Valor Optimal

$$V^*(s) = \max_{\pi} V^{\pi}(s)$$

Ecs. bellman

$$V^*(s) = \underbrace{R(s)}_{\text{recomp. inmediatas}} + \max_a \underbrace{\gamma \sum P_{sa}(s') V^*(s')}_{\text{recomp. esperadas futuras}}$$

Encontrar π^* (Política Optimal) Sabiendo V^* (fn. Valor optimal)

$$\pi^* = \arg\max_a \gamma \sum_{s'} P_{sa}(s') V^*(s')$$

→ Nos da una acción óptima para cada estado.

Notación: $V^*(s) = V^{\pi^*}(s) \geq V^{\pi}(s)$

V^*: fn. de Valor optimal

V^{π^*}: fn Valor para la pol. optimal

Resumiendo 1. → Encontrar V^*

2 → Usar argmax para encontrar π^*

V^* → algo:

Inicio $V(s) = 0$

y para cada estado, se actualiza.

$$V^* \left\{ V(s) = R(s) + \max_a \left\{ \sum_{s'} P_{sa}(s') V(s') \right\} \right.$$

$$\begin{bmatrix} V(1,1) \\ \vdots \\ V(4,3) \end{bmatrix}$$

$[V[4,3]]$

$\mathbb{R}^{11}$

Actualizando ← Sincronizado

No-Sincronizado

Estados absorbidos $(+1, -1)$

$$V(s) = R(s) + \max_a \left[\sum_{s'} P(s' \mid s, a) V(s') \right]$$

$V(s)$: new values

$V(s')$: Old values

(Sincronizado)

Python → Operador de Bellman

Back up:

$$V := B(V)$$

$$V(s) \rightarrow V^*(s)$$

converge

$k = 9 \longrightarrow$ convergence

Ejemplo = Tenemos

V^* =

0.86	0.9	0.95	+1
0.81		0.69	−1
0.78	0.75	0.71	0.49

π =

3	→	→	→	+1
2	↑		↑	−1
1	↑	←	←	←
	1	2	3	4

y, x

(x, y)

$(3, 1)$

" ← "

$$\sum_{s'} P_{sa}(s') V^*(s') = 0.8 \cdot 0.75 + 0.1 \cdot 0.69 + 0.1 \cdot 0.49 = 0.71$$

V^*

Espacio de acciones

$$\{n, s, e, o\}$$

Estados 3,1 $\{n, e, o\}$

Conclusiones

V^* → π^*

V^π Sist. Ec. lineal 11 incog 11 ecs.

Si tes $V^\pi \Rightarrow V^* \Rightarrow \pi^*$

Iteración de la Política

(Policy Iteration)

→ Iniciar con π aleatoria

Repeat

↳ Set $V = V^\pi$ [Ecs Bellman]

↳ Set $\pi(s) = \boxed{\arg\max_a \sum_{s'} P_a(s') V(s')}$

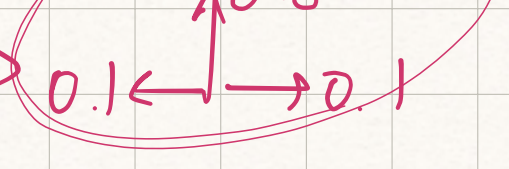
until $\pi = \pi'$

It. II → + lento, + ilado
→ espacio de acciones
→ π^*

It. V → Converge a $V^* \to \pi^*$
→ + usado V, + rápida

Qué pasa si no se $P_{sa}(s')$?

$P_{sa}(s') = ?$

$P_{sa}(s') \Rightarrow$ 

$P_{sa}(s') =$ #veces q' el robot tome la acción a en estado s y llego a s'
#veces tomo acción a en estado s

Prob. de transición.

Algo => Repeat {

E-greedy
90% π
10% Random

pas (1) Toma acción a con respecto (π)

- Actualize estimaciones de $P_{sa}(s')$

(act. estimaciones de R)

- Resuelve Ecs de Bellman para la Iteración

- Actualizo

$$\pi(s) = \arg\max_a \sum_{s'} P_{sa}(s') V(s')$$

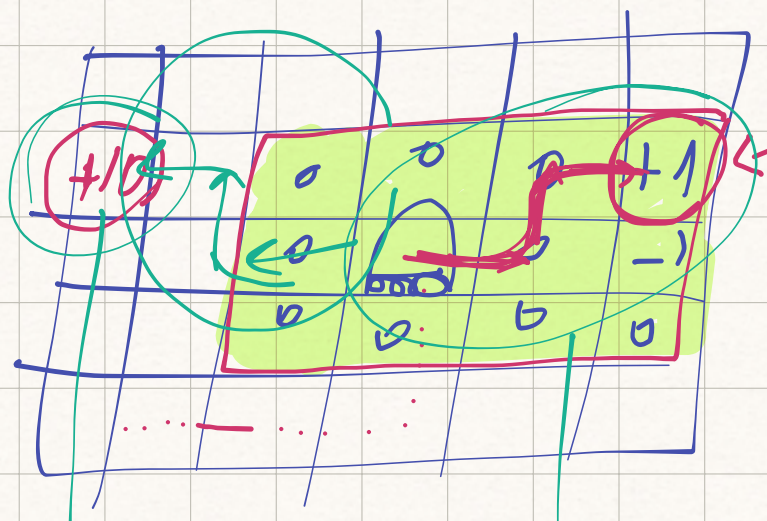
}

Cuando no se cumple de Recomp

→ Algo RL Acciones de Valor

→ Algo RL AA

Dilema de la Exploración vs Explotación



$V^* \rightarrow \pi^*$

(greedy)

Solo explotar

óptimo global

óptimo local