UE Apprentissage par renforcement - séance 1

Valentin Emiya

M2 IAAA

13 novembre 2019

Sommaire

1. Organisation de l'UE
2. L'apprentissage par renforcement : introduction générale
3. Bandits
   - Le problème
   - Stratégie aléatoire
   - Stratégie greedy
   - Stratégie ϵ-greedy
   - Stratégie Upper Confidence Bound (UCB)
   - Stratégie Thomson sampling
4. Bandits contextuels

Sommaire

1. Organisation de l'UE
2. L'apprentissage par renforcement : introduction générale
3. Bandits
   - Le problème
   - Stratégie aléatoire
   - Stratégie greedy
   - Stratégie ϵ-greedy
   - Stratégie Upper Confidence Bound (UCB)
   - Stratégie Thomson sampling
4. Bandits contextuels

Programme

En 7 séances de 4h :

1. Bandits (1/2) : notions et strategies de base, UCB
2. Bandits (2/2) : Thomson sampling, bandits contextuels
3. Monte-Carlo Tree Search
4. Processus de décision de Markov
5. TD learning
6. Miniprojet
7. Miniprojet

Communication

Enseignants :
Valentin Emiya (VE)
prenom.nom@lis-lab.fr

Ametice :
`https://ametice.univ-amu.fr/course/view.php?id=47391`
(tout le monde y a accès ?)

MCC

$$\frac{CC + ET}{2}$$

CC=TP et mini-projet

TP : installation Python

Version de Python conseillée : 3.7 (surtout pas Python 2 !)

Vous avez le choix entre :

- travailler en ligne sur colab
- utiliser un notebook avec une installation locale
- utiliser des fichiers `.py` avec une installation locale

...du moment que c'est une solution qui fonctionne sans perdre de temps d'installation en séance !

Donc : si une solution ne fonctionne pas, basculer sur une autre.

Références pour le cours

Richard S. Sutton et Andrew G. Barto, *Reinforcement Learning : An Introduction*, Second Edition, MIT Press, Cambridge, MA, 2018. Disponible en ligne [1].

Tom Mitchell, *Machine Learning*, 1997, chap. 13.

1. http://www.incompleteideas.net/book/RLbook2018.pdf

Sommaire

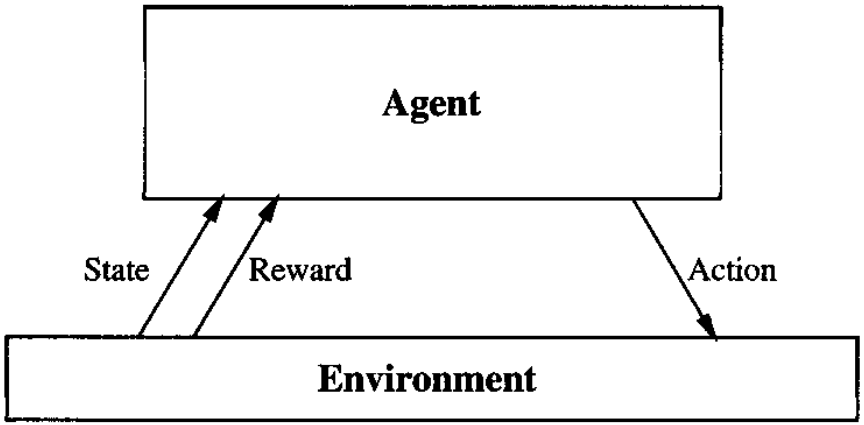
1. Organisation de l'UE
2. L'apprentissage par renforcement : introduction générale
3. Bandits
   - Le problème
   - Stratégie aléatoire
   - Stratégie greedy
   - Stratégie ϵ-greedy
   - Stratégie Upper Confidence Bound (UCB)
   - Stratégie Thomson sampling
4. Bandits contextuels

Préparation de l'UE

Quels retours sur la lecture du document envoyé ?

- temps passé
- impressions
- compréhension
- difficultés

Exemples introductifs



$$s_0 \xrightarrow[r_0]{a_0} s_1 \xrightarrow[r_1]{a_1} s_2 \xrightarrow[r_2]{a_2} \dots$$

Goal: Learn to choose actions that maximize

$$r_0 + \gamma r_1 + \gamma^2 r_2 + \dots \text{ , where } 0 \leqslant \gamma < 1$$

Quelles applications connaissez-vous ?
Quelles sont les observations ?
Quelles sont les actions ?
Quelles sont les récompenses ?

Formalisation du contexte : l'agent dans son environnement

Étant donnés

- un ensemble $\mathcal{S}$ d'*états*,
- un ensemble $\mathcal{A}$ d'*actions*,

un *agent* évolue en interaction dans un *environnement* :

- l'agent observe l'état courant $s \in \mathcal{S}$ et choisit une action $a = \pi(s)$ selon sa *stratégie* (appelée aussi *politique*, *policy*)

$$\pi : \mathcal{S} \to \mathcal{A}$$

- l'environnement entre dans un nouvel état $s' = \delta(s, a)$ selon une fonction

$$\delta : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$$

  éventuellement aléatoire et inconnue de l'agent

- l'agent reçoit une *récompense immédiate* $r(s, a)$ selon une fonction

$$r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$$

  éventuellement aléatoire et inconnue de l'agent.

$\to$ Comment l'agent peut-il ainsi apprendre une stratégie π qui maximise sa *récompense à long terme* ?

Remarques (1/3) : les données d'entrainement

Pour apprendre sa stratégie $\pi : \mathcal{S} \rightarrow \mathcal{A}$,

- l'agent n'a pas d'exemples (s_i, a_i) d'états s_i étiquetés par des actions optimales a_i.

$\neq$ apprentissage supervisé

- l'agent obtient des exemples (s_t, a_t, r_t, s_{t+1}) de récompense r_t et de nouvel état s_{t+1} lorsqu'il choisit une action a_t à partir d'un état s_t.

Les récompenses immédiates ne reflètent pas directement la récompense à long terme.

Remarques (2/3) : exploitation vs. exploration

L'agent doit faire une compromis entre

- *exploitation* : choisir une action qui parait optimale afin de maximiser ses récompenses à court terme.
- *exploration* : choisir une action qui parait sous-optimale afin d'en apprendre davantage sur l'environnement et sur les possibilités d'améliorer ses récompenses.

Remarques (3/3) : interactions

Il y a un aspect *online learning* : les données d'apprentissage arrivent progressivement lors de l'entrainement

Il y a un aspect *active learning* : l'agent a une influence sur les données à venir.

$\rightarrow$ mise en œuvre délicate !

Deux casquettes à utiliser

Que ce soit dans les analyses (cours/TD) ou dans les expérimentations (TP), vous serez amenez à changer de casquette entre deux rôles différents

Le modélisateur qui contrôle et a accès à toute l'information sur l'environnement

Le développeur de solution, qui n'a pas d'accès à l'ensemble de l'environnement (δ et r inconnus).

$\rightarrow$ Ne vous contentez pas de chercher des solutions : étudiez les problèmes
$\rightarrow$ Faites preuve de souplesse, adoptez la bonne perspective !

Aperçu de l'UE

- Bandits (séance 1 et 2)

  Plusieurs actions, un seul état
  Exploration vs. exploitation

- MCTS (séance 3)

  Résoudre des jeux à facteur de branchement élevé et fx d'éval. inconnue

- MDP et TD-learning (séance 4 et 5)

  Le cœur de l'AR

- Mini-projet (séances 6 et 7)

  Un peu d'expérience

Sommaire

1 Organisation de l'UE

2 L'apprentissage par renforcement : introduction générale

3 Bandits
- Le problème
- Stratégie aléatoire
- Stratégie greedy
- Stratégie ϵ-greedy
- Stratégie Upper Confidence Bound (UCB)
- Stratégie Thomson sampling

4 Bandits contextuels

Le problème

- Tâche : choix récurrent entre k actions
- Récompense aléatoire, selon une distribution inconnue dépendant de l'action
- Objectif : maximiser le gain moyen sur une certaine durée

Scenario 1 : la question du matin

Suite à votre brillante réussite en Master IAAA vous avez décroché un CDD dans une nouvelle ville.
Pour aller travailler, vous avez le choix entre trois itinéraires depuis votre nouveau logement : par l'autoroute en face, la nationale à droite ou les petites routes à gauche.
Chaque matin, la même question vous tracasse : quel itinéraire prendre aujourd'hui. En arrivant au travail, vous regardez votre montre et éprouvez plus ou moins de satisfaction en constatant votre temps de trajet.

Comment choisir chaque matin un itinéraire de façon à optimiser votre temps de trajet pendant votre CDD ?

NB : une autre question se pose aussi un peu plus tôt, au réveil : dois-je me lever du pied gauche ou du pied droit pour passer une bonne journée ?

Scenario 2 : publicité ciblée

Votre site internet a un unique encart publicitaire. Un annonceur vous propose de diffuser des publicités de voitures, de matériel informatique et de films, en vous offrant 1 euro par clic sur une publicité. À chaque visite, vous pouvez choisir d'afficher une publicité parmi les trois thèmes proposés. Vous pensez que vos visiteurs sont plus intéressés par un des thèmes mais vous ignorez lequel.

Comment faire ce choix à chaque visite pour devenir aussi riche que possible ?

Scenario 3 : traitement médical

Une épidémie frappe le continent : le virus est inconnu et les médecins hésitent entre dix traitements possibles. Vous êtes en charge de l'assignation des traitements : quand un malade se présente, vous devez choisir un traitement parmi les dix, puis vous apprenez assez vite s'il a survécu ou pas (la maladie est foudroyante).

Comment faire ce choix à chaque nouveau malade afin de sauver un maximum de vies ?

Scenario 4 : premier coup dans un jeu

C'est toujours pareil quand vous jouez aux dames, aux échecs, au go, etc. : à chaque fois en début de partie, vous avez le sentiment que le premier coup a un impact décisif sur l'issue de la partie. Vous aimeriez savoir quoi jouer en premier pour mettre un maximum de chance de votre côté. Ce weekend, vous aurez l'occasion de jouer beaucoup de parties et vous espérez en remporter un maximum.

Comment choisir le premier coup à chaque nouveau début de partie pour maximiser vos victoires ?

Modélisation du problème des bandits à k bras

Ensemble des actions possibles : $\mathcal{A} = \{0, \ldots, k-1\}$.

Pour $a \in \mathcal{A}$, on appelle *valeur du bras a* la récompense moyenne $q^*(a)$ obtenue lorsque l'on joue le bras a. Autrement dit, $q^*(a) = \mathbb{E}[R|a]$, où la variable aléatoire $R|a$ est la récompense obtenue en jouant le bras a.

Exemple : modèle de Bernoulli

On suppose que la récompense de chaque bras a est générée selon une loi de Bernoulli de paramètre $p_a \in [0, 1]$: lorsque l'on tire le bras a, la récompense est de 1 avec une probabilité p_a et de 0 avec une probabilité $1 - p_a$. On a $q^*(a) = p_a$.
Voir code de la classe `BernoulliMultiArmedBandits`.

Autre exemple : modèle gaussien

La récompense de chaque bras a est générée selon une loi $\mathcal{N}(\mu_a, \sigma = 1)$ de moyenne μ_a inconnue par l'agent. On a $q^*(a) = \mu_a$.
Voir code de la classe `NormalMultiArmedBandits`.

Démo notebook.

L'agent

L'agent de ne connait pas les valeurs $q^*(a)$ des bras.

À chaque itération t, il choisit de jouer un bras a_t puis reçoit une récompense r_t lui permettant d'affiner sa connaissance sur le bras joué.

Son objectif est de maximiser ses gains cumulés $\sum_{t=0}^{T-1} r_t$.

Pour choisir une action à l'itération t, il dispose de l'information des $t-1$ actions choisies précédemment et des $t-1$ récompenses associées. Il doit faire un compromis entre choisir l'action qui lui parait la plus rentable (exploitation) et choisir d'autres actions pour affiner ses estimations de la valeur des bras (exploration).

Deux fonctions utiles pour les algorithmes de bandits

Pour chaque bras, on définit :

- $N(a)$: le nombre de fois que le bras a a été joué par l'agent ;
- $R(a)$: la somme des récompenses obtenues par l'agent en jouant le bras a ;
- $Q(a) = \frac{R(a)}{N(a)}$: l'estimation de la valeur du bras a (récompense moyenne).

Mise à jour de $Q(a)$: inutile de stocker $R(a)$, on peut mettre à jour $Q(a)$ à partir de sa valeur précédente et de $N(a)$ seulement. Si on note $\mu_n = \frac{1}{n}\sum_{i=1}^{n} x_i$ la moyenne de n valeurs x_i, avec $\mu_0 = 0$, montrez que pour tout $n > 1$, on a $\mu_n = \mu_{n-1} + \frac{1}{n}(x_n - \mu_{n-1})$.

Notion de regret

Si l'agent connaissait la valeur des bras, la stratégie optimale serait de choisir toujours $a^* = \text{argmax}_{a \in \mathcal{A}} q^*(a)$. Le gain moyen au bout de n itérations serait de $nq^*(a^*)$

Le regret mesure la performance d'une stratégie sur n itérations par rapport à la stratégie optimale. Il est défini par

$$n \times q^*(a^*) - \sum_{t=0}^{n-1} r_t$$

où r_t est la récompense obtenue à l'itération t par la stratégie utilisée. C'est une quantité utilisée notamment pour l'analyse théorique des performances des algorithmes.

Déroulement classique d'un algorithme

Pour $t \in \{0, \ldots, n-1\}$,

- l'agent choisit de jouer un bras A_t
- l'environnement fournit à l'agent une récompense R_t dépendant du bras A_t
- l'agent met à jour sa stratégie en fonction de (A_t, R_t)

Stratégie aléatoire

Idée : peu importe ce qui s'est passé précédemment, choisir une action aléatoirement, uniformément parmi toutes les actions possibles.

Algorithme : pour $t \in \{0, \ldots, n-1\}$,

- l'agent choisit de jouer un bras A^{random} uniformément au hasard
- obtention de la récompense R_t

Limitation : exploration pure, on n'apprend rien, on n'exploite pas les observations accumulées.

Stratégie greedy

Idée : choisir toujours l'action qui parait la plus rentable, au sens du gain moyen observé $Q(a)$.

Algorithme :

- Initialisation : pour chaque bras a
  - jouer a
  - obtenir la récompense r
  - initialiser $Q(a) \leftarrow r_a$ et $N(a) \leftarrow 1$
- Pour $t \in \{0, \ldots, n-1\}$,
  - choix du bras $A^{\text{greedy}} = \text{argmax}_a\, Q(a)$
  - obtention de la récompense R_t
  - mises à jour du nombre de tirage et de la moyenne du bras

$$N\left(A^{\text{greedy}}\right) \leftarrow N\left(A^{\text{greedy}}\right) + 1$$

$$Q\left(A^{\text{greedy}}\right) \leftarrow Q\left(A^{\text{greedy}}\right) + \frac{1}{N\left(A^{\text{greedy}}\right)}\left(R_t - Q\left(A^{\text{greedy}}\right)\right)$$

Limitation : exploitation pure, on peut rester bloqué sur un bras sous-optimal.

Stratégie ϵ-greedy

Idée : exploration/exploitation en mélangeant les stratégies aléatoire et greedy.

Algorithme : étant donné $\epsilon \in [0, 1]$ fixé, répéter pour $t \in \{0, \ldots, n-1\}$,

- choisir le bras $A_t = \begin{cases} A^{\text{greedy}} \text{ avec probabilité } 1 - \epsilon \\ A^{\text{random}} \text{ avec probabilité } \epsilon \end{cases}$
- obtention de la récompense R_t
- mettre à jour du nombre de tirages $N(A_t)$ et de la moyenne du bras $Q(A_t)$.

$\epsilon \to 0$: favorise l'exploitation.
$\epsilon \to 1$: favorise l'exploration.

La moyenne empirique de chaque bras tend vers la vraie valeur du bras quand T augmente mais on continue à explorer des bras sous-optimaux trop souvent.

Inégalité de concentration (Hoeffding)

Pour n variables aléatoires bornées $X_1, \ldots, X_n \in [a, b]$ indépendantes, de même espérance $\mathbb{E}[X_i] = \mu$, on a, pour tout $\delta > 0$,

$$\mathbb{P}\left[\mu < \frac{1}{n}\sum_{i=1}^{n} X_i + (b-a)\sqrt{\frac{\ln \frac{1}{\delta}}{2n}}\right] \geq 1 - \delta$$

Upper Confidence Bound (UCB)

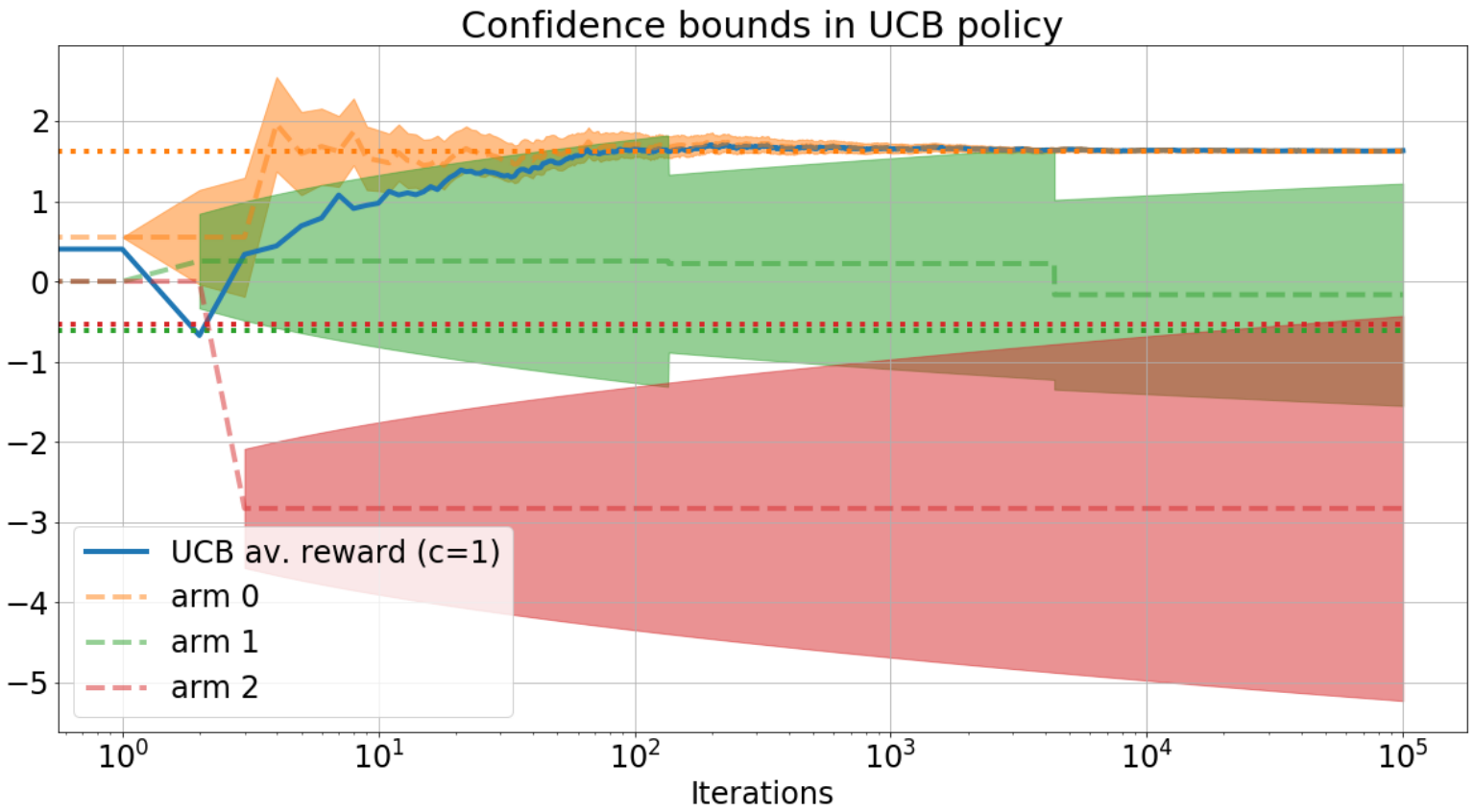
Principe : « Optimisme face à l'incertitude »

Choix du bras : on choisit le bras dont la borne supérieure est maximale

$$A_t^{\text{UCB}} = \underset{a}{\text{argmax}}\, Q(a) + c\sqrt{\frac{\ln t}{N(a)}}$$

avec $c > 0$. Le choix de c permet d'ajuster le compromis exploration/exploitation.

Comportement de la stratégie UCB



Thomson sampling

Voir séance 2.

Sommaire

1. Organisation de l'UE
2. L'apprentissage par renforcement : introduction générale
3. Bandits
   - Le problème
   - Stratégie aléatoire
   - Stratégie greedy
   - Stratégie ϵ-greedy
   - Stratégie Upper Confidence Bound (UCB)
   - Stratégie Thomson sampling
4. Bandits contextuels

Bandits contextuels

Voir séance 2.