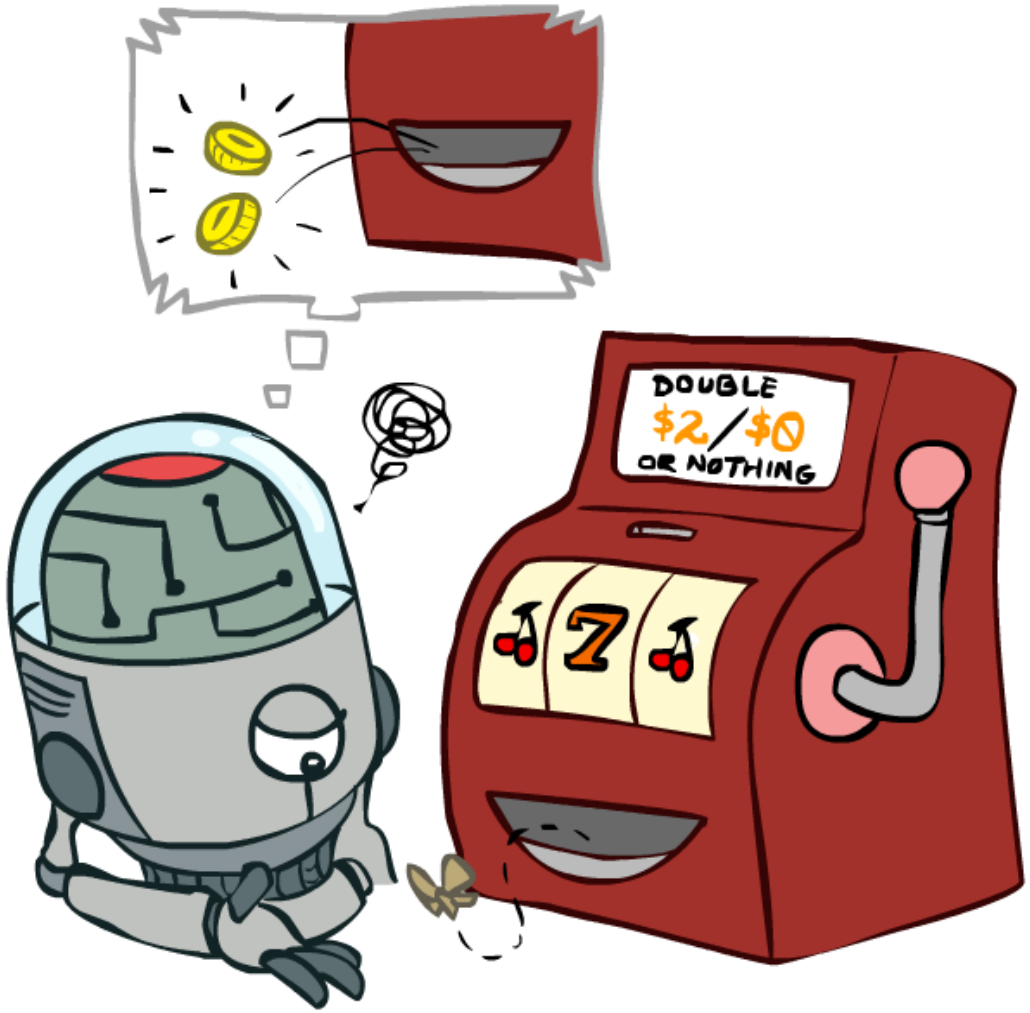
- **یادگیری مستقل از مدل :**

Q-Learning

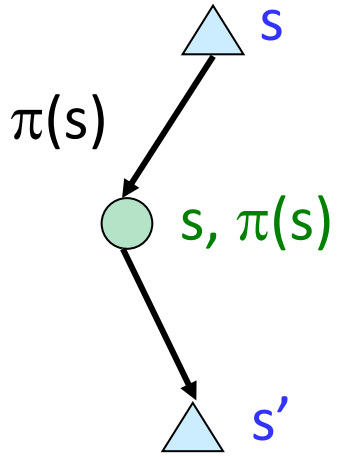
- **Q-Learning:** به‌روزرسانی تدریجی ارزش هر Q-state با استفاده از بیشینه‌ی ارزش Q-state های حالت بعدی
- **Approximate Q-Learning:** به‌روزرسانی تدریجی ارزش هر Q-state با استفاده از یک تابع تقریب

- **چالش‌های اصلی:**
  - چگونه به‌صورت مؤثر محیط را کاوش کند؟
  - چگونه بین کاوش (exploration) و بهره‌برداری (exploitation) تعادل برقرار کند؟

یادگیری تفاضل زمانی
Temporal Difference Learning (TD Learning)



یادگیری تفاضل زمانی



❑ **ایده:** یادگیری از تک تک نمونه‌ها به صورت جداگانه!

- به روز رسانی تخمین ارزش حالت فعلی V(s) پس از هر تجربه (هر گام تعامل با محیط) (s, a, s′, r)
- عامل برای اصلاح ارزش حالت فعلی، به پاداش آن گام و ارزش حالت بعدی نگاه می‌کند.
- حالت‌هایی مانند s′ که احتمال وقوع بیشتری دارند، سهم بیشتری در تعیین ارزش V(s) خواهند داشت

❑ **یادگیری ارزش‌ها به روش تفاضل زمانی:**

- سیاست عامل در طول فرآیند یادگیری ثابت باقی می‌ماند و تمرکز بر ارزیابی ارزش حالت‌هاست.
- با تکرار گام‌های تعامل، ارزش حالت‌ها به‌تدریج به ارزش‌های واقعی همگرا می‌شوند.

نمونه برداری از V(s)

$$sample = R(s, \pi(s), s') + \gamma V^{\pi}(s')$$

به‌روز رسانی ارزش V(s)

$$V^{\pi}(s) \leftarrow (1-\alpha)V^{\pi}(s) + (\alpha)sample$$

همان به‌روزرسانی قبلی

$$V^{\pi}(s) \leftarrow V^{\pi}(s) + \alpha(sample - V^{\pi}(s))$$

خطای تفاضل زمانی (TD Error)

$V^{\pi}(s) \leftarrow (1-\alpha)V^{\pi}(s) + (\alpha)sample$

میانگین گیری نمایی

❑میانگین گیری نمایی

- قاعده‌ی به روز رسانی بر اساس درون‌یابی:

$$\bar{x}_n = (1-\alpha) \cdot \bar{x}_{n-1} + \alpha \cdot x_n$$

- استفاده از این قاعده باعث می‌شود نمونه‌های جدیدتر اهمیت بیشتری داشته باشند:

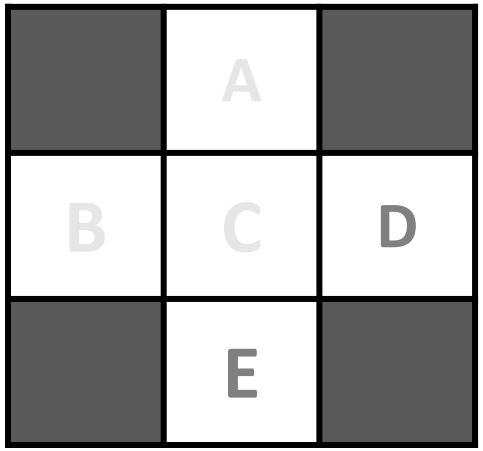
$$\bar{x}_n = \frac{x_n + (1-\alpha) \cdot x_{n-1} + (1-\alpha)^2 \cdot x_{n-2} + \dots}{1 + (1-\alpha) + (1-\alpha)^2 + \dots}$$

- گذشته را نادیده می‌گیرد (چون مقادیر گذشته‌ی دور به‌هرحال خیلی درست نیستند).

❑اگر نرخ یادگیری (α) به تدریج کاهش دهیم، می‌تواند باعث همگرا شدن میانگین‌ها شود.

مثال: یادگیری تفاضل زمانی

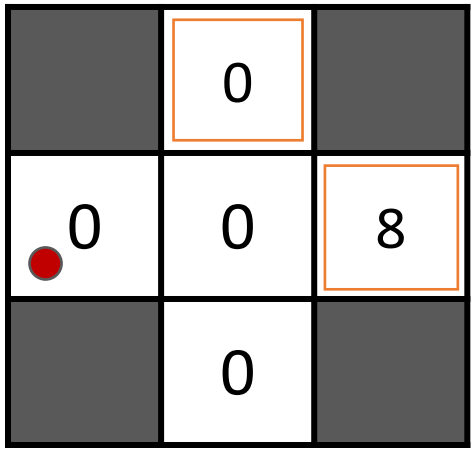
حالت‌ها



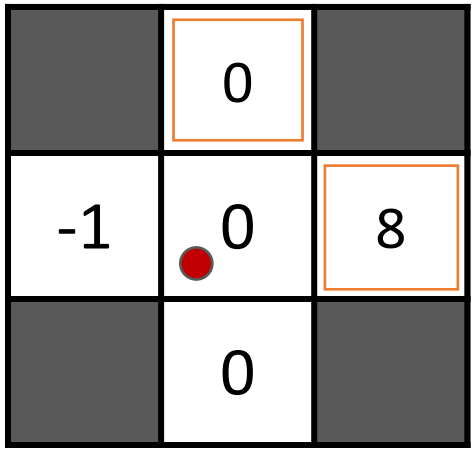
Assume: $\gamma = 1$, $\alpha = 1/2$

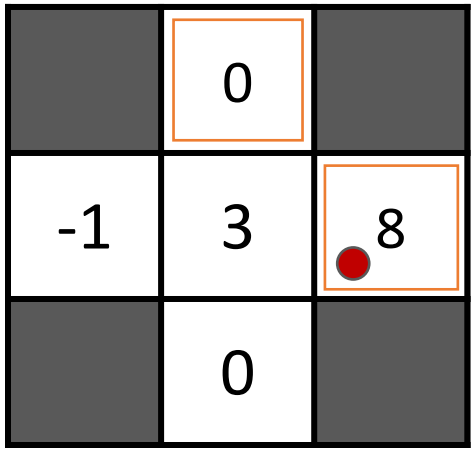
تغییر حالت‌های مشاهده شده

B, east, C, -2



C, east, D, -2

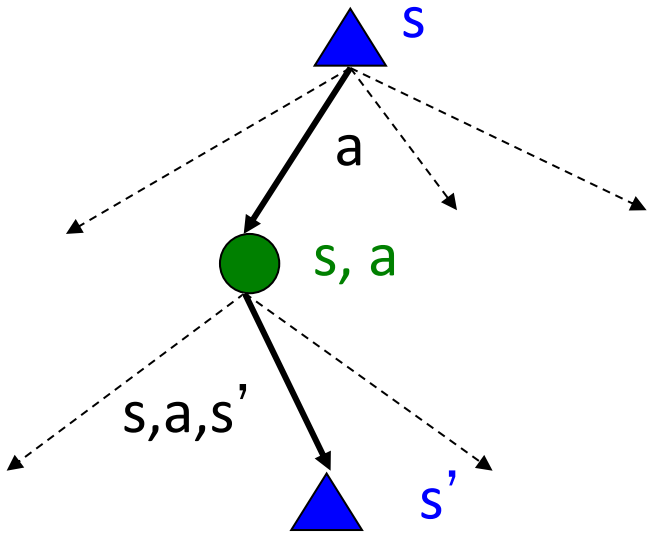




$$V^{\pi}(s) \leftarrow (1-\alpha)V^{\pi}(s) + \alpha\left[R(s,\pi(s),s') + \gamma V^{\pi}(s')\right]$$

مشکلات یادگیری تفاضل زمانی (TD)

- یادگیری ارزش به روش تفاضل زمانی یک روش مستقل از مدل برای ارزیابی سیاست است که با استفاده از میانگین‌گیری نمونه‌ای در حال اجرا، به نوعی به‌روزرسانی‌های بلمن را شبیه‌سازی می‌کند.
- اما اگر بخواهیم این ارزش ها را به یک سیاست جدید تبدیل کنیم، به مشکل برمی‌خوریم!



$$\pi(s) = \arg\max_a Q(s,a)$$

$$Q(s,a) = \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V(s') \right]$$

- ایده: به‌جای یادگیری ارزش حالت‌ها، ارزش حالت‌های Q را یاد بگیریم.
  - در این صورت، انتخاب عمل به صورت مستقل از مدل قابل انجام خواهد بود!

یادگیری تقویتی -- مرور کلی

❑ یادگیری تقویتی منفعل (Passive RL) :

یاد گرفتن از تجربه‌ها (عامل با یک سیاست از پیش‌مشخص‌شده حرکت می‌کند و هدف آن ارزیابی ارزش حالت‌ها است، نه تصمیم‌گیری)

1. یادگیری مبتنی بر مدل :

1. عامل ابتدا مدل MDP (یعنی T و R) را بر پایه‌ی تجربه‌های خود تخمین می‌زند. سپس با استفاده از این مدل تقریبی، ارزش حالت‌ها را محاسبه می‌کند.

2. یادگیری مستقل از مدل:

عامل فقط بر پایه‌ی تجربه، بدون دانستن مدل، ارزش حالت‌ها را محاسبه می‌کند.(مستقیماً ارزش هر حالت را یاد می‌گیرد)

Value Learning

1. **ارزیابی مستقیم:** محاسبه‌ی میانگین مجموع پاداش‌ها در اپیزودها
2. **یادگیری تفاضل زمانی:** به‌روزرسانی تدریجی ارزش هر حالت با استفاده از ارزش حالت بعدی

❑ یادگیری تقویتی فعال (Active RL):

عامل خودش باید تصمیم بگیرد و تجربه جمع‌آوری کند (عامل دیگر فقط یک سیاست را دنبال نمی‌کند، بلکه باید خودش سیاست بهینه را کشف کند. بنابراین، علاوه بر یادگیری ارزش‌ها، باید انتخاب عمل نیز انجام دهد.)

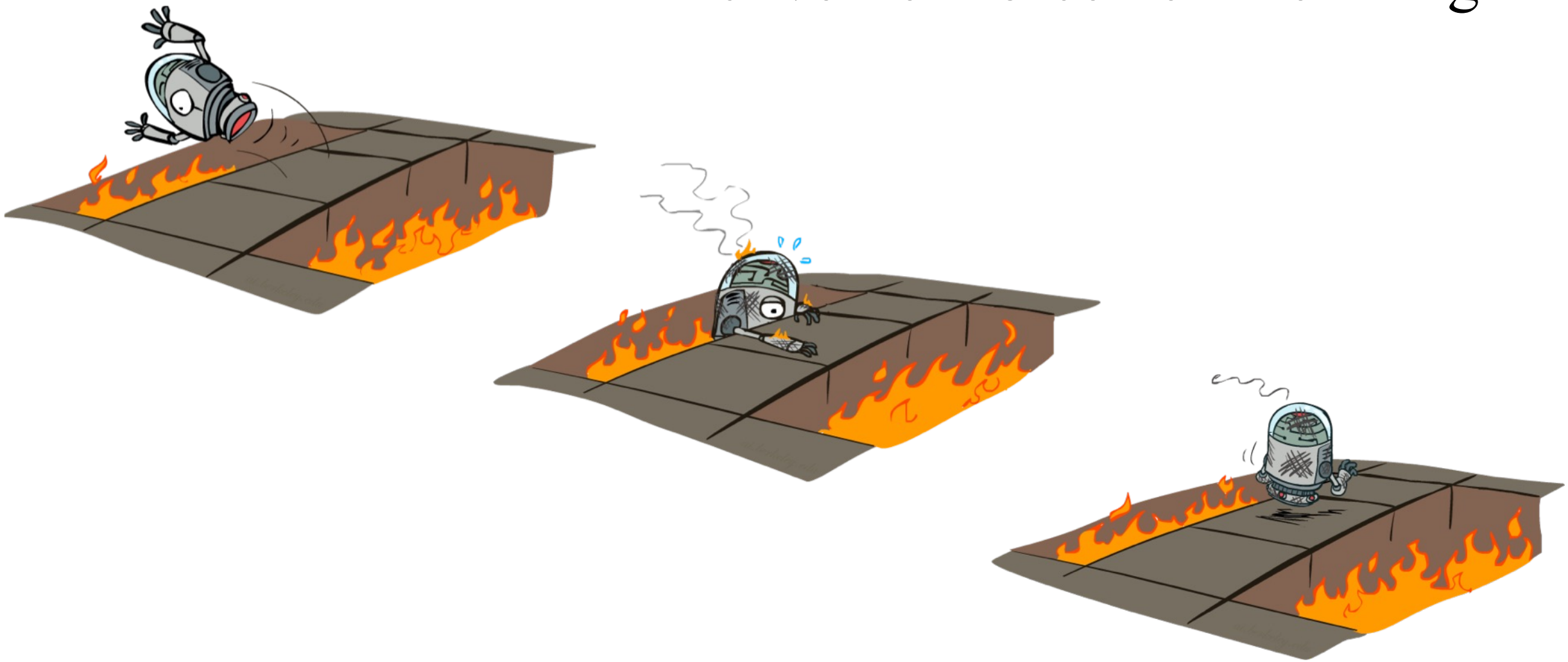
- **یادگیری مستقل از مدل :**

Q-Learning

- **Q-Learning:** به‌روزرسانی تدریجی ارزش هر Q-state با استفاده از بیشینه‌ی ارزش Q-state های حالت بعدی
- **Approximate Q-Learning:** به‌روزرسانی تدریجی ارزش هر Q-state با استفاده از یک تابع تقریب

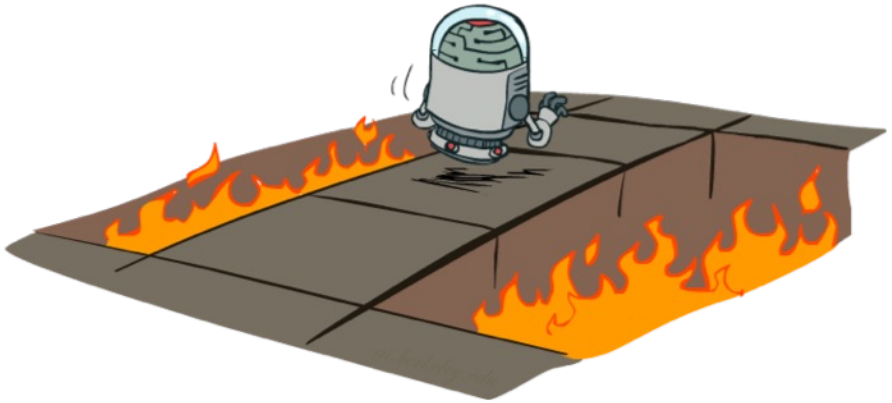
- **چالش‌های اصلی:**
  - چگونه به‌صورت مؤثر محیط را کاوش کند؟
  - چگونه بین کاوش (exploration) و بهره‌برداری (exploitation) تعادل برقرار کند؟

یادگیری تقویتی فعال
Active Reinforcement Learning



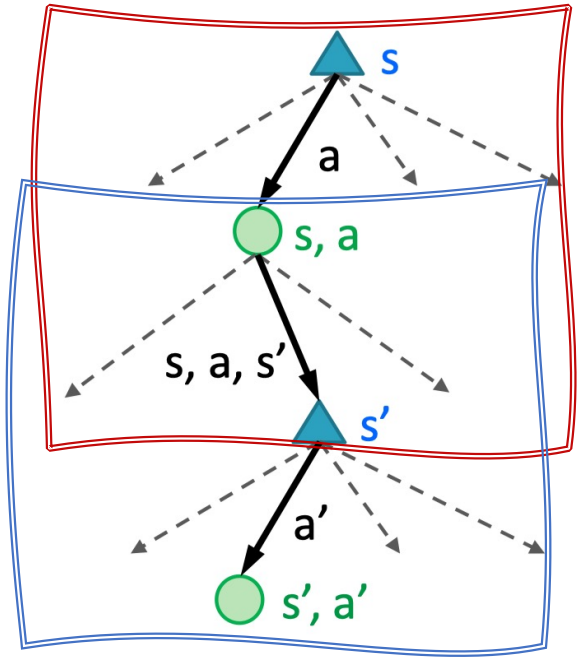
یادگیری تقویتی فعال

- یادگیری تقویتی فعال: محاسبه سیاست‌های بهینه
  - تابع تغییر حالت $T(s,a,s')$ ناشناخته است.
  - تابع پاداش $R(s,a,s')$ نیز ناشناخته است.
  - اکنون عامل خودش اعمال را انتخاب می‌کند.
  - هدف: یادگیری سیاست بهینه / مقادیر بهینه



- در این مورد:
  - یادگیرنده خودش تصمیم می‌گیرد!
  - یک موازنه‌ی اساسی وجود دارد: کاوش (exploration) در برابر بهره‌برداری (exploitation)
  - این برنامه‌ریزی آفلاین نیست! در واقع، عامل واقعاً در محیط اقدام می‌کند و با تجربه، نتیجه را متوجه می‌شود...

Q-Value Iteration



❑ **تکرار مقدار (Value Iteration) : محاسبه ارزش حالت‌ها به صورت تکرار شونده**

- با بردار اولیه $V_0(s) = 0$ شروع می‌کنیم که می‌دانیم مقدار درستی است.
- اگر بردار $V_k(s)$ را داشته باشیم می‌توانیم بردار $V_{k+1}(s)$ را محاسبه کنیم.

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

❑ **اما مقادیر Q (Q-values) کاربردی‌تر هستند، پس به‌جای V آن‌ها را محاسبه می‌کنیم.**

- با مقدار اولیه $Q_0(s,a) = 0$ شروع می‌کنیم، که می‌دانیم مقدار درستی است.
- اگر بردار $Q_k(s,a)$ را داشته باشیم، می‌توانیم بردار $Q_{k+1}(s,a)$ را محاسبه کنیم.

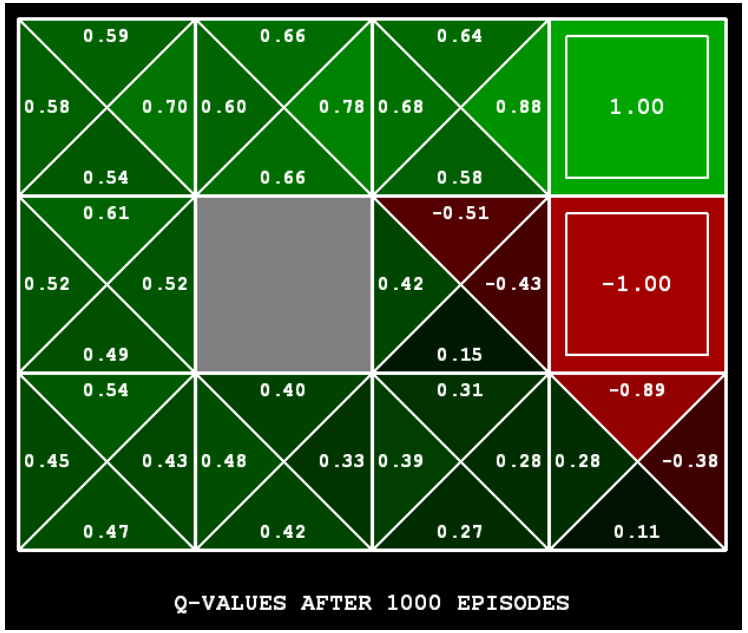
$$Q_{k+1}(s,a) \leftarrow \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma \max_{a'} Q_k(s',a') \right]$$

الگوریتم یادگیری Q
Q-Learning

- می‌خواهیم که به‌روزرسانی‌های مقدار Q را برای هر Q-state انجام دهیم:

$$Q_{k+1}(s,a) \leftarrow \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma \max_{a'} Q_k(s',a') \right]$$

- اما نمی‌توانیم این به‌روزرسانی را انجام دهیم، چون T و R را نمی‌دانیم.



- در عوض، میانگین را در حین اجرا (تجربه) محاسبه می‌کنیم:
  - دریافت یک نمونه (s,a,r,s′)
  - در نظر گرفتن تخمین قبلی: $Q(s,a)$
  - در نظر گرفتن تخمین مربوط به نمونه‌ی جدید:

$$sample = R(s,a,s') + \gamma \max_{a'} Q(s',a')$$

  - اما ما می‌خواهیم میانگین را بر اساس تمام نتایج ممکن از (s,a) بگیریم.
  - بنابراین از ترکیب تخمین جدید با میانگین قبلی برای رسیدن به مقدار پایدارتر استفاده می‌کنیم (running average)

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + (\alpha)\,[sample]$$

$\alpha = 0.5$

$\gamma = 1$

ویدئوی نمایشی Q-Learning در Gridworld

$$sample = R(s, a, s') + \gamma \max_{a'} Q(s', a')$$

$$Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + (\alpha)\,[sample]$$