Apprentissage par Renforcement, morceaux choisis

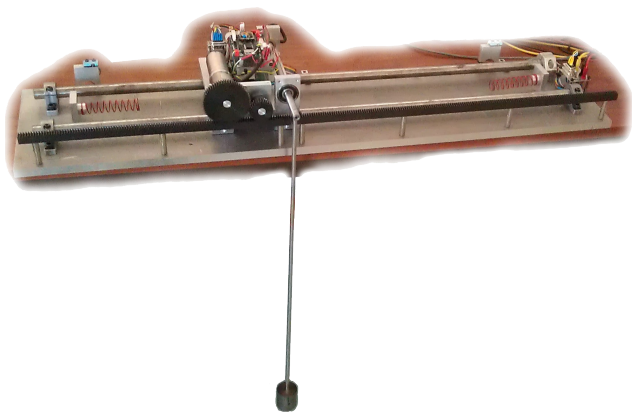
Emmanuel Rachelson
Stage MAAMI'2012

ISAE — SUPAERO

10 mai 2012

Comment feriez-vous pour . . .

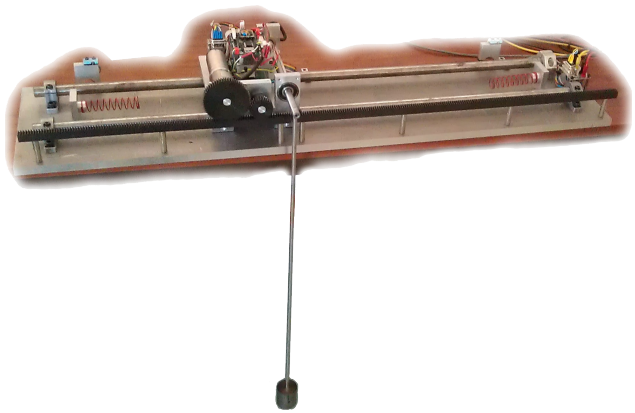
. . . faire tenir droit ce pendule ?



Comment feriez-vous pour . . .

. . . faire tenir droit ce pendule ?

Le faire taper alternativement sur un gong à droite et à gauche ?

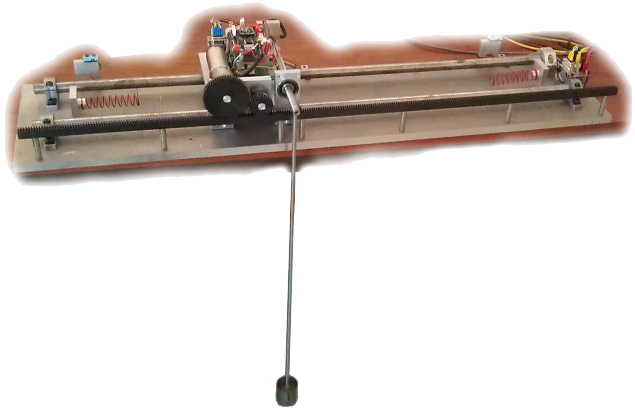


Comment feriez-vous pour . . .

. . . faire tenir droit ce pendule ?

Le faire taper alternativement sur un gong à droite et à gauche ?

En évitant de toucher un obstacle ?



Comment feriez-vous pour . . .

. . . faire tenir droit ce pendule ?
Le faire taper alternativement sur un gong à droite et à gauche ?
En évitant de toucher un obstacle ?

. . . optimiser le temps d'attente des ascenseurs à chaque étage ?

Comment feriez-vous pour . . .

. . . faire tenir droit ce pendule ?
Le faire taper alternativement sur un gong à droite et à gauche ?
En évitant de toucher un obstacle ?

. . . optimiser le temps d'attente des ascenseurs à chaque étage ?

. . . évaluer une stratégie d'investissement sur 10 ans ?

Comment feriez-vous pour . . .

. . . faire tenir droit ce pendule ?
Le faire taper alternativement sur un gong à droite et à gauche ?
En évitant de toucher un obstacle ?

. . . optimiser le temps d'attente des ascenseurs à chaque étage ?

. . . évaluer une stratégie d'investissement sur 10 ans ?
Décider sur quels placements investir ?

Décider sans modèle

Un petit problème de modélisation peut-être ?

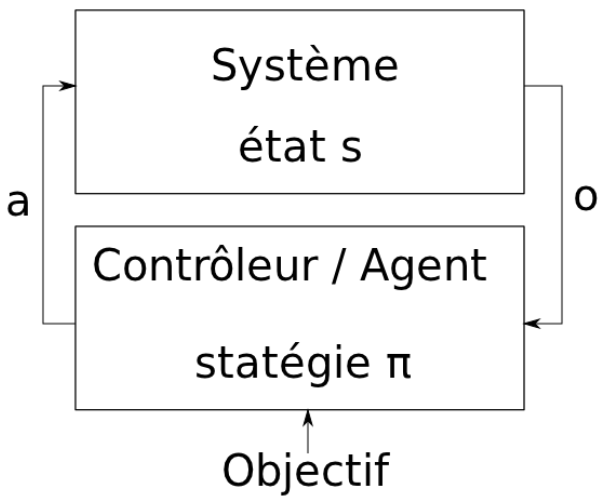
Décider sans modèle

Un petit problème de modélisation peut-être ?

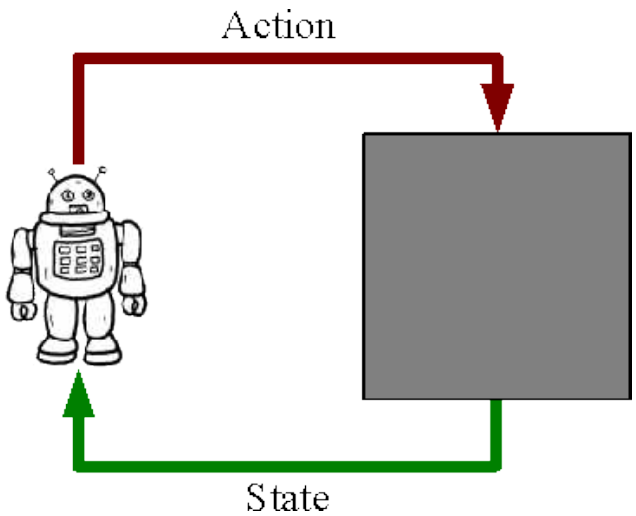
- Problèmes non-linéaires
- Problèmes bruités / stochastiques
- Pas de modèle disponible

Comment peut-on évaluer / contrôler le système quand même ?

Un peu de modélisation

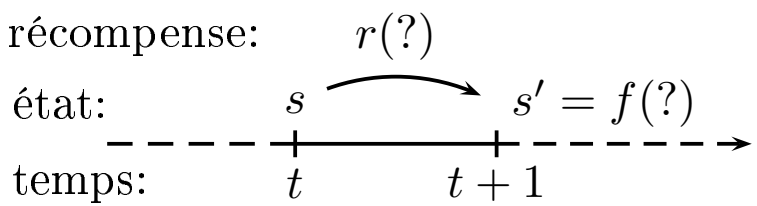


/



Les ingredients

- Ensemble T de *pas de temps*.
- Ensemble S d'*états* s possibles pour le système.
- Ensemble A d'*actions* a possibles pour l'agent.
- Dynamique de *transition* du système $s' \leftarrow f(?)$.
- Signal de *récompense* (de renforcement) par pas de temps $r(?)$.

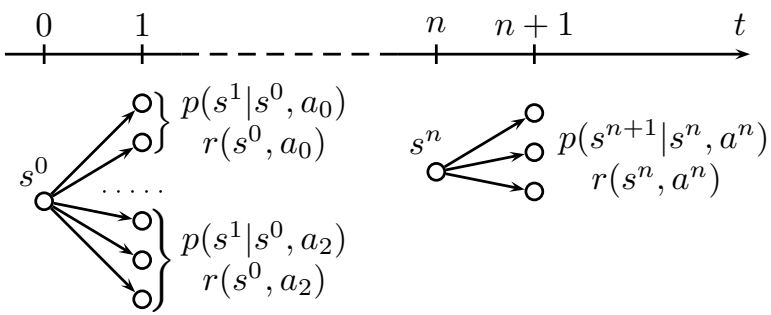


Processus Décisionnels de Markov

Processus Décisionnels de Markov (MDP)

Quintuplet $\langle S, A, p, r, T \rangle$
Modèle de transition Markovien $p(s'|s, a)$
Modèle de récompense $r(s, a)$
Ensemble T de périodes de décision $\{0, 1, \ldots, H\}$

Horizon infini: $H \to \infty$

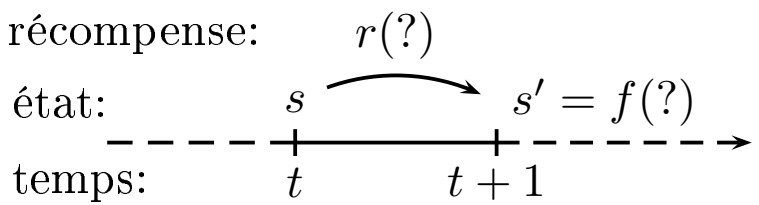


Processus Décisionnels de Markov

Processus Décisionnels de Markov (MDP)

Quintuplet $\langle S, A, p, r, T \rangle$
Modèle de transition Markovien $p(s'|s, a)$
Modèle de récompense $r(s, a)$
Ensemble T de périodes de décision $\{0, 1, \ldots, H\}$

Horizon infini: $H \to \infty$



Qu'est-ce qu'une stratégie / un contrôleur ?

Politique

Une politique π est une séquence de règles de décision δ_t : $\pi = \{\delta_t\}_{t \in \mathbb{N}}$,

avec $\delta_t : \begin{cases} S^{t+1} \times A^t & \rightarrow & \mathscr{P}'(A) \\ h & \mapsto & \delta_t(a|h) \end{cases}$

$\delta_t(a|h)$ indique

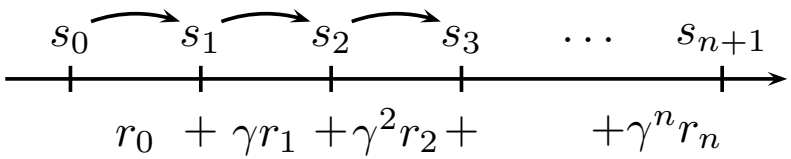
la distribution sur l'action a

à entreprendre à l'étape t, étant donné

l'historique d'états/actions h.

Evaluer une séquence d'actions / une stratégie ?

Comportement sur le long terme



Critère γ-pondéré

Espérance de gain cumulé depuis l'état s, en appliquant π.

$$V^{\pi}(s) = \mathbb{E}\left(\lim_{H\to\infty} \sum_{t=0}^{H} \gamma^t r_t \middle| s_0 = s, \pi\right)$$

N.B. : d'autres critères existent.

Politique optimale

Politique optimale

Une politique π^* est dite optimale si $\pi^* \in \underset{\pi}{\text{argmax}}\, V^\pi$.

Une politique est optimale si elle *domine* toute autre politique :

$$\pi^* \text{ est optimale} \Leftrightarrow \forall s \in S,\ \forall \pi,\ V^{\pi^*}(s) \geq V^\pi(s)$$

Les problèmes de l'apprentissage par renforcement

Quand on ne connait pas p et r, comment peut-on :

- Evaluer V^π ?
- Trouver $\pi^* \in \underset{\pi}{\text{argmax}}\, V^\pi$?

Pour répondre à ces questions on va chercher à caractériser V^π et π^*.

Un résultat fondamental

Espace de recherche des politiques optimales

Pour un MDP avec critère γ-pondéré et horizon infini, parmi les politiques optimales, il en existe une qui est *déterministe*, *Markovienne* et *stationnaire*.

- Markovienne :
  $\forall (s_i, a_i) \in (S \times A)^{t-1}$
  $\forall (s'_i, a'_i) \in (S \times A)^{t-1}$ $, \delta_t(a|s_0, a_0, \ldots, s_t) = \delta_t(a|s'_0, a'_0, \ldots, s_t).$
  On note alors $\delta_t(a|s)$.
- Stationnaire : $\forall (t, t') \in \mathbb{N}^2, \delta_t = \delta'_t$.
  On note alors $\pi = \delta_0$.
- Déterministe : $\delta_t(a|h) = \begin{cases} 1 \text{ pour un unique } a \\ 0 \text{ sinon} \end{cases}$.

Une idée simple

Que vaut a suivie de π ?

$$
\begin{aligned}
Q^{\pi}(s,a) &= \mathbb{E}\left(\sum_{t=0}^{\infty}\gamma^{t} r\left(s_t,a_t\right)\middle| s_0=s, a_0=a, \pi\right)\\
&= r(s,a)+\mathbb{E}\left(\sum_{t=1}^{\infty}\gamma^{t} r\left(s_t,a_t\right)\middle| s_0=s, a_0=a, \pi\right)\\
&= r(s,a)+\gamma\sum_{s'\in S} p\left(s'|s,a\right)\mathbb{E}\left(\sum_{t=1}^{\infty}\gamma^{t-1} r\left(s_t,a_t\right)\middle| s_1=s', \pi\right)\\
&= r(s,a)+\gamma\sum_{s'\in S} p\left(s'|s,a\right) V^{\pi}\left(s'\right)
\end{aligned}
$$

La meilleure action à 1 coup, étant donné π, est celle qui maximise Q^{π}.

- Pour améliorer des politiques, on a intérêt à calculer Q^{π} plutôt que V^{π} et à choisir l'action "gloutonne", puisqu'on ne connait pas p et r.
- $V^{\pi}(s) = Q^{\pi}(s,\pi(s))$.

Equation d'évaluation

Equation d'évaluation

V^π est solution du système linéaire :

$$V^\pi(s) = r(s, \pi(s)) + \gamma \sum_{s' \in S} p\left(s' | s, \pi(s)\right) V^\pi\left(s'\right)$$

$$V^\pi = r^\pi + \gamma P^\pi V^\pi = T^\pi V^\pi$$

De même :

$$Q^\pi(s, a) = r(s, a) + \gamma \sum_{s' \in S} p\left(s' | s, a\right) Q^\pi\left(s', \pi\left(s'\right)\right)$$

$$Q^\pi = r + \gamma P Q^\pi = T^\pi Q^\pi$$

Résoudre ce système :

- Inversion matricielle
- V^π est point fixe de T^π

Caractérisation des politiques optimales (1/3)

On note $V^* = \max_\pi V^\pi$ et $Q^* = \max_\pi Q^\pi$.

On a $Q^*(s,a) = r(s,a) + \gamma \sum_{s' \in S} p(s'|s,a)\, V^*(s')$.

Si π^* est une politique optimale, alors $V^*(s) = Q^*(s, \pi^*(s))$.

Politique gloutonne optimale

Toute politique π définie par $\pi(s) \in \operatorname{argmax}_{a \in A} Q^*(s,a)$ est optimale.

Caractérisation des politiques optimales (2/3)

Equation (d'optimalité) de Bellman

V^* est solution du système :

$$V^*(s) = \max_{a \in A} r(s,a) + \gamma \sum_{s' \in S} p\left(s'|s,a\right) V^*\left(s'\right)$$

$$V^* = T^* V^*$$

De même :

$$Q^*(s,a) = r(s,a) + \gamma \sum_{s' \in S} p\left(s'|s,a\right) \max_{a' \in A} Q^*\left(s',a'\right)$$

$$Q^* = T^* Q^*$$

Caractérisation des politiques optimales (3/3)

- L'équation d'optimalité de Bellman est une équation de *Programmation Dynamique*.
- T^* est un opérateur contractant sur un espace de Banach, V^* (Q^*) est son unique point fixe.

Résoudre l'équation de Bellman ?

- La suite des $V_{n+1} = T^* V_n$ converge vers V^*
  $\rightarrow$ algorithme d'itération sur les valeurs.
- La suite des $\pi_{n+1} \in \underset{a}{\text{argmax}}\, Q^{\pi_n}$ converge vers π^*
  $\rightarrow$ algorithme d'itération sur les politiques.
- V^* est la plus petite fonction t.q. $V(s) \geq r(s,a) + \gamma \sum_{s' \in S} p(s'|s,a)\, V(s)$
  $\rightarrow$ résolution par prog. linéaire.

Et maintenant ?

C'est bien beau tout ça, mais on a dit qu'on ne connaissait pas p et r !

Alors comment fait-on pour calculer Q^π ou Q^* ou π^* ?

Et maintenant ?

C'est bien beau tout ça, mais on a dit qu'on ne connaissait pas p et r !

Alors comment fait-on pour calculer Q^π ou Q^* ou π^* ?

On apprend par interaction avec le système,
en recueillant des échantillons (s, a, r, s').

Evaluer une politique

Problème de *prédiction*

Trois familles de méthodes :

- Programmation dynamique adaptative / évaluation sur modèle / apprentissage par renforcement indirect.
- Evaluation de Monte-Carlo.
- Différences Temporelles.

On ze road again!

Estimation du temps de trajet vers la maison.

Heure	Distance	Description	ETA
18h00	0	Fin des cours	18h30
18h05	1	Démarrage de la voiture, il pleut	18h40
18h20	12	Sortie de l'autoroute plus tôt que prévu	18h35
18h30	16	Coincé derrière un camion	18h40
18h40	18	Dernière ligne droite	18h43
18h43	18.5	Arrivée à la maison	18h43

- Etat : distance parcourue.
- Actions : une seule action, avancer.
- Récompenses : durées entre deux positions.
- Valeur d'un état : Temps de trajet restant.

Méthode de Monte-Carlo

A la fin d'un épisode, m.à.j. des estimations des valeurs des états traversés :

$$V(s_t) \leftarrow V(s_t) + \alpha\,[R_t - V(s_t)]$$

Sur l'exemple précédent :

Distance	Heure	ETA	$V(s_t)$	R_t	$R_t - V(s_t)$
0	18h00	18h30	30	43	+13
1	18h05	18h40	35	38	+3
12	18h20	18h35	15	23	+8
16	18h30	18h40	10	13	+3
18	18h40	18h43	3	3	0
18.5	18h43	18h43	0	0	0

Cette méthode converge à la limite vers V^π, mais deux faiblesses :

- Nécessite interactions par épisodes de longueur finie. Pourtant à 18h05 on sait déjà que l'estimation de 18h00 était trop optimiste
- Un événement rare (le camion) affecte tous les états précédemment visités ce qui n'était peut-être pas nécessaire.

Différences Temporelles

A chaque échantillon (s_t, a_t, r_t, s_{t+1}) :

$$V(s_t) \leftarrow V(s_t) + \alpha\left[r_t + \gamma V(s_{t+1}) - V(s_t)\right]$$

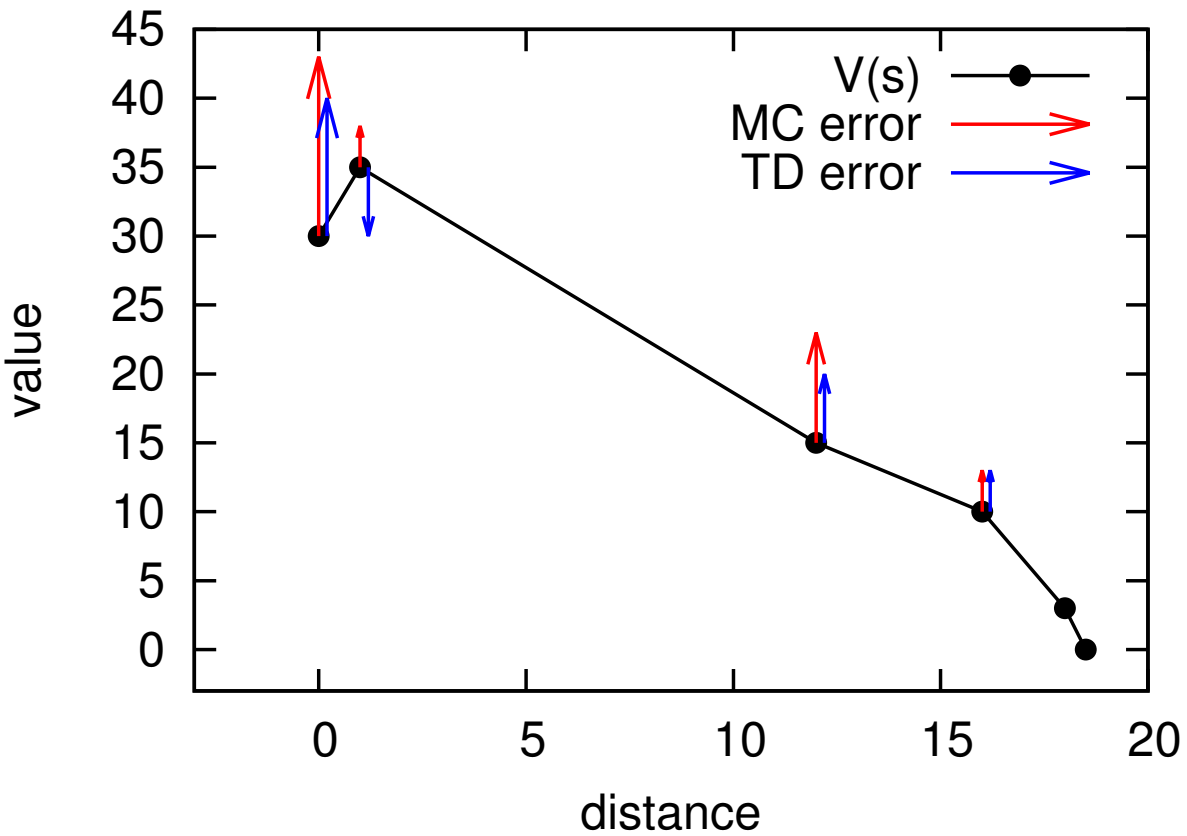
Sur l'exemple précédent :

Dist.	Heure	ETA	$V(s_t)$	R_t	$R_t - V(s_t)$	r_t	$r_t + \gamma V(s_{t+1}) - V(s_t)$
0	18h00	18h30	30	43	+13	5	+10
1	18h05	18h40	35	38	+3	15	-5
12	18h20	18h35	15	23	+8	10	+5
16	18h30	18h40	10	13	+3	10	+3
18	18h40	18h43	3	3	0	3	0
18.5	18h43	18h43	0	0	0	—	—

Convergence à la limite vers $V^{\pi}(s)$.

- $r_t + \gamma V(s_{t+1}) - V(s_t)$ = *différence temporelle* de prédiction
- Mise à jour en ligne par "bootstrap".
- Pas besoin d'épisodes.

On ze road again!



Conditions de Robbins-Monroe

Conditions de Robbins-Monroe

Les algorithmes d'estimation stochastique tels que MC ou TD convergent si les pas α_t de mise à jour respectent les conditions de Robbins-Monroe :

- $\sum_{t=0}^{\infty} \alpha_t = +\infty$
- $\sum_{t=0}^{\infty} \alpha_t^2 < +\infty$

MC ou TD ?

- MC est plus sensible que TD aux événements rares ou aux actions exploratoires.
- MC converge plus vite sur les problèmes quasi-déterministes.

En pratique, souvent TD converge plus vite, mais il n'existe pas de preuve (et il y a même des contre-exemples).

Pour aller plus loin

- Méthode qui combine TD et MC ? $\rightarrow$ TD(λ)
- Malédiction de la dimension (curse of dimensionality)
- Version batch de ces algorithmes
- Possibilité de “reset” d’une simulation ?
- Nécessité d’appliquer la politique qu’on souhaite évaluer ?

Apprendre un contrôleur optimal

Deux cadres principaux

- En ligne / interactif
- Hors ligne / non-interactif

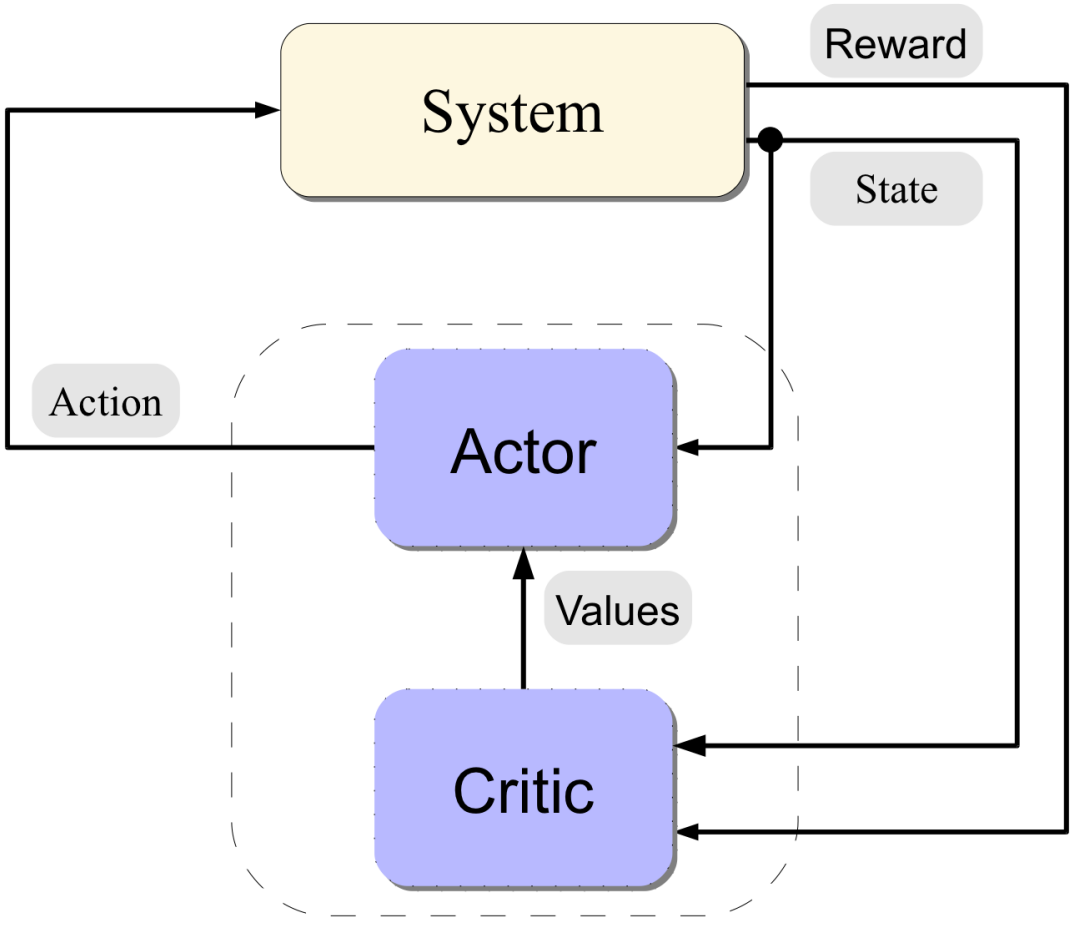
Architecture Actor-Critic

En ligne, deux processus interagissent naturellement :

- Evaluation d'une politique (courante ou cible)
- Choix des actions à entreprendre

Architecture Actor-Critic

Schéma général



Presque tous les algorithmes de RL entrent dans cette architecture.
On va en examiner deux fondamentaux.

SARSA — l'intuition

On dispose de l'algo TD:
essayons d'évaluer la politique courante $\rightarrow Q$,
. . . tout en laissant celle-ci être Q-gloutonne.

Que se passe-t-il alors ?
Convergence $Q \rightarrow Q^*$, $\pi \rightarrow \pi^*$

SARSA — calcul de la différence temporelle

Rappel TD(0): $\delta = r + \gamma V(s') - V(s)$

Or $V(s') = Q(s', \pi(s'))$

Evaluation de la politique π courante : $\delta = r + \gamma Q(s', a') - Q(s, a)$

SARSA — l'algorithme

En s, choisir (*actor*) a en utilisant Q, puis :

1. Observer r, s'
2. Choisir a' (*actor*) en utilisant Q
3. $\delta = r + \gamma Q(s', a') - Q(s, a)$
4. $Q(s, a) \leftarrow Q(s, a) + \alpha\delta$
5. $s \leftarrow s'$, $a \leftarrow a'$ et répéter

SARSA — convergence

Convergence de SARSA

Si, quand $t \to \infty$,

1. tous les (s, a) ont été explorés une infinité de fois,
2. l'actor converge vers une politique Q-gloutonne,
   Greedy in the limit of infinite exploration (GLIE)

alors l'actor converge vers π^* et Q vers Q^*.

Pour s'assurer de (1), exploration nécessaire !

Définir un acteur GLIE :

- ε-soft, ε-greedy: $\pi\left(a \neq \underset{a'}{\operatorname{argmax}}\, Q(s, a') | s\right) = \varepsilon$
- Politiques de Boltzmann $\pi(a|s) = \frac{e^{\frac{Q(s,a)}{\tau}}}{\sum_{a'} e^{\frac{Q(s,a')}{\tau}}}$

SARSA — on-policy critic

SARSA évalue en permanence la politique π courante...
...qui dérive doucement vers π^*

Quand le critic évalue la politique courante de l'actor,
l'on parle d'algorithme *on-policy*.

Exemple de méthode *off-policy* : Q-learning.

Q-learning — l'intuition

Le critic vise à évaluer Q^*,
indépendement des actions de l'actor.

Puis, l'actor devenant Q-glouton, il converge vers π^*.

Q-learning — calcul de la différence temporelle

Rappel TD(0): $\delta = r + \gamma V(s') - V(s)$

$$V^*(s') = Q^*(s', \pi(s')) = \max_{a'} Q^*(s', a')$$

Evaluation de la politique π courante : $\delta = r + \gamma \max_{a'} Q(s', a') - Q(s, a)$

Q-learning — l'algorithme

En s,

1. Choisir a (*actor*) en utilisant Q
2. Observer r, s'
3. $\delta = r + \gamma \max_{a'} Q(s', a') - Q(s, a)$
4. $Q(s, a) \leftarrow Q(s, a) + \alpha\delta$
5. $s \leftarrow s'$ et répéter

Q-learning — convergence

Convergence de Q-learning

Comme pour SARSA, si, à la limite,

1. tous les (s, a) ont été explorés une infinité de fois,
2. l'actor converge vers une politique Q-gloutonne,

alors l'actor converge vers π^*.

De nouveau, pour s'assurer de (1), exploration nécessaire !

Définir un acteur GLIE :

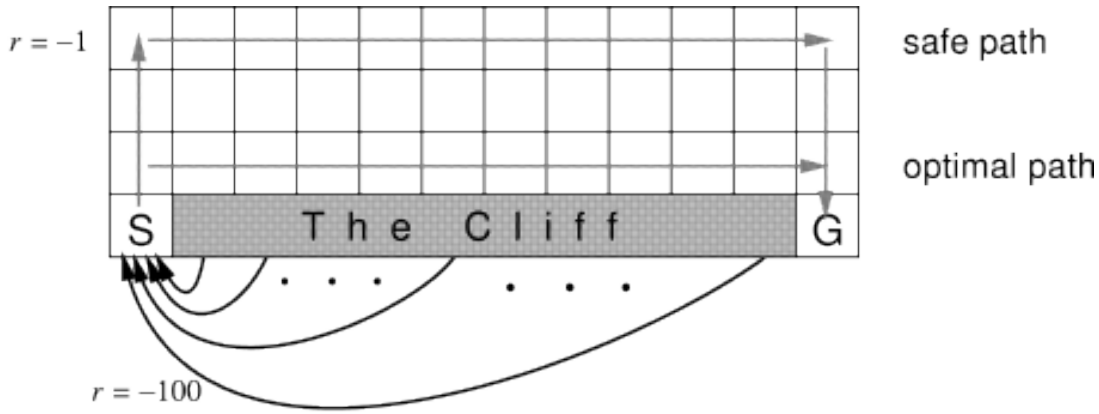
- ε-soft, ε-greedy: $\pi\left(a \neq \underset{a'}{\operatorname{argmax}}\, Q(s, a') | s\right) = \varepsilon$
- Politiques de Boltzmann $\pi(a|s) = \frac{e^{\frac{Q(s,a)}{\tau}}}{\sum_{a'} e^{\frac{Q(s,a')}{\tau}}}$

Q-learning — Off-policy critic

Q-learning évalue l'optimum Q^*
et non la politique appliquée π.

Il s'agit d'un algorithme *off-policy*.

Comparison amusante : la falaise



Etats positions sur la grille

Actions N, S, E, O

Transitions déterministes

Récompenses -100 si chute, -1 sur la grille, $+1$ au but.

- Quelle est la politique optimale selon vous ?
- Fixons $\varepsilon = 0.1$, quelle différence entre les choix de SARSA et de Q-learning ?
- Que se passe-t-il quand ε tend vers zéro ?

Problèmes hors ligne / non-interactifs

Autre famille de problèmes d'apprentissage :
Pas d'interaction avec l'environnement.
Données acquises a priori : $\mathscr{D} = \{(s_i, a_i, r_i, s'_i\}_{i \in [1,N]}$

Pas de compromis exploration / exploitation.

En revanche, un nouveau problème :
Echantillons dans seulement un sous-ensemble de $S \times A$.
Besoin d'*approcher* Q et π,
et de *généraliser* aux états/actions non-connus.

Résumons un peu tout cela

- Apprentissage par renforcement = pb de décision séquentielle qui maximise un retour sur le long terme, en exploitant des données issues de l'interaction instantanée avec l'environnement et non d'un modèle.

Résumons un peu tout cela

- Apprentissage par renforcement = pb de décision séquentielle qui maximise un retour sur le long terme, en exploitant des données issues de l'interaction instantanée avec l'environnement et non d'un modèle.
- Hypothèses: environnement Markov Decision Process.

Résumons un peu tout cela

- Apprentissage par renforcement = pb de décision séquentielle qui maximise un retour sur le long terme, en exploitant des données issues de l'interaction instantanée avec l'environnement et non d'un modèle.
- Hypothèses: environnement Markov Decision Process.
- Comportement = politique, $\pi(s) = a$.

Résumons un peu tout cela

- Apprentissage par renforcement = pb de décision séquentielle qui maximise un retour sur le long terme, en exploitant des données issues de l'interaction instantanée avec l'environnement et non d'un modèle.
- Hypothèses: environnement Markov Decision Process.
- Comportement = politique, $\pi(s) = a$.
- Quelques résultats fondamentaux : $Q^\pi = T^\pi Q^\pi$, $Q^* = T^* Q^*$.

Résumons un peu tout cela

- Apprentissage par renforcement = pb de décision séquentielle qui maximise un retour sur le long terme, en exploitant des données issues de l'interaction instantanée avec l'environnement et non d'un modèle.
- Hypothèses: environnement Markov Decision Process.
- Comportement = politique, $\pi(s) = a$.
- Quelques résultats fondamentaux : $Q^\pi = T^\pi Q^\pi$, $Q^* = T^* Q^*$.
- Deux problèmes principaux : évaluation et contrôle.

Résumons un peu tout cela

- Apprentissage par renforcement = pb de décision séquentielle qui maximise un retour sur le long terme, en exploitant des données issues de l'interaction instantanée avec l'environnement et non d'un modèle.
- Hypothèses: environnement Markov Decision Process.
- Comportement = politique, $\pi(s) = a$.
- Quelques résultats fondamentaux : $Q^\pi = T^\pi Q^\pi$, $Q^* = T^* Q^*$.
- Deux problèmes principaux : évaluation et contrôle.
- Evaluation :
  - Indirecte.
  - Monte-Carlo.
  - Différences temporelles.

Résumons un peu tout cela

- Apprentissage par renforcement = pb de décision séquentielle qui maximise un retour sur le long terme, en exploitant des données issues de l'interaction instantanée avec l'environnement et non d'un modèle.
- Hypothèses: environnement Markov Decision Process.
- Comportement = politique, $\pi(s) = a$.
- Quelques résultats fondamentaux : $Q^\pi = T^\pi Q^\pi$, $Q^* = T^* Q^*$.
- Deux problèmes principaux : évaluation et contrôle.
- Evaluation :
  - Indirecte.
  - Monte-Carlo.
  - Différences temporelles.
- Contrôle :
  - En ligne, on-policy : SARSA.
  - En ligne, off-policy : Q-learning.
  - Hors ligne.

Axes de lecture de la litterature RL

On peut classer les algorithmes et les problèmes de RL comme :

- Model-based vs. Model-free
- On-policy vs. Off-policy
- Online vs. Episodic vs. Offline
- Incremental vs. Batch

Défis ouverts

- Espaces S et A de grande taille, continus ou hybrides.
- Compromis optimal exploration vs. exploitation.
- Bornes de convergence à nombre d'échantillons finis.
- Applications concrètes en automatique, finance, jeux, systèmes industriels etc.

Et une littérature très riche !

Conseils de lecture