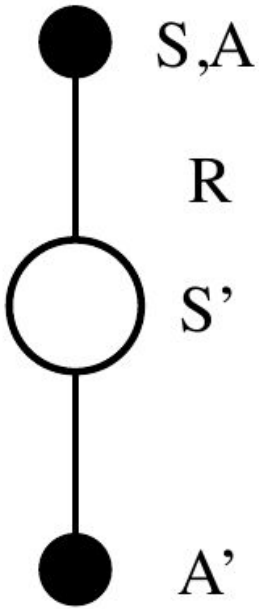
Aprendizaje Reforzado

Maestría en Ciencia de Datos, DC - UBA

Julián Martínez
Javier Kreiner

Sarsa

$$Q(S,A) \leftarrow Q(S,A) + \alpha \left(R + \gamma Q(S',A') - Q(S,A) \right)$$

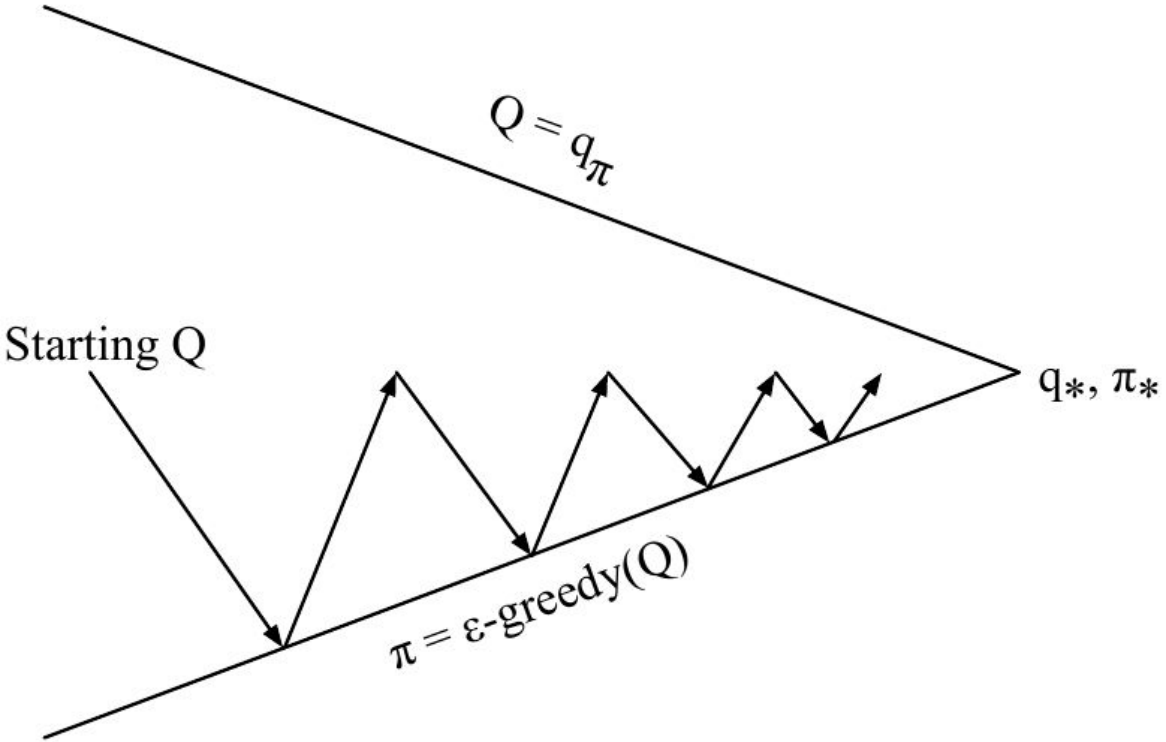


Misma idea que TD pero para función de acción-valor.

$$\sum_{t=1}^{\infty} \alpha_t = \infty$$

$$\sum_{t=1}^{\infty} \alpha_t^2 < \infty$$

Control (improvement) on-policy con Sarsa



On policy: Tomo acciones con la misma política que estoy mejorando

Sarsa - pseudocódigo

Initialize $Q(s,a), \forall s \in \mathcal{S}, a \in \mathcal{A}(s)$, arbitrarily, and $Q(\textit{terminal-state}, \cdot) = 0$

Repeat (for each episode):

  Initialize S

  Choose A from S using policy derived from Q (e.g., ε-greedy)

  Repeat (for each step of episode):

    Take action A, observe R, S'

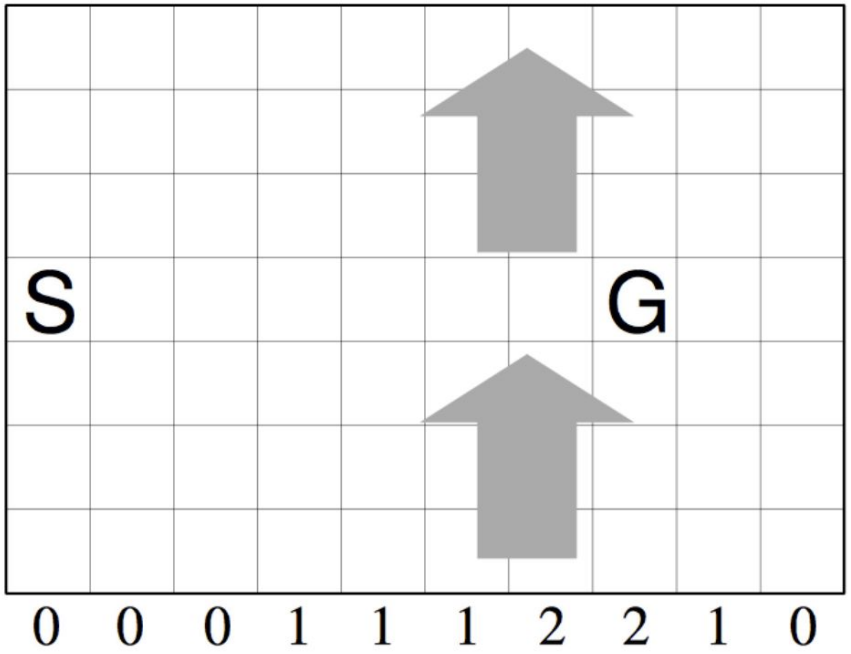
    Choose A' from S' using policy derived from Q (e.g., ε-greedy)

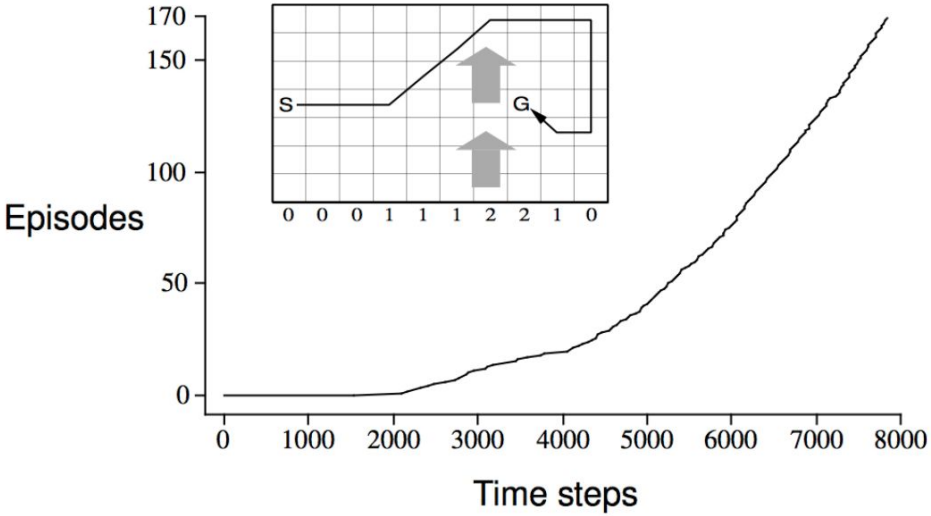
    $Q(S,A) \leftarrow Q(S,A) + \alpha\big[R + \gamma Q(S',A') - Q(S,A)\big]$

    $S \leftarrow S'$; $A \leftarrow A'$;

  until S is terminal

Gridworld con viento





Apredizaje off-policy

Utilizo una política *exploratoria* $\mu(a|s)$ para mejorar la política *óptima* $\pi(a|s)$.

Aprendo observando la experiencia de otros agentes.

¿Cómo mezclar las dos experiencias?

Importance Sampling - Off-policy MC

$$G_t^{\pi/\mu} = \frac{\pi(A_t|S_t)}{\mu(A_t|S_t)} \frac{\pi(A_{t+1}|S_{t+1})}{\mu(A_{t+1}|S_{t+1})} \cdots \frac{\pi(A_T|S_T)}{\mu(A_T|S_T)} G_t$$

$$V(S_t) \leftarrow V(S_t) + \alpha \left(G_t^{\pi/\mu} - V(S_t) \right)$$

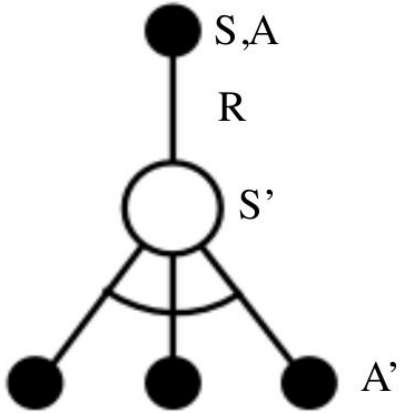
Q-learning

Los episodios los genero con μ pero la estimación del retorno esperado la calculo con una acción tomada con π.

$$A_{t+1} \sim \mu(\cdot|S_t)$$
$$A' \sim \pi(\cdot|S_t)$$

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \left(R_{t+1} + \gamma Q(S_{t+1}, A') - Q(S_t, A_t) \right)$$

Off-Policy, Q-learning



Dada $Q^k(s,a)$:

$$\pi_{k+1}(s) = \arg\max_{a'} Q^k(S_t, a'), \qquad \mu_{k+1}(a|s) = \pi^{\varepsilon}_{k+1}.$$

$$Q^{k+1}(S,A) = Q^k(S,A) + \alpha(R + \gamma \max_{a'} Q^k(S', a') - Q^k(S,A))$$

Q-learning (off-policy TD control) for estimating $\pi \approx \pi_*$

Algorithm parameters: step size $\alpha \in (0, 1]$, small $\varepsilon > 0$
Initialize $Q(s, a)$, for all $s \in \mathcal{S}^+, a \in \mathcal{A}(s)$, arbitrarily except that $Q(\mathit{terminal}, \cdot) = 0$

Loop for each episode:
    Initialize S
    Loop for each step of episode:
        Choose A from S using policy derived from Q (e.g., ε-greedy)
        Take action A, observe R, S'
        $Q(S, A) \leftarrow Q(S, A) + \alpha\big[R + \gamma \max_a Q(S', a) - Q(S, A)\big]$
        $S \leftarrow S'$
    until S is terminal

Diferencias Temporales (programación)

- Predicción TD
- Sarsa
- Q-learning

Problema (programación)

- Implementar expected-Sarsa, Sutton sección 6.6, es igual Q-learning, pero la ecuación de update es, usarlo para resolver Windy Gridworld y Cliff Environment :

$$
\begin{aligned}
Q(S_t, A_t) &\leftarrow Q(S_t, A_t) + \alpha \Big[R_{t+1} + \gamma \mathbb{E}[Q(S_{t+1}, A_{t+1}) \mid S_{t+1}] - Q(S_t, A_t) \Big] \\
&\leftarrow Q(S_t, A_t) + \alpha \Big[R_{t+1} + \gamma \sum \pi(a|S_{t+1}) Q(S_{t+1}, a) - Q(S_t, A_t) \Big],
\end{aligned}
$$