Processus de décision markovien

Cours 10: Décisions séquentielles - PDM

Stéphane Airiau

Université Paris-Dauphine

Définition (Processus décisionnel de Markov)

Un *Processus décisionnel de Markov* est un tuple $\langle S, A, T, R, \gamma \rangle$ où

- S est un ensemble fini d'états
- A est un ensemble fini d'actions
- T est une matrice de transition
  $T^a_{ss'} = \mathbb{P}\left[S_{t+1} = s' \mid S_t = s, A_t = a\right]$ probabilité d'arriver dans l'état s' à l'instant +1 quand on a pris l'action a dans l'état s à l'instant t
- R est le vecteur de récompenses
  $R^a_s = \mathbb{E}\left[R_{t+1} \mid S_t = s, A_t = a\right]$ valeur moyenne obtenue après avoir pris l'action a dans l'état s
- un ensemble d'état initial
- parfois un ensemble d'états terminaux

Les composants d'un agent : politique

C'est ce qui gouverne le comportement de l'agent

politique déterministe : La fonction associe à chaque état **une action**

$$\pi : S \mapsto A$$

3	→	→	→	+1
2	↑		↑	-1
1	START ↑	→	↑	←
	1	2	3	4

politique optimale pour un pénalité de 0.03 par déplacement

Focus pour le cours

- problèmes itératifs en continue.
- objectif : maximiser la somme "avec dévaluation" $G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}$
  - Pour éviter une récompense infinie si on tombe dans des cycles
  - Le futur reste uncertain ! Bon compromis entre court et long terme
  - Tendance naturelle vers le court terme
  - Mathématiquement, c'est quand même pratique !
- la fonction de transition est stochastique
- la fonction de récompense est connue

Comment trouver la meilleure politique ?

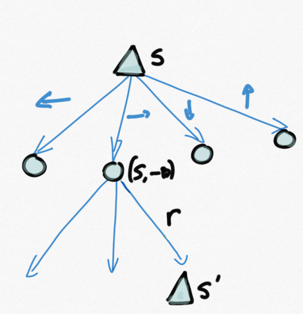
Evaluation

On va s'aider de deux quantités

$v^\star(s)$ quelle est la valeur de me trouver dans l'état s puis de continuer avec la politique optimale

$q^\star(s,a)$ quelle est la valeur de prendre l'action a dans l'état s puis de continuer avec la politique optimale

$\pi^\star(s)$ politique optimale pour l'état s (i.e. quelle est la meilleure action).



Valeur optimale d'un état $v^\star(s)$

- on calcule la valeur espérée en supposant qu'on suive la politique optimale
- on prend la moyenne pondérée des récompenses escomptée

➩ comme dans expectimax !

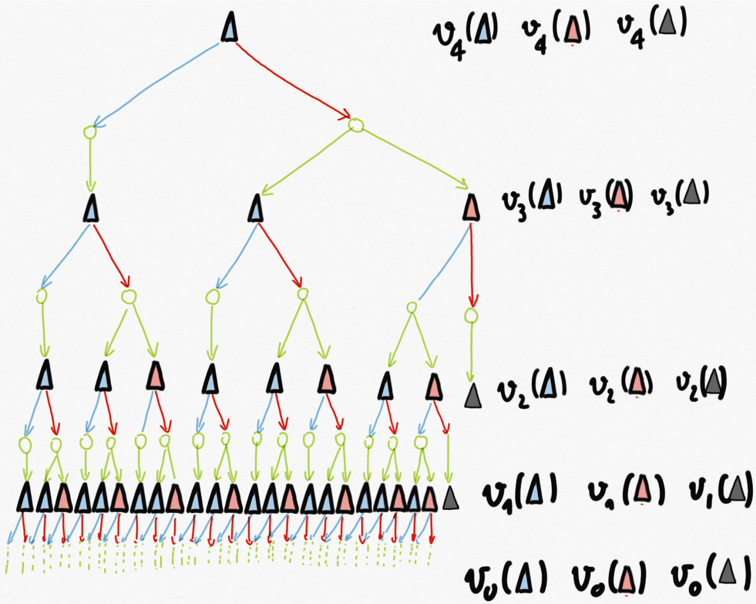
$$v^\star(s) = \max_{a \in A} q^\star(s,a)$$

$$q^\star(s,a) = R_s^a + \gamma \sum_{s' \in S} T_{ss'}^a v^\star(s')$$

donc

$$v^\star(s) = \max_{a \in A} \left[R_s^a + \gamma \sum_{s' \in S} T_{ss'}^a v^\star(s') \right]$$

Idée d'itération sur les valeurs



Value Iteration

```
for each s ∈ S and k ∈ ℕ
    V_k(s) ← 0

repeat for k = 0 to ...

   for each s ∈ S

      V_{k+1}(s) ← max_{a∈A} [ R_s^a + γ Σ_{s'∈S} T_{ss'}^a V_k(s') ]        /* mise à jour */

until convergence
```

$$V_{k+1}(s) \leftarrow \max_{a \in A}\left[R_s^a + \gamma \sum_{s' \in S} T_{ss'}^a V_k(s')\right]$$

Value Iteration – plus efficace pour la mémoire

```
for each s ∈ S
    V(s) ← 0

repeat

    for each s ∈ S

        V(s) ← max_{a∈A} [ R_s^a + γ Σ_{s'∈S} T_{ss'}^a V(s') ]        /* mise à jour */

until convergence
```

$$V(s) \leftarrow \max_{a \in A} \left[R_s^a + \gamma \sum_{s' \in S} T_{ss'}^a V(s') \right]$$

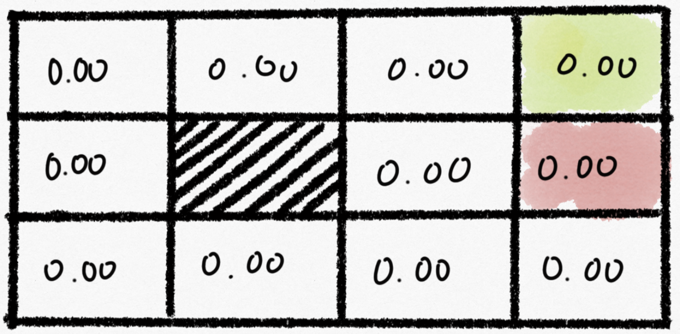
Value Iteration – avec test de convergence

for each $s \in S$
$\quad V(s) \leftarrow 0$

repeat
$\quad \Delta \leftarrow 0$ /* mesure le plus grand changement */
$\quad$ **for each** $s \in S$
$\quad\quad v \leftarrow V(s)$ /* sauvegarde l'ancienne valeur pour mesurer le changement */
$\quad\quad V(s) \leftarrow \max_{a \in A} \left[R_s^a + \gamma \sum_{s' \in S} T_{ss'}^a V(s') \right]$ /* mise à jour */
$\quad\quad \Delta \leftarrow \max(\Delta, |v - V(s)|)$ /* mise à jour du plus grand changement*/
until $\Delta < \epsilon$ /* test convergence */

- On n'a pas de politique explicite
- On a un théorème de convergence

Itération sur les valeurs : $k = 0$



$$r = 0$$
$$\gamma = 0.9$$
$$bruit = 0.2$$

Itération sur les valeurs : $k = 1$

0.00	0.00	0.00	+1.00
0.00		0.00	−1.00
0.00	0.00	0.00	0.00

$$r = 0$$
$$\gamma = 0.9$$
$$bruit = 0.2$$

Itération sur les valeurs : $k = 2$

0.00	0.00	0.72	+1.00
0.00		0.00	−1.00
0.00	0.00	0.00	0.00

$$r = 0$$
$$\gamma = 0.9$$
$$bruit = 0.2$$

Itération sur les valeurs : $k = 3$

0.00	0.52	0.78	+1.00
0.00		0.43	−1.00
0.00	0.00	0.00	0.00

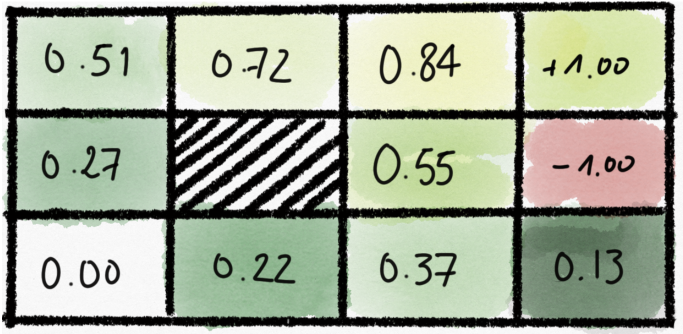
$$r = 0$$
$$\gamma = 0.9$$
$$bruit = 0.2$$

Itération sur les valeurs : $k = 4$

0.37	0.66	0.83	+1.00
0.00		0.51	−1.00
0.00	0.00	0.31	0.00

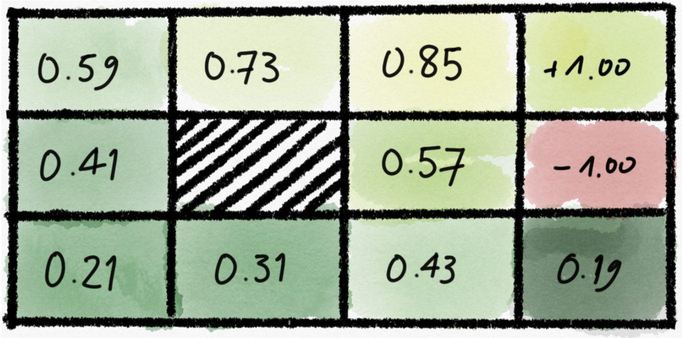
$$r = 0$$
$$\gamma = 0.9$$
$$bruit = 0.2$$

Itération sur les valeurs : $k = 5$



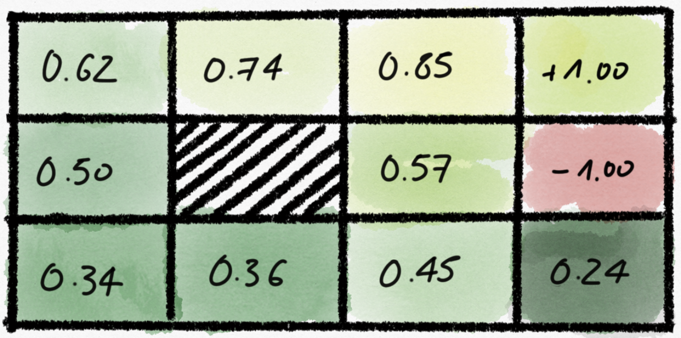
$$r = 0$$
$$\gamma = 0.9$$
$$bruit = 0.2$$

Itération sur les valeurs : $k = 6$



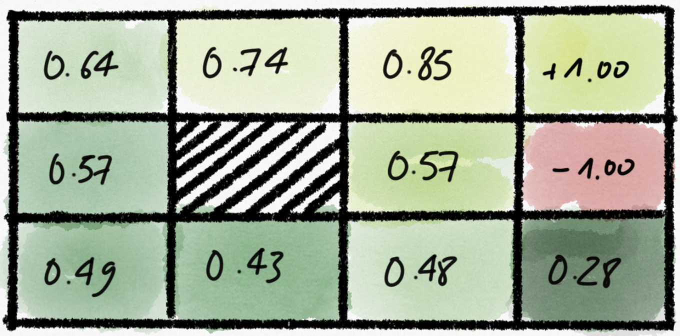
$$r = 0$$
$$\gamma = 0.9$$
$$bruit = 0.2$$

Itération sur les valeurs : $k = 7$



$$r = 0$$
$$\gamma = 0.9$$
$$bruit = 0.2$$

Itération sur les valeurs : $k = 100$



$$r = 0$$
$$\gamma = 0.9$$
$$bruit = 0.2$$

Limitations de Value Iteration

- la méthode est lente
- la valeur du max change rarement
- ⇨ pourtant c'est cela qui va aider à changer toutes les valeurs !
- les valeurs peuvent mettre longtems à converger exactement alors que la politique, elle, est déjà optimale depuis longtemps

⇨ on va essayer de travailler sur la politique.

Fonctions de valeurs

Similairement à la fonction optimale, on peut définir deux fonctions de valeurs pour une politique fixée π.

Définition (fonction de valeurs pour les états)

La *fonction de valeurs pour les états* $v_\pi(s)$ d'un PDM est la valeur espérée de gains en partant dans l'état s et en poursuivant la politique π.

Définition (fonction de valeurs pour les paires (état, action))

La *fonction de valeurs pour les paires état-actions* $q_\pi(s,a)$ d'un PDM est la valeur espérée de gains en partant dans l'état s, en effectuant l'action a *puis* et en poursuivant la politique π.

Equation de Bellman : pour les états

Dans l'état s

on tire notre action avec la politique $\pi(s)$

pour chaque action, on choisit l'action a avec la probabilité $\pi(a|s)$,

on va effectuer l'action a puis continuer avec π dans l'état suivant

⇨ on peut utiliser q_π !

$$
\begin{aligned}
v_\pi(s) &= \mathbb{E}_\pi \left[r_{t+1} + \gamma v_\pi(S_{t+1}) \mid S_t = s \right] \\
&= \sum_{a \in A} \pi(a|s) q_\pi(s,a)
\end{aligned}
$$

Equation de Bellman : pour les actions

Similairement pour la fonction de valeurs pour les actions

- le modèle de récompense nous donne la récompense pour avoir effectué l'action a dans l'état s.
- le modèle de transition nous donne l'état suivant s'

➩ dans ce nouvel état s', on peut utiliser v_π !

$$
\begin{aligned}
q_\pi(s,a) &= \mathbb{E}_\pi\left[r_{t+1} + \gamma q_\pi(S_{t+1}, A_{t+1}) \mid S_t = s, A_t = a\right] \\
&= R_s^a + \gamma \sum_{s' \in S} T_{ss'}^a v_\pi(s')
\end{aligned}
$$

Equation de Bellman

On a établi :

$$v_\pi(s) = \sum_{a \in A} \pi(a|s) q_\pi(s,a)$$

$$q_\pi(s,a) = R_s^a + \gamma \sum_{s' \in S} T_{ss'}^a v_\pi(s')$$

Ensemble on obtient :

$$v_\pi(s) = \sum_{a \in A} \pi(a|s) \left(R_s^a + \gamma \sum_{s' \in S} T_{ss'}^a v_\pi(s') \right)$$

$$q_\pi(s,a) = R_s^a + \gamma \sum_{s' \in S} T_{ss'}^a \sum_{a' \in A} \pi(a'|s') q_\pi(s',a')$$

Equation de Bellman

$$
\begin{aligned}
v_\pi(s) &= \sum_{a\in A} \pi(a|s) \left(R_s^a + \gamma \sum_{s'\in S} T_{ss'}^a v_\pi(s') \right) \\
&= \sum_{a\in A} \pi(a|s) R_s^a + \gamma \sum_{a\in A} \sum_{s'\in S} \pi(a|s) T_{ss'}^a v_\pi(s') \\
&= \sum_{a\in A} \pi(a|s) R_s^a + \gamma \sum_{s'\in S} \sum_{a\in A} \pi(a|s) T_{ss'}^a v_\pi(s') \\
&= R_s^\pi + \gamma \sum_{s'\in S} T_{ss'}^\pi v_\pi(s')
\end{aligned}
$$

On peut donc écrire l'expression vectorielle

$$v_\pi = R^\pi + \gamma T^\pi v_\pi$$

Evaluation d'une politique fixée π

On peut utiliser l'équation de Bellman comme une règle de mise à jour :

$$v_{k+1}(s) \leftarrow \sum_{a\in A} \pi(a \mid s) \left(R_s^a + \gamma \sum_{s'\in S} T_{ss'}^a v_k(s') \right)$$

Theorème

La séquence $\{v_k\}_{k\in\mathbb{N}}$ converge vers v^π.

Seconde idée : sans les max, les équations de Bellman forment un système linéaire à résoudre !

Convergence plus rapide

Pour réaliser l'algorithme :

- avoir deux vecteur v_{old} et v_{new}
- calculer completement v_{new} à partir de v_{old}

➩ "full back up"

On peut aussi n'utiliser qu'*un seul* vecteur

- on remplace directement l'ancienne entrée par la nouvelle
- le vecteur v contient à la fois des nouvelles et des anciennes valeurs

➩ on utilise les nouvelles valeurs au plus vite
convergence toujours garantie et plus rapide
l'ordre de mise à jour joue un rôle sur la vitesse de convergence.

Critère d'arrêt de l'algorithme

- garantie de convergence à la limite
- en pratique, on peut arrêter avant
  par exemple : $\max_{s \in S} |v_{k+1}(s) - v_k(s)| < \epsilon$ pour une valeur de ϵ donnée.

Comment déterminer une bonne action à partir de v ?

Supposons qu'on connaise les valeurs optimales $v^\star(s)$

0.95	0.96	0.98	+1.00
0.94		0.89	-1.00
0.92	0.91	0.90	0.80

Comment devons-nous agir ?

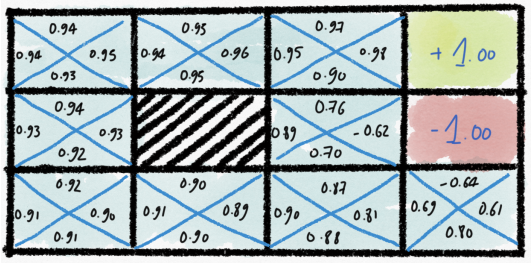
On doit faire une étape d'expectimax !

$$\pi^\star(s) = arg\max_a \left[R_s^a + \gamma \sum_{s'} T_{ss'}^a v^\star(s') \right]$$

⇨ on peut appeler cette étape faire une extraction de politique.

Comment déterminer une bonne action à partir de q ?

Supposons qu'on connaise les valeurs optimales $q^\star(s,a)$



Comment devons-nous agir ?

Trivial, on choisit la meilleure action ! (ou une des meilleurs actions).

$$\pi^\star(s) = arg \max_a q*(s,a)$$

➩ morale de l'histoire : les actions sont plus faciles à obtenir à partir de q qu'à partir de v ! !

Améliorer une politique

On peut essayer d'améliorer la politique en se comportant de manière "gloutonne"

Une fois v_π évaluée : on peut calculer $q^\pi(s,a) = R_s^a + \gamma \sum_{s' \in S} T_{ss'}^a v_\pi(s')$

- si $q^\pi(s,a) > v_\pi(s)$: on a trouvé une amélioration ![a]
- ➩ on peut regarder tous les états $s \in S$ et mettre à jour la politique $\pi'(s) = arg\max_{a \in A} q_\pi(s,a)$

Si aucune amélioration n'est trouvée, on a donc $v_\pi = v_{\pi'}$
➩ $v_{\pi'} = \max_{a \in A} R_s^a + \gamma \sum_{s' \in S} T_{ss'}^a v_{\pi'}(s')$
On reconnait là l'équation de Bellman pour la fonction de valeurs *optimale*
On a donc trouvé $v^\star = v_{\pi'}$!

a. il faut une petite démonstration sur ce point

Itération de politique

L'idée est donc d'alterner

1- l'évaluation d'une politique

2- l'amélioration de la politique

*jusqu'à ce qu'*on converge vers une politique qui sera la politique optimale.

Pour les politiques deterministes, il y a un nombre fini de politiques, on va converger en un nombre fini d'itérations.

Variantes : quand arrêter l'évaluation ?

- convergence à un ϵ près
- après k itérations (k a une petite valeur)
- pourquoi pas après chaque itération ?

Comparaison

- Les deux méthodes calcule le même résultat : à la fin, on a les mêmes valeurs optimales ($v^\star$ et $q^\star$)
- Itération sur les valeurs
  - la politique est implicite. A chaque itération, on met à jour les valeurs
  - si on travaille avec v, on doit utiliser l'extraction d'une politique pour obtenir la politique optimale.
- Iteration sur les politiques
  - plusieurs itérations pour mettre à jour les valeurs d'une politique fixe (mais pour chaque itération, on ne considère qu'une seule action, ce qui devrait être rapide).
  - on met à jour la politique, on doit comparer toutes les actions (peut être lent)
  - soit on améliore la politique, soit on a terminé !