CLASE 3 - PROGRAMACIÓN DINÁMICA

C3 H1

$$v_*(s) = \max_a \mathbb{E}\left[R_{t+1} + \gamma v_*(S_{t+1}) \mid S_t = s, A_t = a\right]$$

(v_* ... q_*)

$$= \max_a \sum_{s',r} \left[r + \gamma v_*(s')\right] p(s', r \mid s, a) \qquad q_*(s,a)$$

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

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

¿Cómo obtener q_*, v_*, π_*?

POLICY EVALUATION (PREDICTION) v_π?

Menú 1. Vimos ecuaciones lineales (⚠ si $|S| \nearrow$)

Método iterativo: $v_{k+1}(s) := \mathbb{E}_\pi\left[R_{t+1} + \gamma v_k(S_{t+1}) \mid S_t = s\right]$

Esto quizás se podría poner en slides recordando la ecuación lineal!

$$= \sum_a \pi(a \mid s) \sum_{s',r} p(s', r \mid s, a)\left[r + \gamma v_k(s')\right]$$

Obs: $v_k \equiv v_\pi$ es el punto fijo! ($\gamma < 1$ o desde todos los estados, bajo π, se llega a un estado terminal)

Obs 2: Podría ir updateando $v_k(s')$ por $v_{k+1}(s')$ a medida que voy calculando!

HACER SLIDE con pseudocódigo y ¿Ejemplo 4.1 Gridworld?

Scanned by CamScanner

Mejorando una política π (o hasta encerr y pulir ☺)

C3 H2

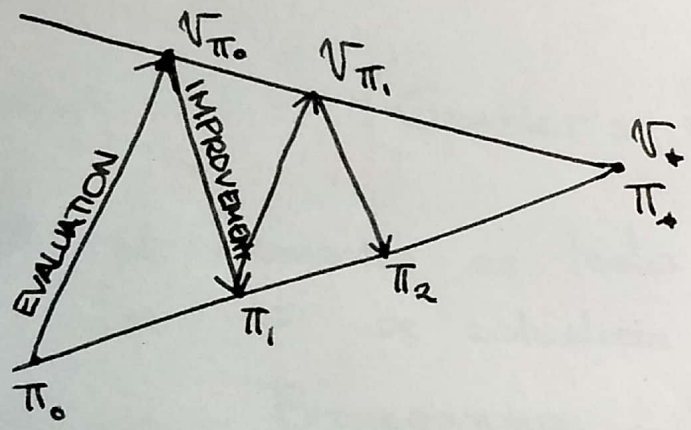
HACER UN SLIDE !!!

$$\pi_0 \xrightarrow{E} v_{\pi_0} \xrightarrow[\text{(ECUACIÓN } q_\pi \text{ mismo } v_\pi)]{} q_{\pi_0} \xrightarrow{I} \pi_1(s) := \arg\max_a q_{\pi_0}(s,a)$$

greedy ( )

Recordemos $v_\pi(s) = \sum_a \left[R_s^a + \sum_{s'} v_\pi(s') p_{ss'}^a \right] \pi(a|s)$

DOS "INGREDIENTES": v_π y π. Busco $v^* \equiv \max_\pi v_\pi \equiv v_{\pi^*}$



TEOREMA: π', π determinísticas / $q_\pi(s, \pi'(s)) \geq v_\pi($

$$\Rightarrow v_{\pi'}(s) \geq v_\pi(s)$$

Obs: $\pi'(s) := \arg\max_a q_\pi(s,a)$ $\left(q_\pi(s,a) = R_s^a + \gamma \sum_{s'} v_\pi(s') p_{ss'}^a \right)$

Como π determinística $v_\pi(s) = q_\pi(s, \pi(s))$

$$v_\pi(s) \leq q_\pi(s, \pi'(s)) = E\left[R_{t+1} + \gamma v_\pi(S_{t+1}) \mid S_t = s, A_t = \pi'(s) \right]$$

$$\equiv E_{\pi'}\left[R_{t+1} + \gamma v_\pi(S_{t+1}) \mid S_t = s \right]$$

Como π' determinística $\Rightarrow A_t = \pi'(S_t)$

$$\leq E_{\pi'}\left[R_{t+1} + \gamma q_\pi(S_{t+1}, \pi'(S_{t+1})) \mid S_t = s \right]$$

$$\leq E_{\pi'}\left[R_{t+1} + \gamma R_{t+2} + \gamma^2 v_\pi(S_{t+2}) \mid S_t = s \right] \leq \ldots \leq v_{\pi'}(s$$

Scanned by CamScanner

C3
H3

Obs: Si $q_\pi(s, \pi'(s)) = v_\pi(s)$

$$\Rightarrow v_\pi(s) = \arg\max_a q_\pi(s,a)$$

- Ejemplo en slides del car rental.

VALUE ITERATION: Variante para reducir policy EVALUATION

- Se utiliza sólo 1 paso de update cuando hacemos la evaluación de la policy. Combina una de evaluation y una de improvement

Recordar: $$v_{k+1}(s) = \max_a \sum_{s',r} [r + \gamma v_k(s')] p(s', r | s, a) \quad (4.10)$$

c/iteración puede NO corresponder a una VALUE FUNCTION!

Ejemplo 4.3 (Gambler's Problem) En slides!

- Hasta el momento es todo SINCRÓNICO: Todos los s se actualizan en SIMULTÁNEO. (Meter el slide donde hace el summary de los algoritmos)

x ejemplo: Backgammon $\simeq 10^{20} = |S|$

ASINCRÓNICO (capítulo 8)

- Usar 4.10 actualizando 1 sólo s en c/iteración.
- Actualizar algunos estados s en evaluation y otros en improvement.
- No actualizar estados poco probables!

Scanned by CamScanner

C3
H4

¿POR QUÉ CONVERGEN ESTAS ITERACIONES?

BELLMAN EXPECTATION BACKUP OPERATOR $T^\pi(v) := R^\pi + \gamma P^\pi v$

$$\|T^\pi(u) - T^\pi(v)\|_\infty = \|\gamma P^\pi (u - v)\|_\infty$$

$$\leq \gamma \|u - v\|_\infty$$

COMO $\gamma < 1$ es un contracción! $\Rightarrow$ CONTRACTION MAPPING THM.

(T^π tiene ! punto fijo)

Idem con $T^*(v) := \max_a R^a + \gamma P^a v$

Scanned by CamScanner