

Aprendizaje por refuerzo

Métodos basados en muestreo (2)

Antonio Manjavacas Lucas

manjavacas@ugr.es

1. Métodos *on-policy*
2. Métodos *off-policy*

Control MC sin inicios de exploración

Vamos a estudiar dos alternativas a MC con inicios de exploración:

Control MC sin inicios de exploración

Vamos a estudiar dos alternativas a MC con inicios de exploración:

Métodos *on-policy*

- 👤 Se emplea **una única política** que mejora progresivamente, permitiendo siempre cierta exploración.
 - Mejoran y evalúan constantemente la misma política.

Control MC sin inicios de exploración

Vamos a estudiar dos alternativas a MC con inicios de exploración:

Métodos *on-policy*

- 👤 Se emplea **una única política** que mejora progresivamente, permitiendo siempre cierta exploración.
 - Mejoran y evalúan constantemente la misma política.

Métodos *off-policy*

- 👤 El agente aprende una **política objetivo** (**target policy**) a partir de datos generados por otra **política exploratoria** (**behaviour policy**).
 - La política que empleamos para aprender/explorar “está fuera” (off) de la que empleamos para seleccionar acciones.

Métodos *on-policy*

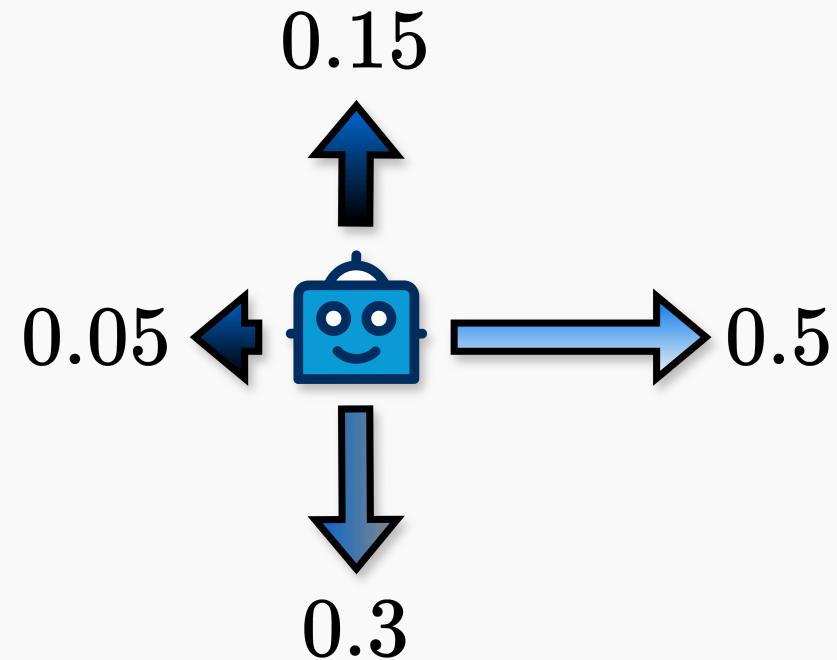
Métodos *on-policy*

Los métodos  ***on-policy*** emplean una única política.

Esta política *aspira* a un comportamiento óptimo, pero siempre debe reservar cierta **probabilidad de explorar**.

Las políticas empleadas generalmente son ***soft*** (“suaves”), es decir:

$$\pi(a|s) > 0 \quad \forall s \in \mathcal{S}, a \in \mathcal{A}$$



Métodos *on-policy*

MC con inicios de exploración es *on-policy* ...

Métodos *on-policy*

MC con inicios de exploración es *on-policy* ...
... aunque poco viable, como hemos adelantado.

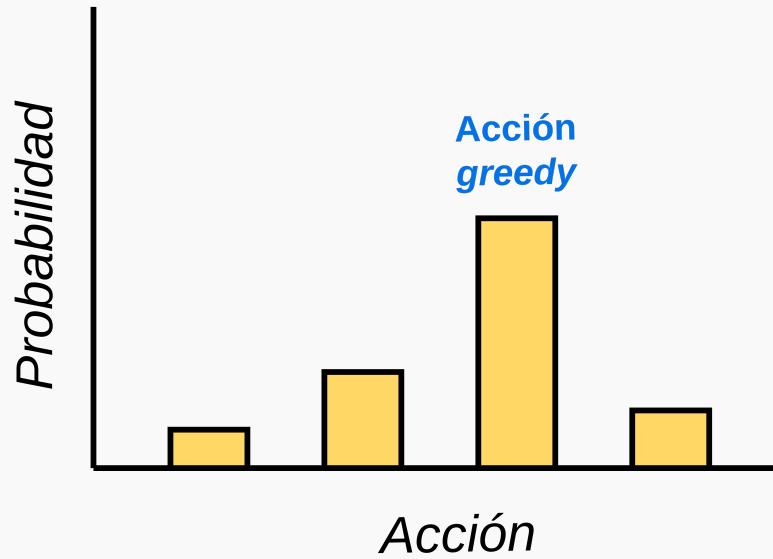
Métodos *on-policy*

MC con inicios de exploración es *on-policy* ...
... aunque poco viable, como hemos adelantado.

Una opción más apropiada son las **políticas ϵ -greedy**.

Políticas ϵ -greedy

Las políticas ϵ -greedy son políticas estocásticas que siempre permiten cierta probabilidad $\epsilon > 0$ de explorar.



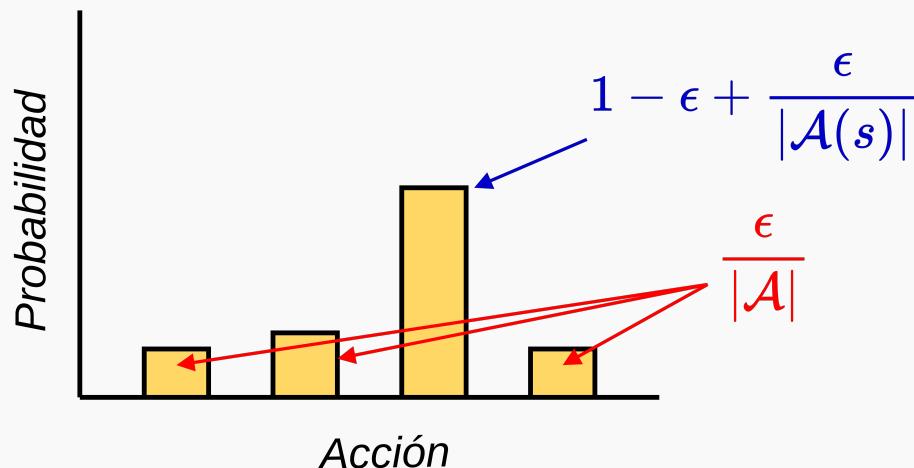
La acción *greedy* es aquella que se elige con mayor probabilidad.

Eventualmente, el resto de acciones (no óptimas) podrían explorarse con probabilidad ϵ .

- El valor de ϵ puede reducirse gradualmente, hasta que la política sea prácticamente determinista.

Políticas ϵ -soft

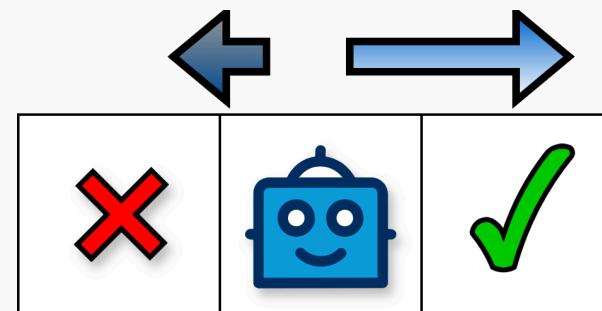
ϵ -greedy es un subconjunto de las políticas conocidas como ϵ -soft.



Siempre permiten cierta exploración.
En el caso de ϵ -greedy:

$$\pi(a|s) \geq \frac{\epsilon}{|A(s)|}$$

- Si $\varepsilon > 0$ estas políticas nunca pueden ser óptimas. Esto se debe a que **siempre existe cierta probabilidad de realizar acciones sub-óptimas** (explorar).
- No convergen en una política óptima, pero sí en una **muy aproximada**. Además, evitan emplear inicios de exploración.



Métodos on-policy

On-policy first-visit MC control (for ε -soft policies), estimates $\pi \approx \pi_*$

Algorithm parameter: small $\varepsilon > 0$

Initialize:

$\pi \leftarrow$ an arbitrary ε -soft policy

$Q(s, a) \in \mathbb{R}$ (arbitrarily), for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

$Returns(s, a) \leftarrow$ empty list, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

Repeat forever (for each episode):

Generate an episode following π : $S_0, A_0, R_1, \dots, S_{T-1}, A_{T-1}, R_T$

$G \leftarrow 0$

Loop for each step of episode, $t = T-1, T-2, \dots, 0$:

$G \leftarrow \gamma G + R_{t+1}$

Unless the pair S_t, A_t appears in $S_0, A_0, S_1, A_1, \dots, S_{t-1}, A_{t-1}$:

Append G to $Returns(S_t, A_t)$

$Q(S_t, A_t) \leftarrow$ average($Returns(S_t, A_t)$)

$A^* \leftarrow \arg \max_a Q(S_t, a)$ (with ties broken arbitrarily)

For all $a \in \mathcal{A}(S_t)$:

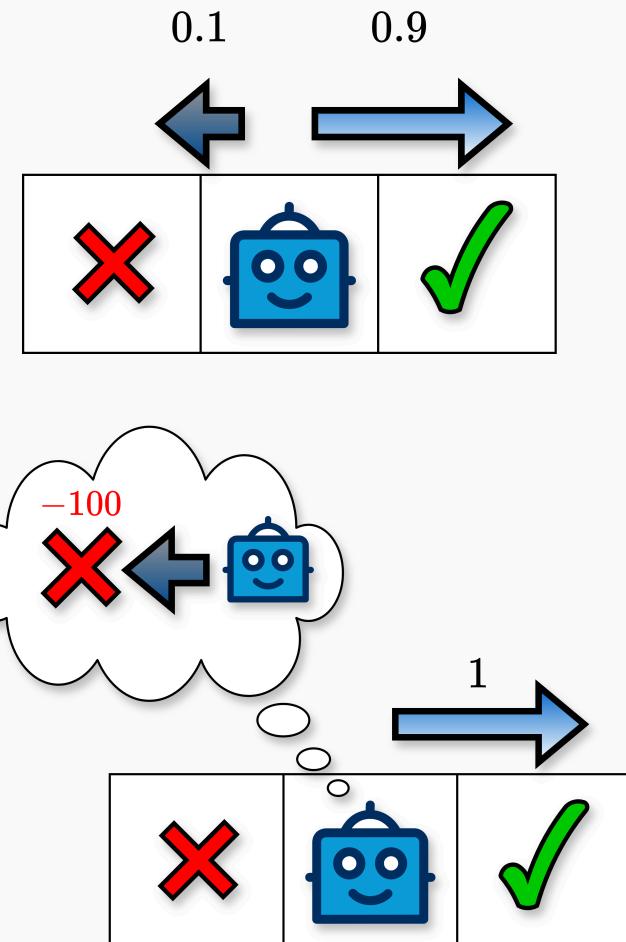
$$\pi(a|S_t) \leftarrow \begin{cases} 1 - \varepsilon + \varepsilon/|\mathcal{A}(S_t)| & \text{if } a = A^* \\ \varepsilon/|\mathcal{A}(S_t)| & \text{if } a \neq A^* \end{cases}$$

Limitaciones de los métodos *on-policy*

? ¿Existe alguna **alternativa** a mantener siempre cierta probabilidad de explorar?

MC *on-policy* supone aprender una política **muy cercana a la óptima**, pero siempre existe cierta probabilidad de elegir acciones sub-óptimas.

Los métodos  ***off-policy*** son una alternativa.



Métodos *off-policy*

Métodos *off-policy*

Los métodos  ***off-policy*** hacen uso de dos políticas:

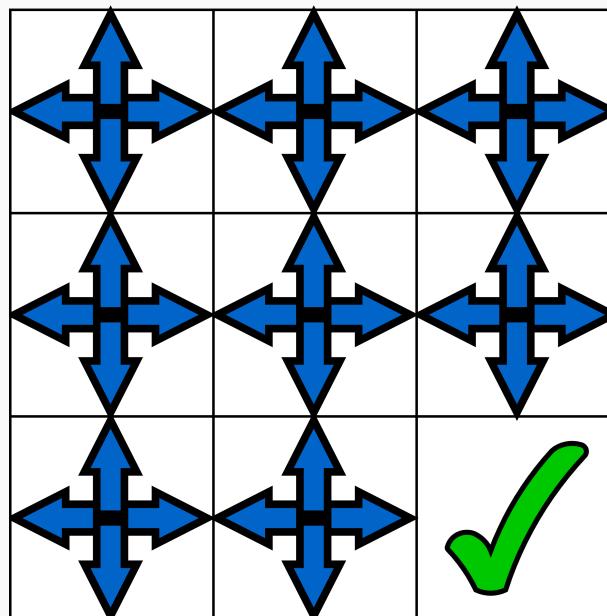
1. **Política objetivo** (*target policy*). Destinada a ser óptima.
2. **Política de comportamiento** (*behaviour policy*). Política exploratoria empleada para “generar comportamiento” (muestrear, acumular experiencia).

En este caso, decimos que **el aprendizaje de la política óptima se hace a partir de datos/resultados “fuera” (*off*) de la política objetivo.**

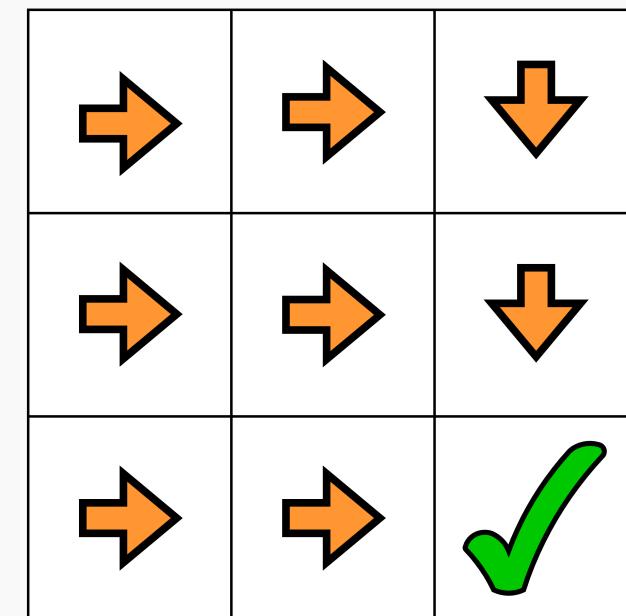
- Es decir, mediante información obtenida por la política de comportamiento.

Métodos off-policy

*Política de
comportamiento*



Política objetivo



Ejemplo. On-policy vs. Off-policy

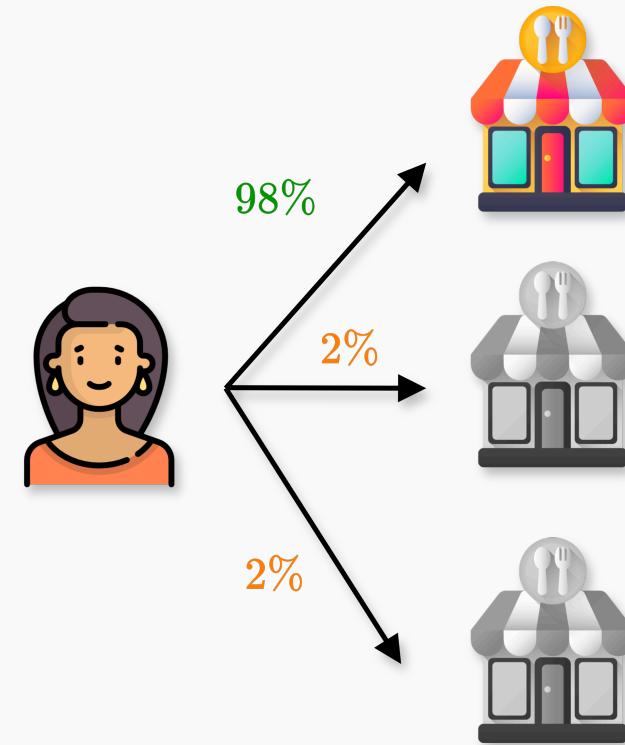
👤 On-policy

Imagina que tienes un restaurante favorito al que sueles ir a comer (**acción greedy**).

- Al principio, es posible que tu criterio no sea muy preciso pero, a medida que visitas todos los restaurantes varias veces, cada vez repites más el mismo.

Algunos días vuelves a otros restaurantes que consideras peores para ver si la calidad ha mejorado (**exploración**).

- Eventualmente estás abierto a dar una nueva oportunidad a restaurantes peores.

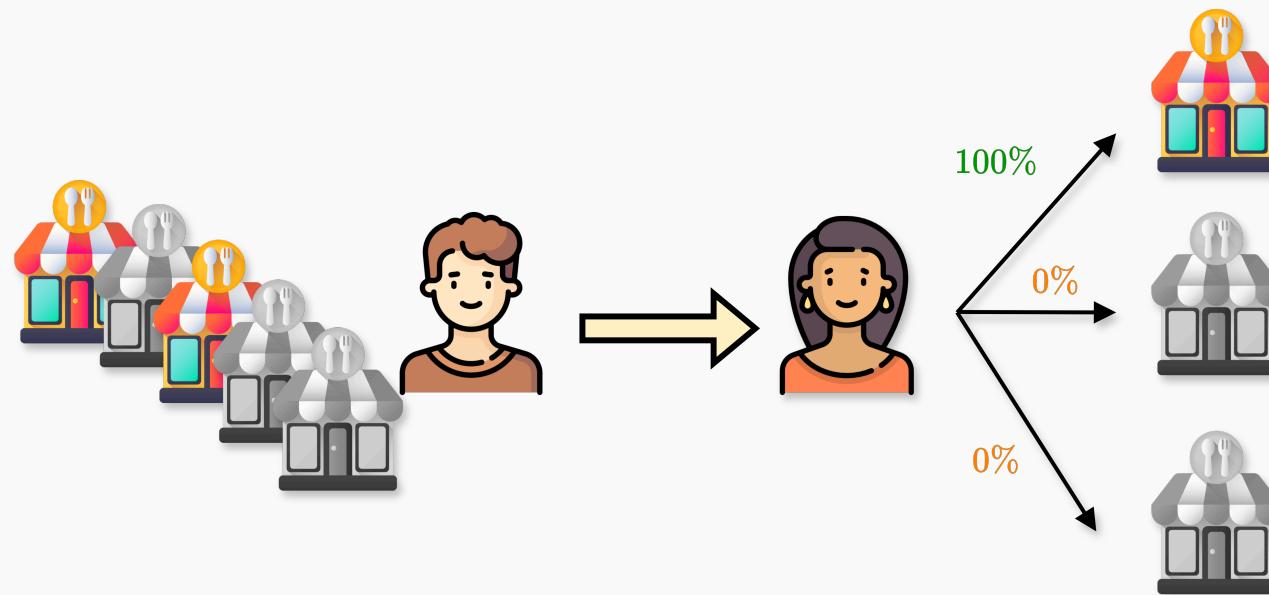


Ejemplo. On-policy vs. Off-policy

Off-policy

Dejas que otra persona pruebe todos los restaurantes de la ciudad durante un tiempo (**política de comportamiento**). En base a su experiencia, eliges ir siempre al restaurante que te recomiende (**política objetivo**).

- Tú no pruebas nuevos restaurantes (no exploras), lo hace alguien por ti.



On-policy vs. Off-policy

 Los métodos ***on-policy*** son más simples, porque sólo se requiere una política.

 Los métodos ***off-policy*** requieren más tiempo para converger y presentan una mayor varianza (cambios de comportamiento más bruscos).

- No obstante, son más potentes y generales.
- De hecho, **son una generalización de los métodos *on-policy***, en el caso concreto en que las políticas objetivo y de comportamiento sean las mismas.

 Los métodos ***off-policy*** suelen emplearse para aprender a partir de datos generados por un controlador reactivo o por humanos.

¿Cuándo es preferible el aprendizaje off-policy?

- Aprendizaje a partir de datos generados por **humanos u otros agentes**.
- Aprendizaje a partir de la experiencia generada por **políticas anteriores**.
 - Reutilización de experiencia proveniente de versiones anteriores de la misma política.
- Aprendizaje de una política óptima **determinista empleando otra política exploratoria**.
- Aprendizaje a partir de la experiencia de **múltiples políticas combinadas**.

Predicción *off-policy*

Predicción off-policy

Objetivo: estimar v_π o q_π dadas las políticas π (objetivo) y b (comportamiento).

Si queremos emplear episodios de b para estimar valores para π , es necesario que cada acción tomada por b la pueda tomar también π (al menos, eventualmente).

Denominamos a esto **supuesto de cobertura**:

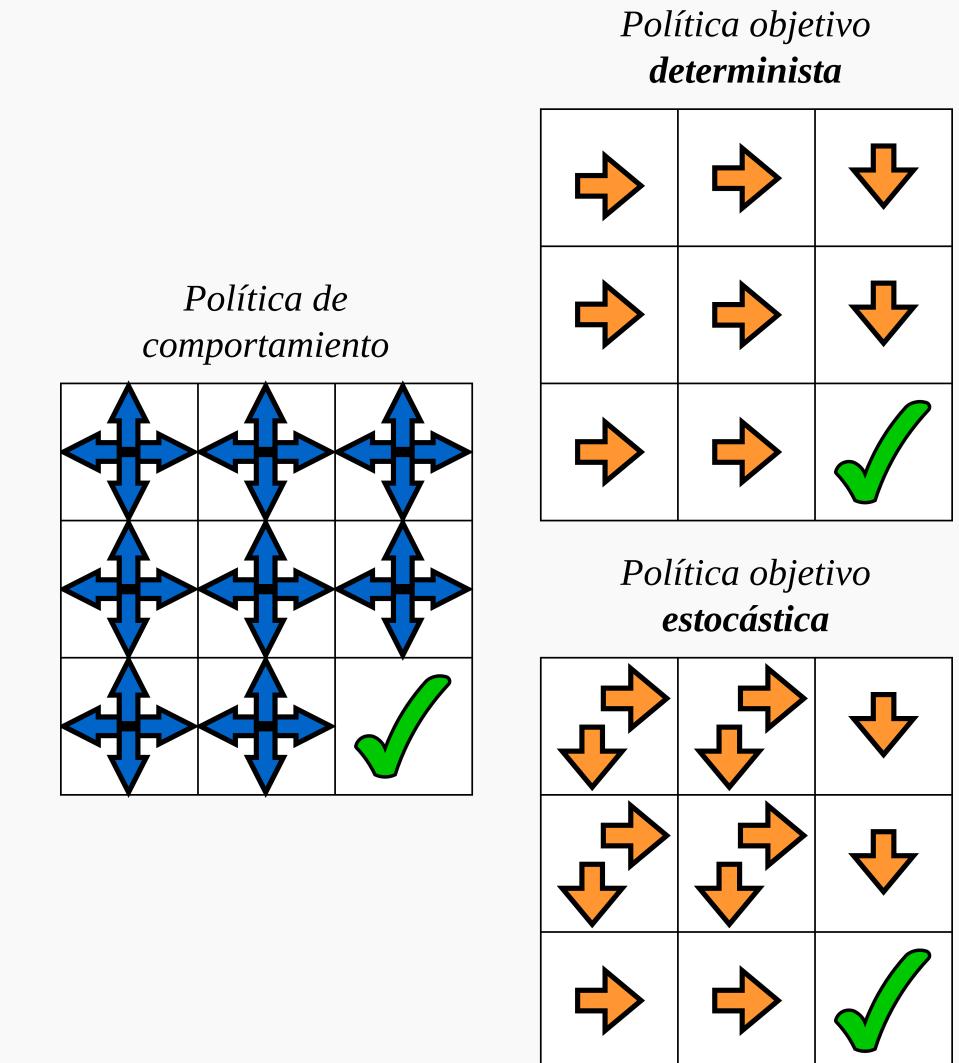
Si $\pi(a|s) > 0$, entonces $b(a|s) > 0$

- b es una política **estocástica** (no tiene por qué serlo al 100%, puede ser ε -greedy).
- π puede ser **determinista** o **estocástica**.

Políticas objetivo estocásticas

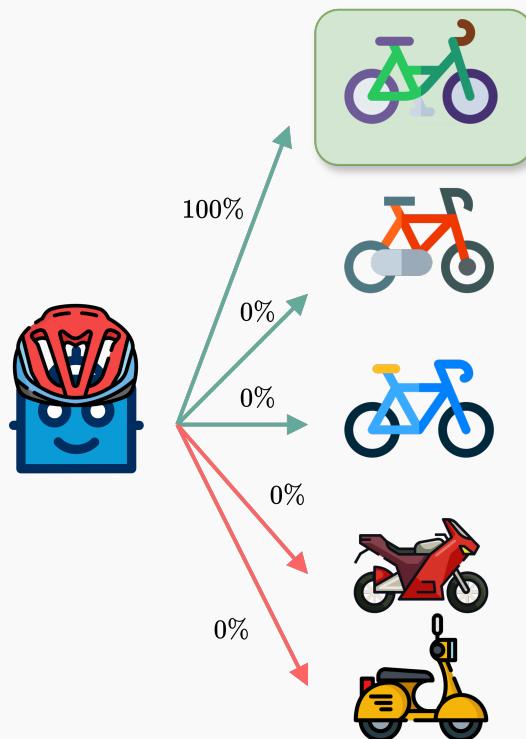
Generalmente consideraremos políticas objetivo π deterministas.

Aunque existen problemas donde puede ser útil que π sea estocástica.

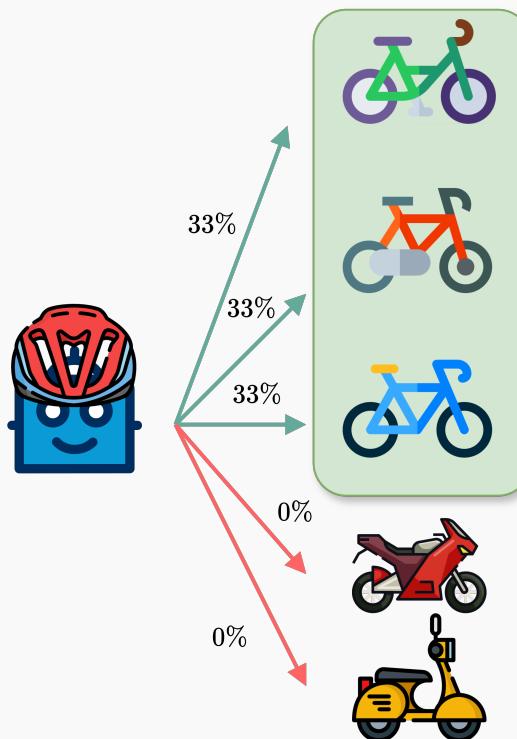


Políticas objetivo estocásticas

Política objetivo determinista



Política objetivo estocástica



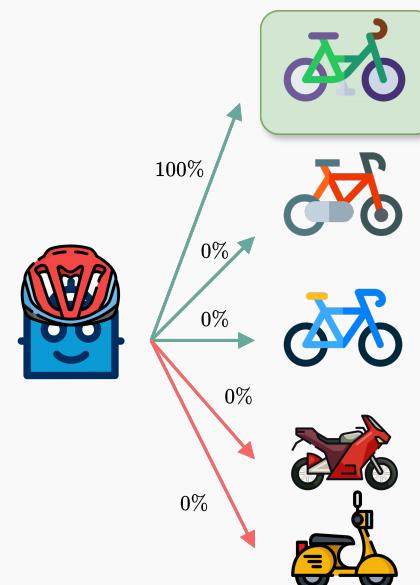
Políticas de comportamiento estocásticas

¿Para qué una **política objetivo π estocástica?**

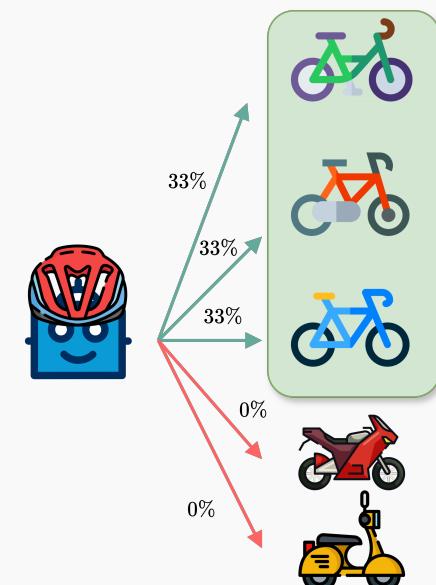
En este ejemplo, si π es **determinista** sólo contemplará una acción por estado, incluso si hay varias acciones óptimas.

Pero si es **estocástica**, tenemos una política que permite una **mayor variedad de acciones óptimas** para un mismo estado.

Política objetivo **determinista**



Política objetivo **estocástica**



Métodos off-policy

Sea como sea la política objetivo π , estamos tratando de obtener $v_\pi(s)$ a partir de experiencia generada por una política b diferente. Es decir, lo que tenemos es:

$$v_b(s) = \mathbb{E}[G_t | S_t = s]$$

El problema es que las distribuciones de estados y acciones bajo b y π pueden ser diferentes, dando lugar a un sesgo.

Si un subconjunto de estados es más frecuente siguiendo b , entonces π únicamente contará con información sobre esos estados, ignorando el resto.

 Una forma de solucionar esto es emplear *importance sampling*.

Importance sampling

Importance sampling

Importance sampling

El **muestreo por importancia**, o *importance sampling* es una técnica empleada en estadística para estimar el valor esperado de una distribución en base a ejemplos muestreados de una distribución diferente.

Veamos de forma intuitiva en qué consiste...

Importance sampling

Quiero obtener:

$$\mathbb{E}[g(X)]$$

Genero muestras aleatorias de una distribución:

$$X_1, X_2, \dots, X_n \sim \mathcal{D}$$

Aproximo el valor esperado con Monte Carlo:

$$\mathbb{E}[g(X)] \approx \frac{1}{n} \sum_{i=1}^n g(X_i)$$

Importance sampling

Valores de $g(X)$ podrían ser **poco probables** pero con una **contribución muy significativa** sobre $\mathbb{E}[g(X)]$.

Si no se *samplean* durante la estimación Monte Carlo, $\mathbb{E}[g(X)]$ se estimará mal.

El **muestreo por importancia** consiste en emplear una distribución “*modificada*” donde los valores más importantes (los que más afectan a la estimación de $\mathbb{E}[g(X)]$) se vuelven **más probables**.

Aseguramos así que sean muestreados y formen parte de la estimación Monte Carlo de $\mathbb{E}[g(X)]$.

Para paliar el efecto de este aumento de probabilidad, los valores se escalan dándoles un menor peso al ser muestreados.

Importance sampling

Muestreamos valores que provienen de la distribución *modificada*:

$$Y_1, Y_2, \dots, Y_n \sim \mathcal{D}'$$

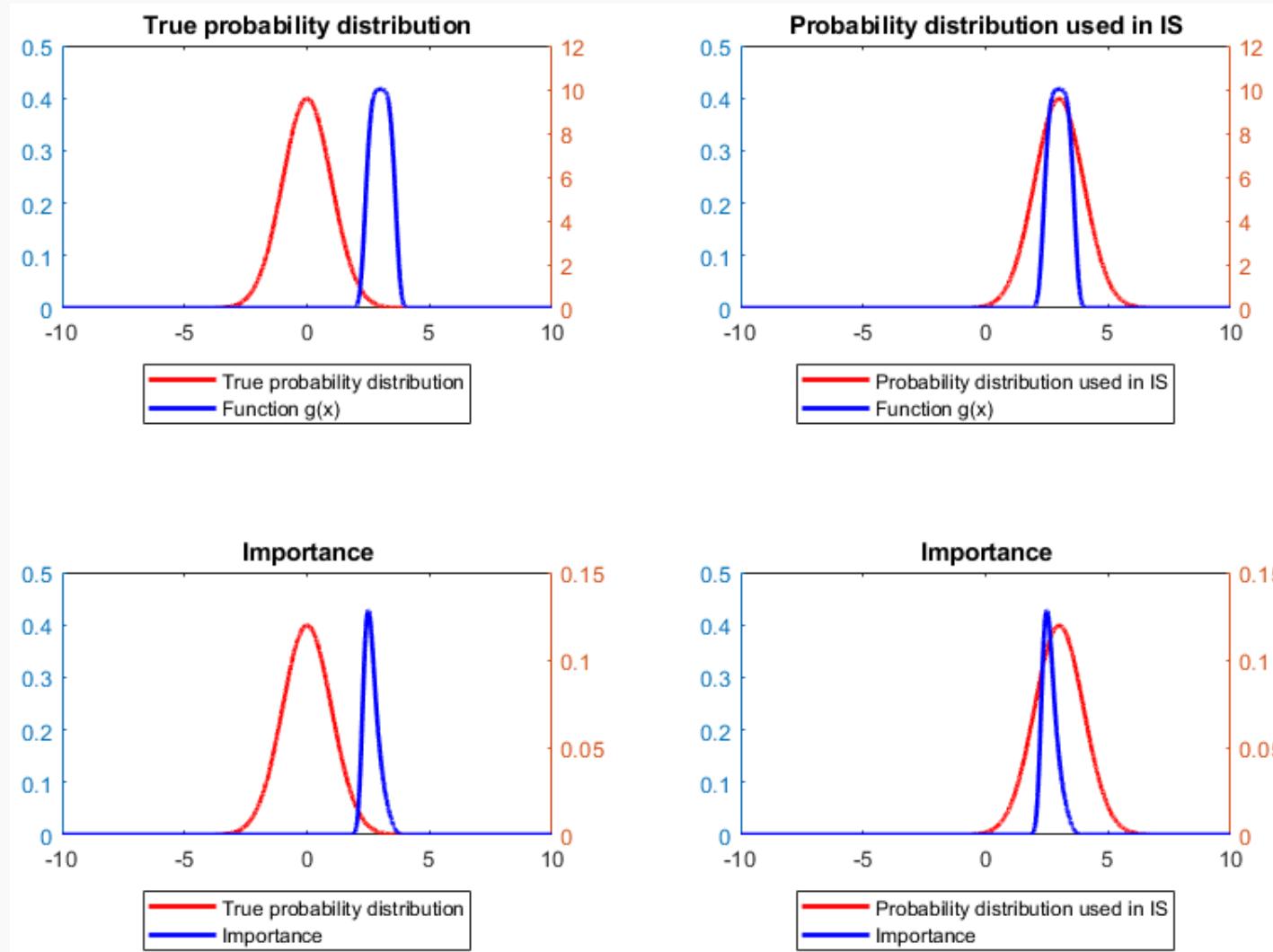
Y aproximamos de la siguiente manera:

$$\mathbb{E}[g(Y)] = \frac{1}{n} \sum_{i=1}^n \frac{p_{\mathcal{D}}(Y_i)}{p_{\mathcal{D}'}(Y_i)} g(Y_i)$$

Siendo:

$$\mathbb{E}[g(Y)] \approx \mathbb{E}[g(X)]$$

Importance sampling



Importance sampling

Intuitivamente:

- **Ampliamos** (*scale up*) los eventos que son raros en \mathcal{D}' pero comunes en \mathcal{D} .
- **Reducimos** (*scale down*) los eventos que son comunes en \mathcal{D}' pero raros en \mathcal{D} .

$$\mathbb{E}[g(Y)] = \frac{1}{n} \sum_{i=1}^n \frac{p_{\mathcal{D}}(Y_i)}{p_{\mathcal{D}'}(Y_i)} g(Y_i)$$

Importance sampling

$$\mathbb{E}[g(Y)] = \frac{1}{n} \sum_{i=1}^n \frac{p_{\mathcal{D}}(Y_i)}{p_{\mathcal{D}'}(Y_i)} g(Y_i)$$

$\frac{p_{\mathcal{D}}(Y_i)}{p_{\mathcal{D}'}(Y_i)} = 1$	Misma probabilidad en \mathcal{D} y \mathcal{D}' . La aportación no varía.
$\frac{p_{\mathcal{D}}(Y_i)}{p_{\mathcal{D}'}(Y_i)} > 1$	Y_i es más probable en la distribución original \mathcal{D} , por lo que su peso es mayor.
$\frac{p_{\mathcal{D}}(Y_i)}{p_{\mathcal{D}'}(Y_i)} < 1$	Y_i es más probable en la distribución modificada \mathcal{D}' , por lo que su peso es menor.
$\frac{p_{\mathcal{D}}(Y_i)}{p_{\mathcal{D}'}(Y_i)} = 0$	Y_i no aporta nada, porque no puede obtenerse en la distribución original.
$p_{\mathcal{D}'}(Y_i) = 0$	No se cumple el principio de cobertura.

¿Cómo se aplica en predicción *off-policy*?

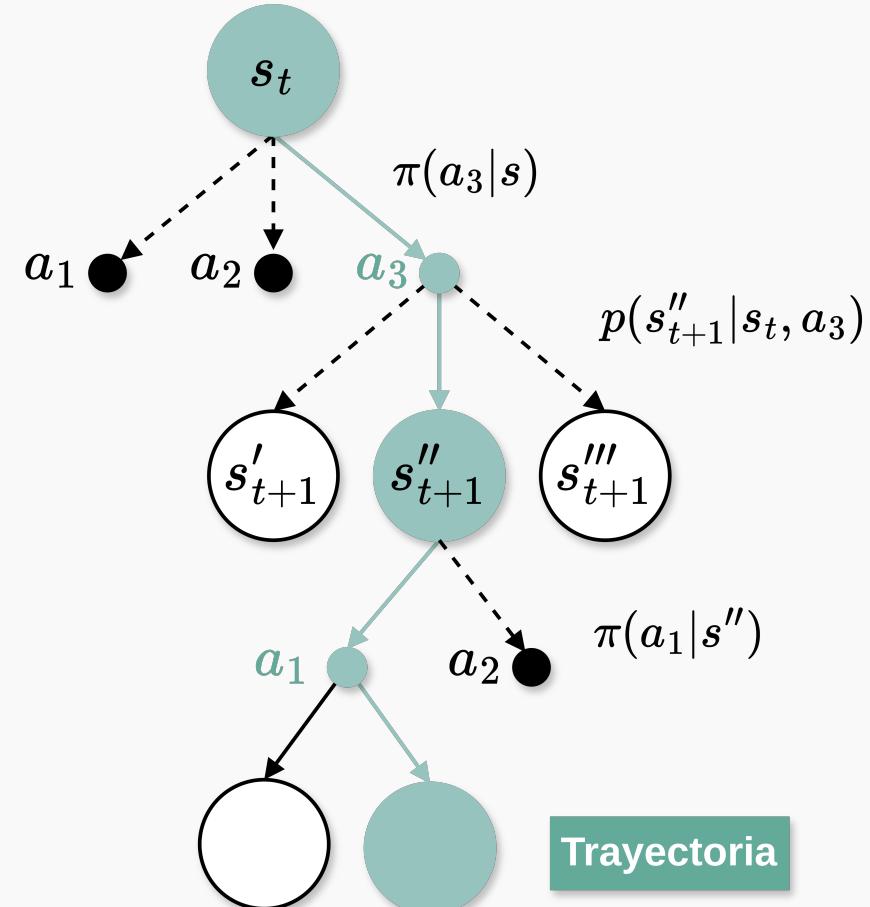
Predicción off-policy con *importance sampling*

Recordemos el concepto de **trayectoria**:

$$\tau = \{S_t, A_t, S_{t+1}, A_{t+1}, \dots, S_T\}$$

La **probabilidad** de realizar una trayectoria τ bajo una política π es:

$$\text{Prob}(\tau_\pi) = \prod_{k=t}^{T-1} \pi(A_k | S_k) p(S_{k+1} | S_k, A_k)$$



Predicción off-policy con *importance sampling*

¿Cómo de probable es seguir una trayectoria bajo la política π con respecto a la probabilidad de seguirla bajo la política b ?

Esto viene dado por el **importance sampling ratio**:

$$\rho_{t:T-1} = \frac{\text{Prob}(\tau_\pi)}{\text{Prob}(\tau_b)} = \frac{\prod_{k=t}^{T-1} \pi(A_k | S_k) p(S_{k+1} | S_k, A_k)}{\prod_{k=t}^{T-1} b(A_k | S_k) p(S_{k+1} | S_k, A_k)}$$

Las dinámicas del MDP no influyen:

$$\rho_{t:T-1} = \frac{\text{Prob}(\tau_\pi)}{\text{Prob}(\tau_b)} = \frac{\prod_{k=t}^{T-1} \pi(A_k | S_k) \cancel{p(S_{k+1} | S_k, A_k)}}{\prod_{k=t}^{T-1} b(A_k | S_k) \cancel{p(S_{k+1} | S_k, A_k)}} = \frac{\prod_{k=t}^{T-1} \pi(A_k | S_k)}{\prod_{k=t}^{T-1} b(A_k | S_k)}$$

Predicción off-policy con *importance sampling*

Utilizamos ρ para **ponderar las recompensas finales** obtenidas en cada trayectoria.

$\rho = 1$	El valor de G obtenido se mantiene, ya que es igual de probable con b y π .
$\rho > 1$	El valor de G obtenido por b tiene mayor peso, porque es una trayectoria probable con π .
$\rho < 1$	El valor de G obtenido por b se reduce porque es una trayectoria poco probable con π .
$\rho = 0$	El valor de G se anula porque no es una trayectoria que podamos obtener con π .

Predicción *off-policy* con *importance sampling*

Finalmente, lo que tenemos es:

$$\mathbb{E}[\rho_{t:T-1} G_t \mid S_t = s] = v_\pi(s)$$

Predicción off-policy con *importance sampling*

Finalmente, lo que tenemos es:

$$\mathbb{E}[\underbrace{\rho_{t:T-1} G_t}_{\text{Retorno obtenido tras una serie de trayectorias siguiendo } b} \mid S_t = s] = \underbrace{v_\pi(s)}_{\text{Función de valor correspondiente a la política objetivo } \pi}$$

Predicción off-policy con *importance sampling*

Finalmente, lo que tenemos es:

$$\mathbb{E}[\underbrace{\rho_{t:T-1} G_t}_{\text{Retorno obtenido tras una serie de trayectorias siguiendo } b} \mid S_t = s] = \underbrace{v_\pi(s)}_{\text{Función de valor correspondiente a la política objetivo } \pi}$$

Ponderamos la recompensa acumulada obtenida tras una trayectoria en base a su probabilidad.

Ejemplo

La política b interactúa con el entorno durante un episodio y obtiene un retorno $G = 10$.

Si la trayectoria seguida por b es 3 veces **menos** probable que ocurra empleando π , aumentamos $G \times 3$:

$$\pi(a|s) > b(a|s) \text{ (ej. } \times 3) \longrightarrow G = 10 \times 3 = 30$$

Si, por el contrario, es **más** probable que ocurra en b que en π , reducimos el valor de G :

$$\pi(a|s) < b(a|s) \text{ (ej. } \times 0.25) \longrightarrow G = 10 \times 0.25 = 2.5$$

Predicción Monte Carlo de v_π
con *importance sampling*

Predicción MC de v_π con *importance sampling*

Dado un conjunto (*batch*) de episodios observados a partir de una política b , procedemos a estimar v_π . Tenemos dos opciones:

- *Importance sampling ordinario:*

$$V(s) = \frac{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1} G_t}{|\mathcal{T}(s)|}$$

- *Importance sampling ponderado:*

$$V(s) = \frac{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1} G_t}{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1}}$$

Predicción MC de v_π con *importance sampling* ordinario

$$V(s) = \frac{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1} G_t}{|\mathcal{T}(s)|}$$

- t = *time step* donde se visita s .
- $T(t)$ = siguiente terminación de episodio tras t .
- **Retorno obtenido al final de la trayectoria.**
- *Time steps* en los que se ha visitado s .

Predicción MC de v_π con *importance sampling* ordinario

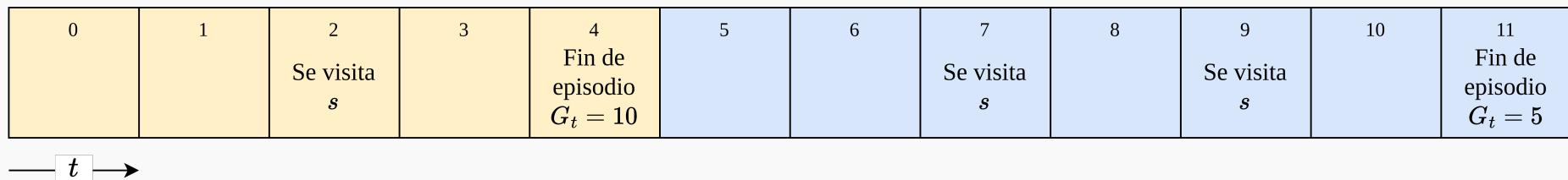
$$V(s) = \frac{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1} G_t}{|\mathcal{T}(s)|}$$

- **First-visit:** $\mathcal{T}(s)$ sólo considera los *time steps* de la primera visita a s en cada episodio.
- **Every-visit:** $\mathcal{T}(s)$ incluye todas las visitas a s .

Ejemplo

$$V(s) = \frac{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1} G_t}{|\mathcal{T}(s)|}$$

Batch de 2 episodios:



First-visit: $\mathcal{T}(s) = \{1, 7\}$

Every-visit: $\mathcal{T}(s) = \{1, 7, 9\}$

$$V(s) = \frac{\rho_{1:3} \cdot 10 + \rho_{7:10} \cdot 5}{2}$$

$$V(s) = \frac{\rho_{1:3} \cdot 10 + \rho_{7:10} \cdot 5 + \rho_{9:10} \cdot 5}{3}$$

Limitaciones

El *importance sampling* ordinario no presenta sesgos (*bias*) en favor de la política de comportamiento b , pero puede ser demasiado extremo.

Por ejemplo, si una trayectoria es $\times 10$ veces más probable bajo π que bajo b , la estimación será diez veces G , que se aleja bastante del realmente observado.

Se plantea como alternativa el *importance sampling* ponderado:

$$V(s) = \frac{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1} G_t}{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1}}$$

Limitaciones

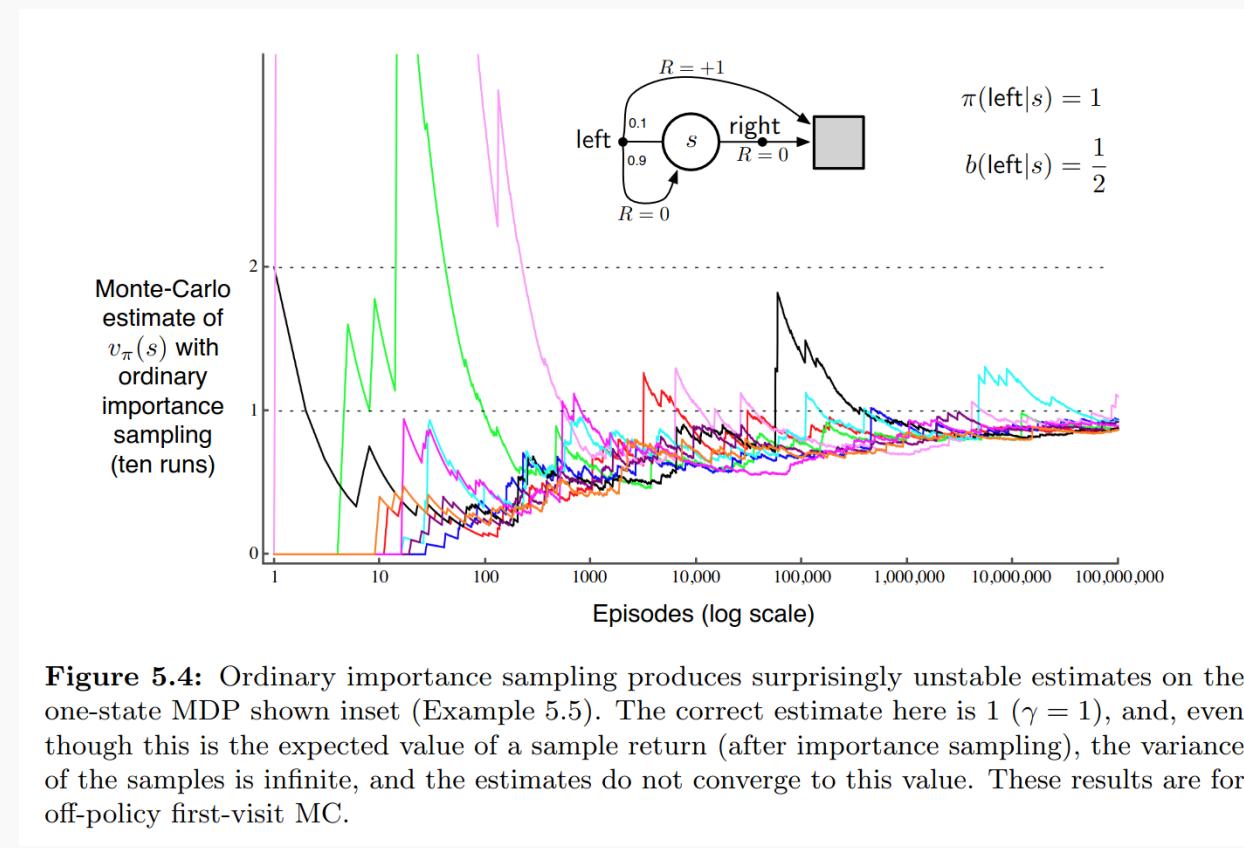


Figure 5.4: Ordinary importance sampling produces surprisingly unstable estimates on the one-state MDP shown inset (Example 5.5). The correct estimate here is 1 ($\gamma = 1$), and, even though this is the expected value of a sample return (after importance sampling), the variance of the samples is infinite, and the estimates do not converge to this value. These results are for off-policy first-visit MC.

Predicción MC de v_π con *importance sampling* ponderado

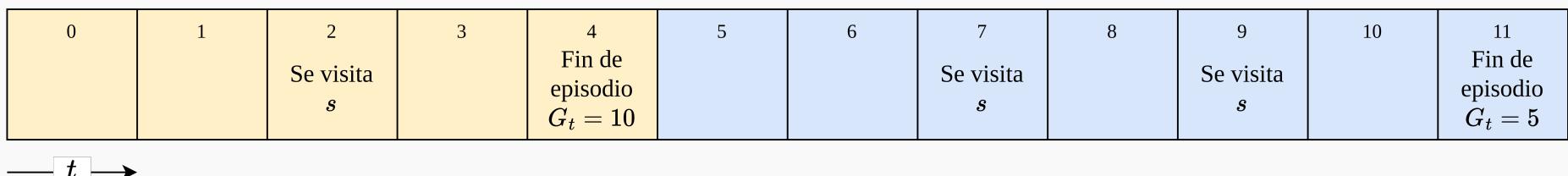
$$V(s) = \frac{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1} G_t}{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1}}$$

- Presenta mayor sesgo:
 - Si sólo se visita un estado una vez, $v_\pi(s) = v_b(s)$, el *importance sampling ratio* se cancela y hay un sesgo hacia la política de comportamiento.
- Sin embargo, la varianza es menor (actualizaciones menos extremas).
- Suele ser una mejor opción

Ejemplo

$$V(s) = \frac{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1} G_t}{\sum_{t \in \mathcal{T}(s)} \rho_{t:T(t)-1}}$$

Batch de 2 episodios:



First-visit: $\mathcal{T}(s) = \{1, 7\}$

Every-visit: $\mathcal{T}(s) = \{1, 7, 9\}$

$$V(s) = \frac{\rho_{1:3} \cdot 10 + \rho_{7:10} \cdot 5}{\rho_{1:3} + \rho_{7:10}}$$

$$V(s) = \frac{\rho_{1:3} \cdot 10 + \rho_{7:10} \cdot 5 + \rho_{9:10} \cdot 5}{\rho_{1:3} + \rho_{7:10} + \rho_{9:10}}$$

Comparativa

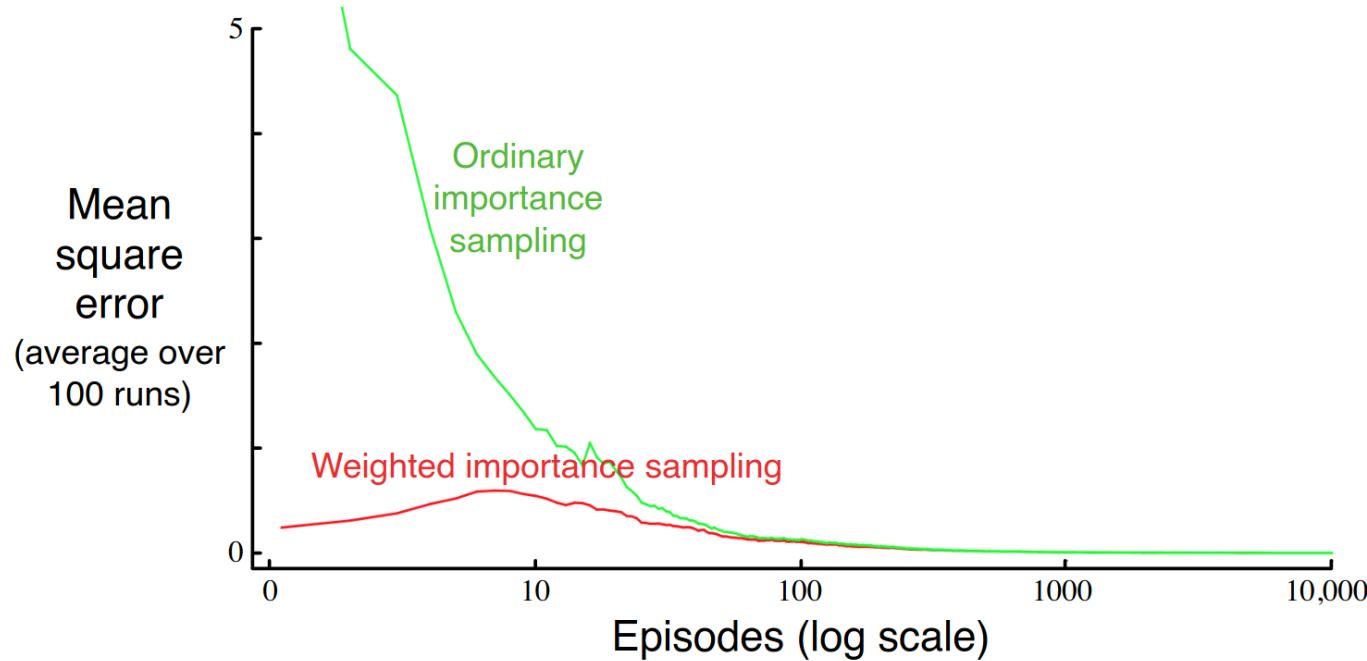


Figure 5.3: Weighted importance sampling produces lower error estimates of the value of a single blackjack state from off-policy episodes. ■

Predicción Monte Carlo incremental

Predicción MC incremental

La predicción Monte Carlo puede realizarse de forma **incremental**.

- Supongamos una secuencia de retornos: G_1, G_2, \dots, G_{n-1} obtenidos partiendo de un mismo estado.
- Cada retorno tiene una ponderación $W_i = \rho_{t_i:T(t_i)-1}$
- Empleando *importance sampling* ponderado tenemos:

$$V_n = \frac{\sum_{k=1}^{n-1} W_k G_k}{\sum_{k=1}^{n-1} W_k}, \quad n \geq 2$$

Predicción MC incremental

$$V_n = \frac{\sum_{k=1}^{n-1} W_k G_k}{\sum_{k=1}^{n-1} W_k}, \quad n \geq 2$$

Buscamos **actualizar incrementalmente** V_n cada vez que se obtiene un nuevo retorno G_n .

- Para cada estado consideramos C_n , que es la suma acumulada de los pesos de los n primeros retornos:

$$C_{n+1} = C_n + W_{n+1}$$

- El cálculo de W_{n+1} es recursivo:

$$W_1 \leftarrow \rho_{T-1}$$

$$W_2 \leftarrow \rho_{T-1}\rho_{T-2}$$

$$W_3 \leftarrow \rho_{T-1}\rho_{T-2}\rho_{T-3}$$

...

$$W_{n+1} \leftarrow W_n \rho_n$$

Predicción MC incremental

Así, la **regla de actualización** empleada es:

$$V_{n+1} = V_n + \frac{W_n}{C_n} [G_n - V_n], \quad n \geq 1$$

V puede ser el valor de un estado o de un par acción-estado.

Si bien esta implementación se corresponde con el algoritmo de **predicción off-policy con importance sampling ponderado**, también se aplica al caso **on-policy** si $\pi = b$ (W es siempre = 1).

Predictión MC incremental

Off-policy MC prediction (policy evaluation) for estimating $Q \approx q_\pi$

Input: an arbitrary target policy π

Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:

$$Q(s, a) \in \mathbb{R} \text{ (arbitrarily)}$$

$$C(s, a) \leftarrow 0$$

Loop forever (for each episode):

$$b \leftarrow \text{any policy with coverage of } \pi$$

Generate an episode following b : $S_0, A_0, R_1, \dots, S_{T-1}, A_{T-1}, R_T$

$$G \leftarrow 0$$

$$W \leftarrow 1$$

Loop for each step of episode, $t = T-1, T-2, \dots, 0$, while $W \neq 0$:

$$G \leftarrow \gamma G + R_{t+1}$$

$$C(S_t, A_t) \leftarrow C(S_t, A_t) + W$$

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{W}{C(S_t, A_t)} [G - Q(S_t, A_t)]$$

$$W \leftarrow W \frac{\pi(A_t | S_t)}{b(A_t | S_t)}$$

Control Monte Carlo *off-policy*

Control Monte Carlo off-policy

Off-policy MC control, for estimating $\pi \approx \pi_*$

Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:

$$Q(s, a) \in \mathbb{R} \text{ (arbitrarily)}$$

$$C(s, a) \leftarrow 0$$

$$\pi(s) \leftarrow \operatorname{argmax}_a Q(s, a) \quad (\text{with ties broken consistently})$$

Loop forever (for each episode):

$$b \leftarrow \text{any soft policy}$$

Generate an episode using b : $S_0, A_0, R_1, \dots, S_{T-1}, A_{T-1}, R_T$

$$G \leftarrow 0$$

$$W \leftarrow 1$$

Loop for each step of episode, $t = T-1, T-2, \dots, 0$:

$$G \leftarrow \gamma G + R_{t+1}$$

$$C(S_t, A_t) \leftarrow C(S_t, A_t) + W$$

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{W}{C(S_t, A_t)} [G - Q(S_t, A_t)]$$

$$\pi(S_t) \leftarrow \operatorname{argmax}_a Q(S_t, a) \quad (\text{with ties broken consistently})$$

If $A_t \neq \pi(S_t)$ then exit inner Loop (proceed to next episode)

$$W \leftarrow W \frac{1}{b(A_t | S_t)}$$

Resumiendo...

Resumiendo...

- Los métodos **Monte Carlo** aprenden funciones de valor y políticas óptimas a partir de experiencia procedente de **episodios muestrados / aleatorios**.

Resumiendo...

- Los métodos **Monte Carlo** aprenden funciones de valor y políticas óptimas a partir de experiencia procedente de **episodios muestrados / aleatorios**.
- Las políticas óptimas se aprenden directamente, **sin requerir un modelo del entorno** ni emplear *bootstrapping* (estimaciones a partir de estimaciones).

Resumiendo...

- Los métodos **Monte Carlo** aprenden funciones de valor y políticas óptimas a partir de experiencia procedente de **episodios muestrados / aleatorios**.
- Las políticas óptimas se aprenden directamente, **sin requerir un modelo del entorno** ni emplear *bootstrapping* (estimaciones a partir de estimaciones).
- El esquema general de **GPI** también se aplica a estos métodos (evaluación + mejora de la política).

Resumiendo...

- Los métodos **Monte Carlo** aprenden funciones de valor y políticas óptimas a partir de experiencia procedente de **episodios muestrados / aleatorios**.
- Las políticas óptimas se aprenden directamente, **sin requerir un modelo del entorno** ni emplear *bootstrapping* (estimaciones a partir de estimaciones).
- El esquema general de **GPI** también se aplica a estos métodos (evaluación + mejora de la política).
- La aproximación de las funciones de valor se realiza en base al **retorno promedio** desde cada estado.

Resumiendo...

- Para garantizar la exploración, empleamos técnicas como **inicios de exploración**, aunque presenta algunas limitaciones.

Resumiendo...

- Para garantizar la exploración, empleamos técnicas como **inicios de exploración**, aunque presenta algunas limitaciones.
- Los métodos ***on-policy*** permiten asegurar la exploración y alcanzar una política muy cercana a la óptima.

Resumiendo...

- Para garantizar la exploración, empleamos técnicas como **inicios de exploración**, aunque presenta algunas limitaciones.
- Los métodos ***on-policy*** permiten asegurar la exploración y alcanzar una política muy cercana a la óptima.
- Los métodos ***off-policy*** emplean dos políticas: una para explorar, y otra para actuar.

Resumiendo...

- Para garantizar la exploración, empleamos técnicas como **inicios de exploración**, aunque presenta algunas limitaciones.
- Los métodos ***on-policy*** permiten asegurar la exploración y alcanzar una política muy cercana a la óptima.
- Los métodos ***off-policy*** emplean dos políticas: una para explorar, y otra para actuar.
- El uso de dos políticas requiere aplicar ***importance sampling*** ordinario o ponderado para que las estimaciones no estén sesgadas.

Aprendizaje por refuerzo

Métodos basados en muestreo (2)

Antonio Manjavacas Lucas

manjavacas@ugr.es