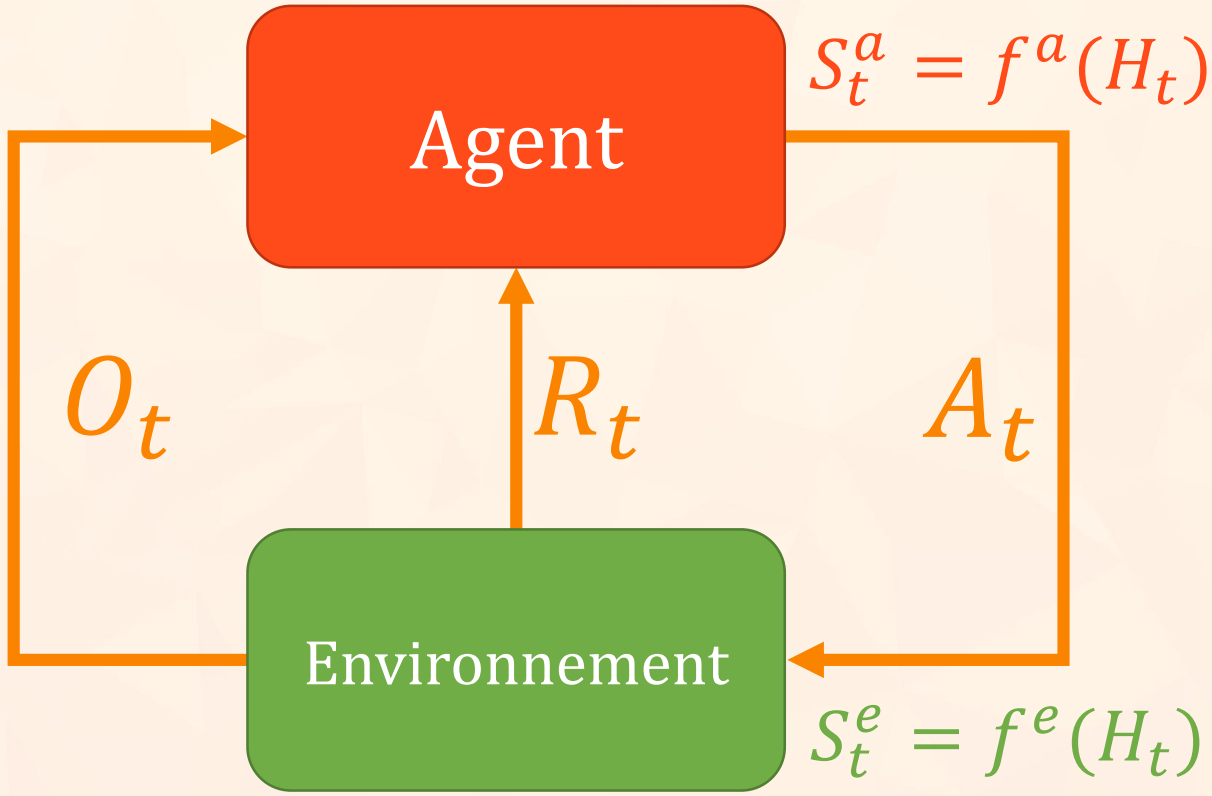
Apprentissage par Renforcement

Basé sur le cours de David Silver :

https://www.youtube.com/playlist?list=PLqYmG7hTraZDM-OYHWgPebj2MfCFzFObQ

Par Mathïs Fédérico 11/2019

Posons les bases



$$H_t = (O_0, A_0, R_0, O_1, A_1, R_1, \ldots, O_t, A_t, R_t)$$

Etat de Markov:

$$\mathbb{P}(S_{t+1} | S_t) = \mathbb{P}(S_{t+1} | S_t, S_{t-1}, \ldots, S_0)$$

Parfaitement observable :

$$S_t = S_t^a = S_t^e = O_t$$

Modèle:

$$\mathcal{P}_{ss'}^a = \mathbb{P}(S_{t+1} = s' | S_t = s, A_t = a)$$
$$\mathcal{R}_s^a = \mathbb{E}(R_{t+1} | S_t = s, A_t = a)$$

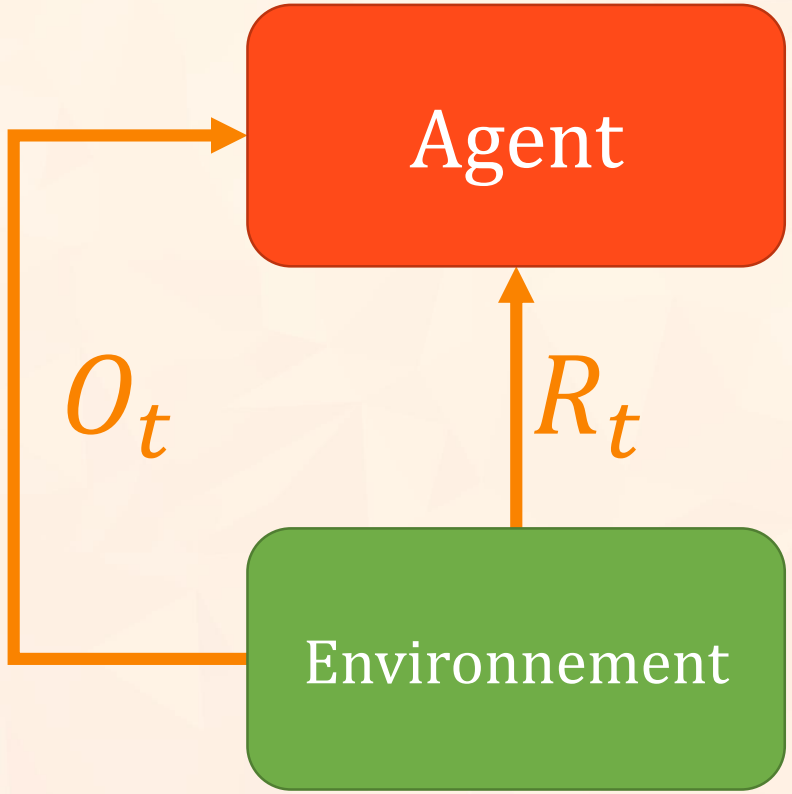
Connu → Planification

Inconnu → Apprentissage par renforcement

Planification

Avec un modèle

Planification - MRPs



L' Agent subit son environnement !

Mais si il choisit son état de départ, quel serait le meilleur état ?

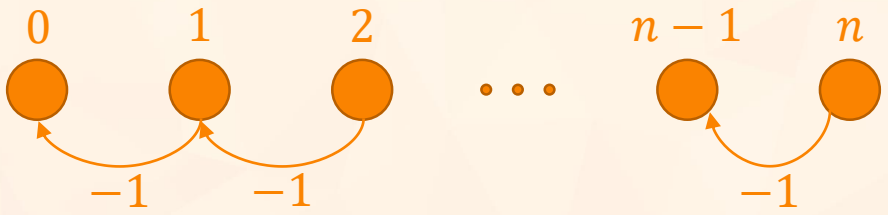
Gain futur $$G_t = \sum_{\tau=t+1}^{+\infty} \gamma^{\tau-(t+1)} R_\tau$$

Fonction de valeur d'état:

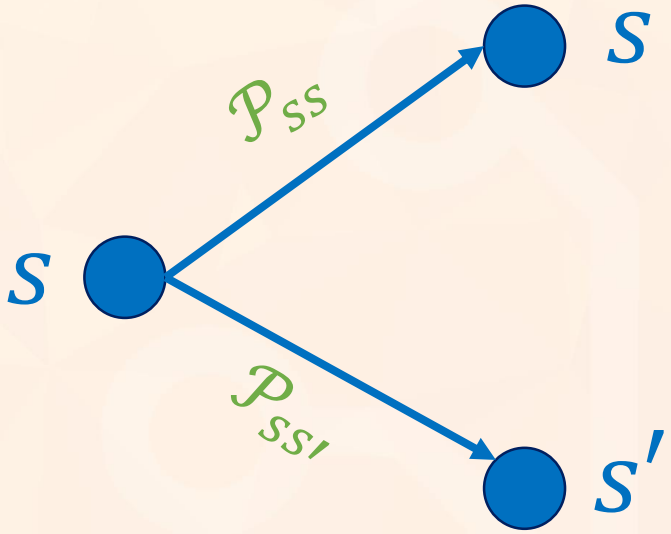
$$\forall s \in \mathcal{S}, v(s) = \mathbb{E}(G_t | S_t = s)$$

Planification – MRPs – Ford-Bellman

Un « jeu » très simple



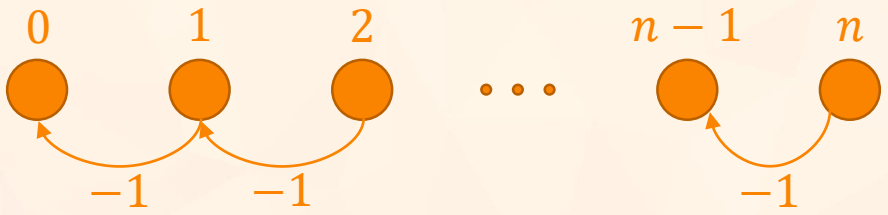
$$\forall s \in S, v(s) = -s$$



$$v(s) = R_{t+1} + \gamma \sum_{s \in S} \mathcal{P}_{ss'} \, v(s')$$

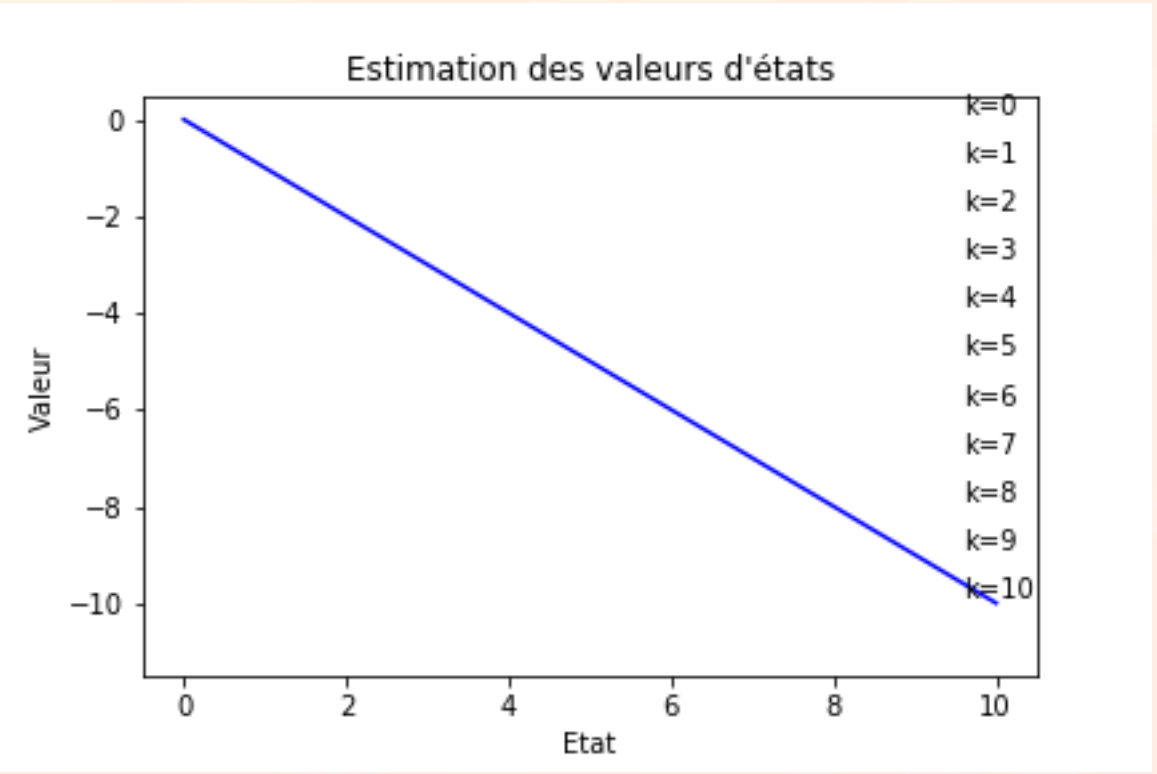
Planification – MRPs – Implémentation

Un « jeu » très simple

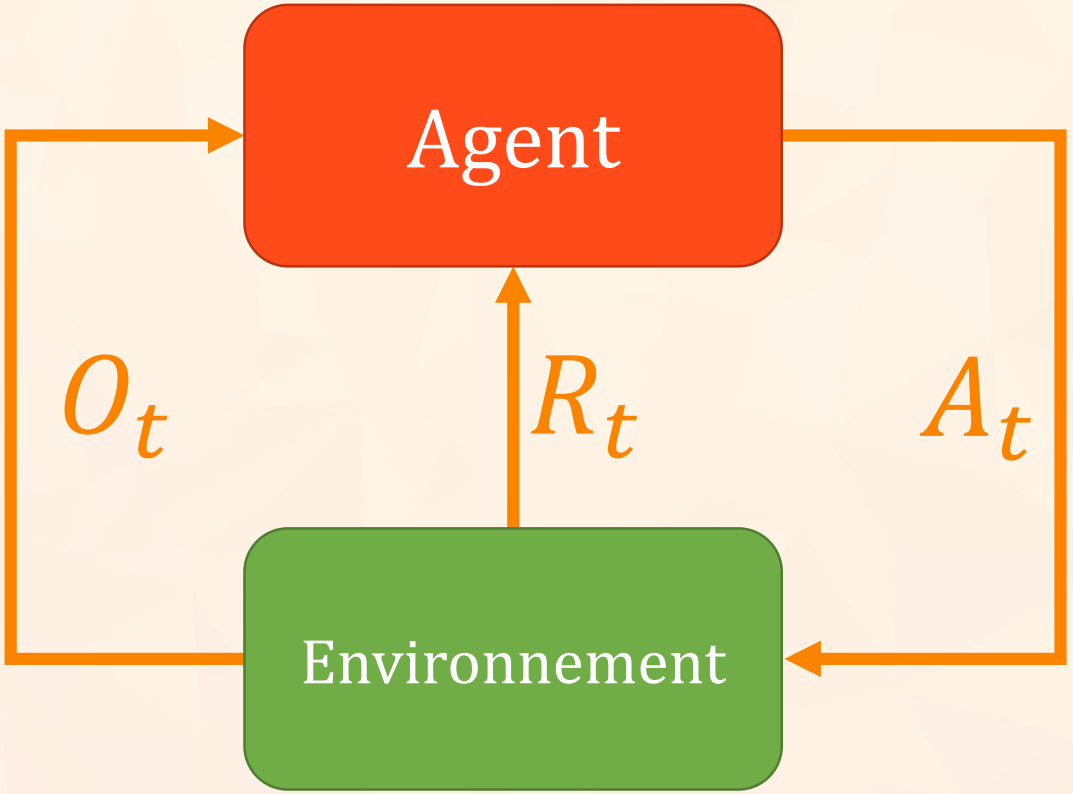


$$\forall s \in S, v(s) = -s$$

$$V_{k+1} = R + PV_k$$



Planification - MDPs



L' Agent prend maintenant des décisions !

On définit pour cela la *politique* :

$$\pi(a|s) = \mathbb{P}(A_t = a \mid S_t = s)$$

Fonction de valeur d'états:

$$\forall s \in \mathcal{S},$$

$$v_\pi(s) = \mathbb{E}_\pi(G_t|S_t = s)$$

Fonction de valeur d'actions:

$$\forall s, a \in (\mathcal{S}, \mathcal{A}),$$

$$q_\pi(s,a) = \mathbb{E}_\pi(G_t \,|S_t = s, A_t = a)$$

Le jeu de Nim Trivial

$\mathcal{S}$

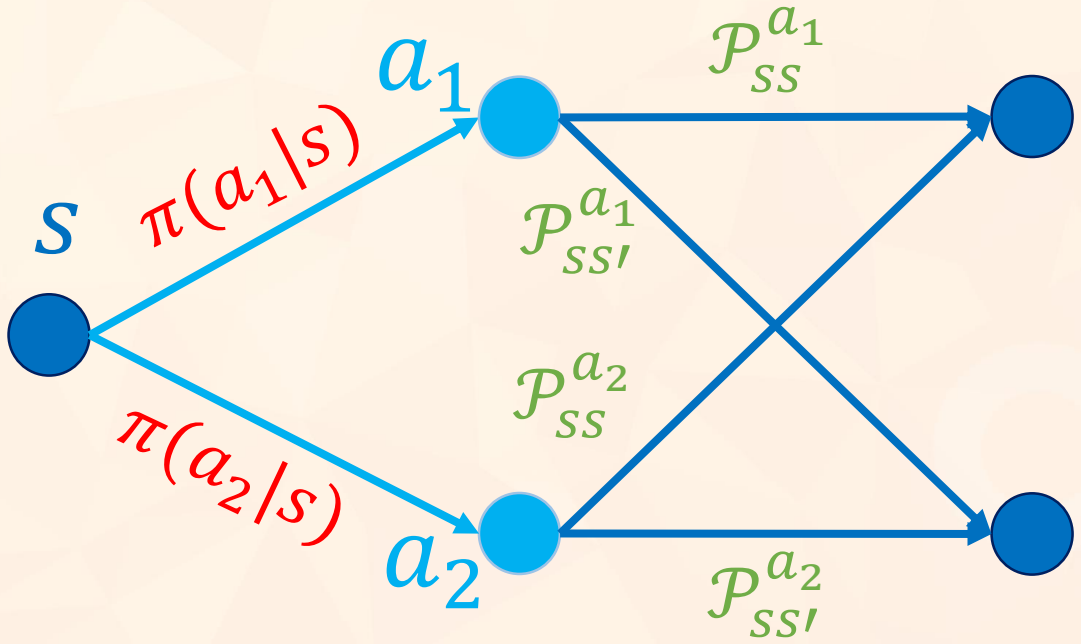
$\mathcal{A}$

1 2 3

Joueur aléatoire

$\mathcal{P}$ $\mathcal{R}$

Planification – MDPs – Ford-Bellman Dynamique



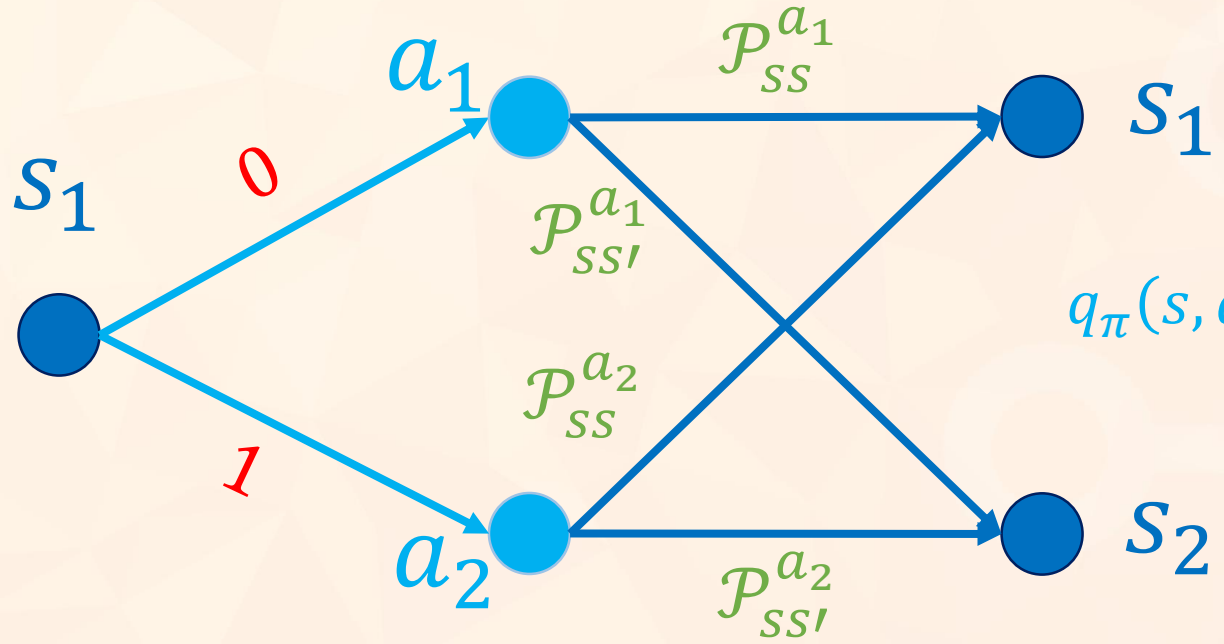
$$v_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s)\, q_\pi(s,a)$$

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

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

$$q_\pi(s,a) = R_s^a + \sum_{s \in \mathcal{S}} \gamma \mathcal{P}_{ss'}^a \left(\sum_{a \in \mathcal{A}} \pi(a|s)\, q_\pi(s,a)\right)$$

Planification – MDPs – Ford-Bellman Optimale



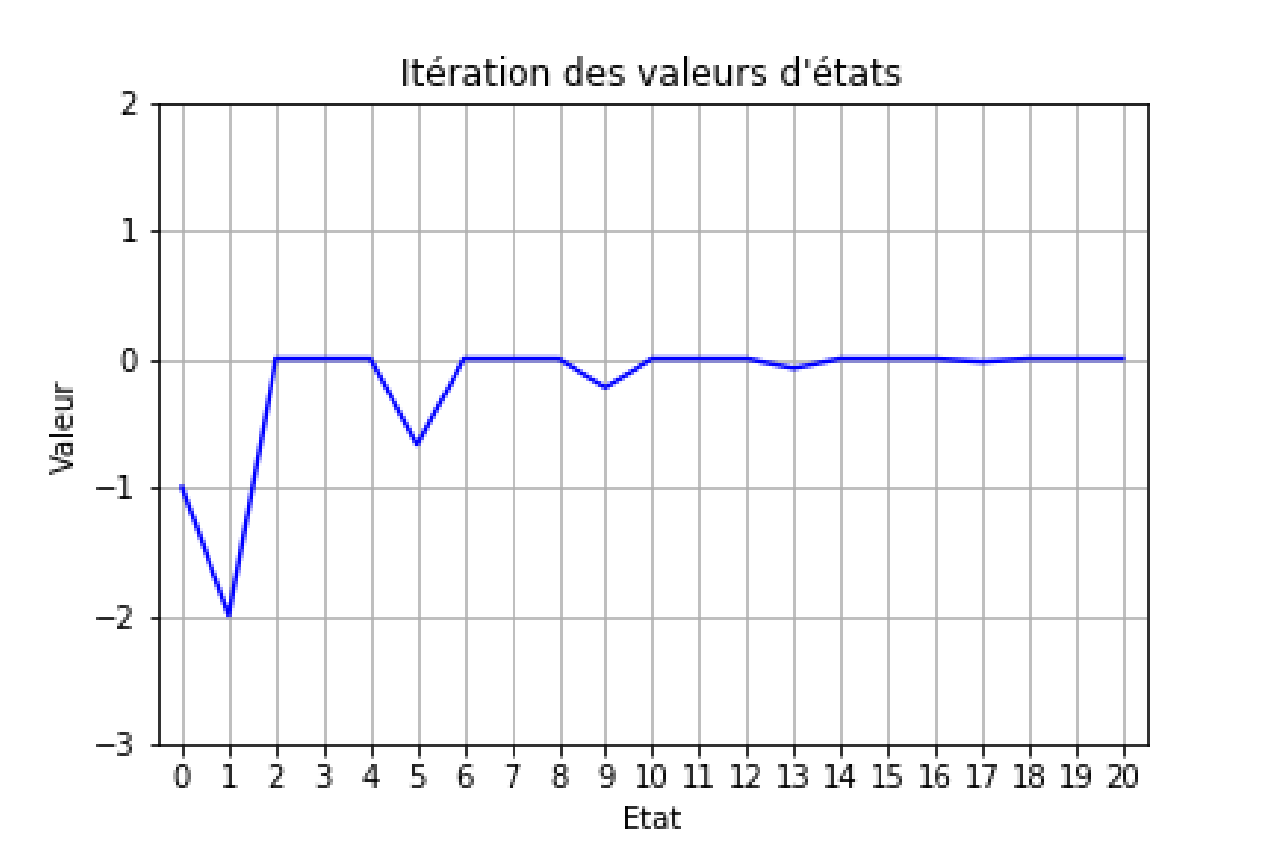
$$v_\pi(s) = \max_{a \in \mathcal{A}} q_\pi(s,a)$$

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

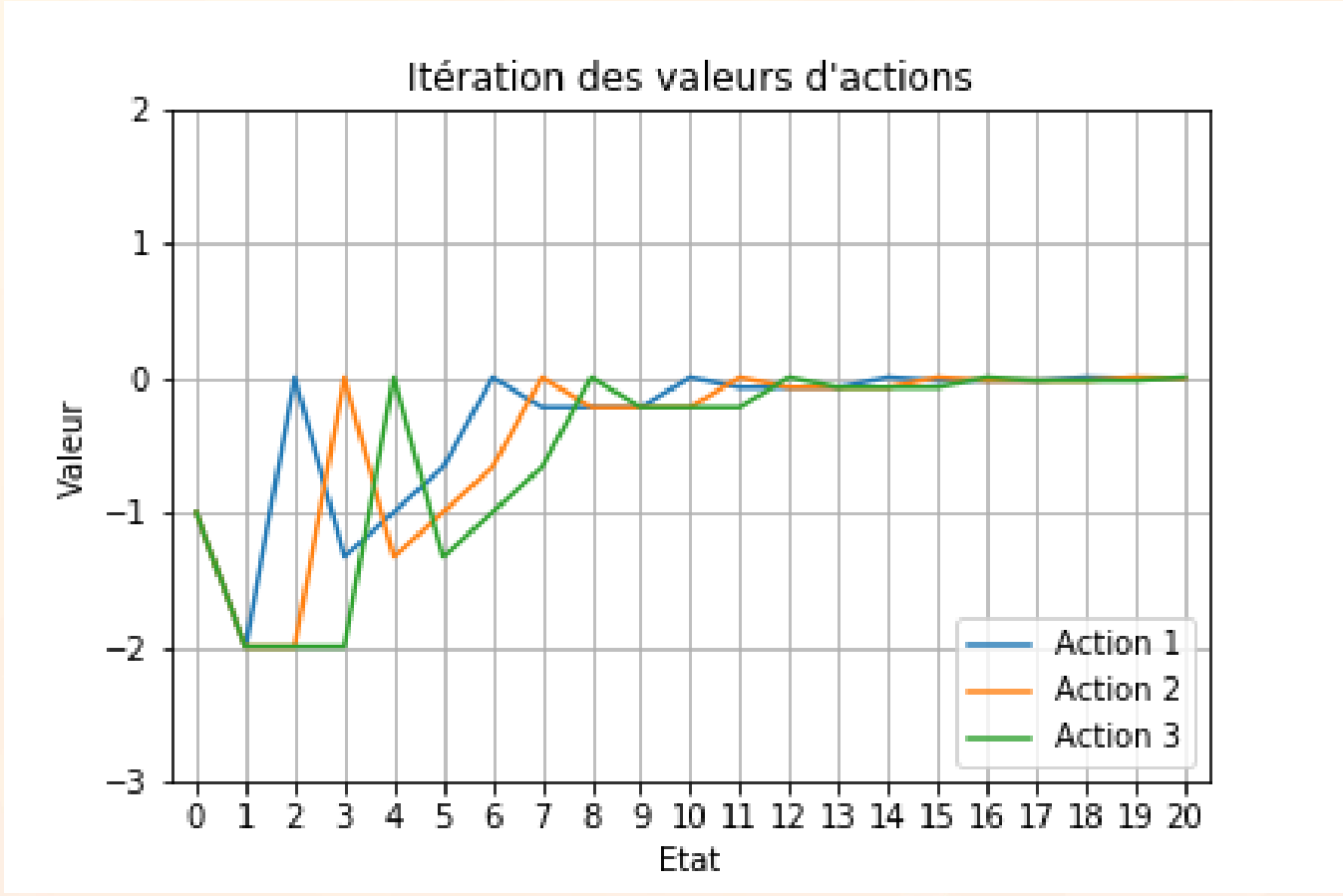
$$v_\pi(s) = \max_{a \in \mathcal{A}} R_s^a + \sum_{s' \in \mathcal{S}} \gamma \mathcal{P}_{ss'}^a \, v_\pi(s')$$

$$q_\pi(s,a) = R_s^a + \sum_{s \in \mathcal{S}} \gamma \mathcal{P}_{ss'}^a \max_{a \in \mathcal{A}} q_\pi(s,a)$$

Planification – MDPs – Implémentation



Planification – MDPs – Implémentation




```
We can deduce the optimal policy:
[[ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20]
 [ 1  1  1  2  3  1  1  2  3  3  1  2  3  3  1  2  3  3  1  2  3]]
```

Apprentissage par renforcement

Evaluation sans modèle

RL – Evaluation – Monte-Carlo

Méthode des moments :

$$v_\pi(s) = \mathbb{E}_\pi(G_t \,|S_t = s) \leftarrow \widehat{\mathbb{E}}_\pi(G_t \,|S_t = s)$$

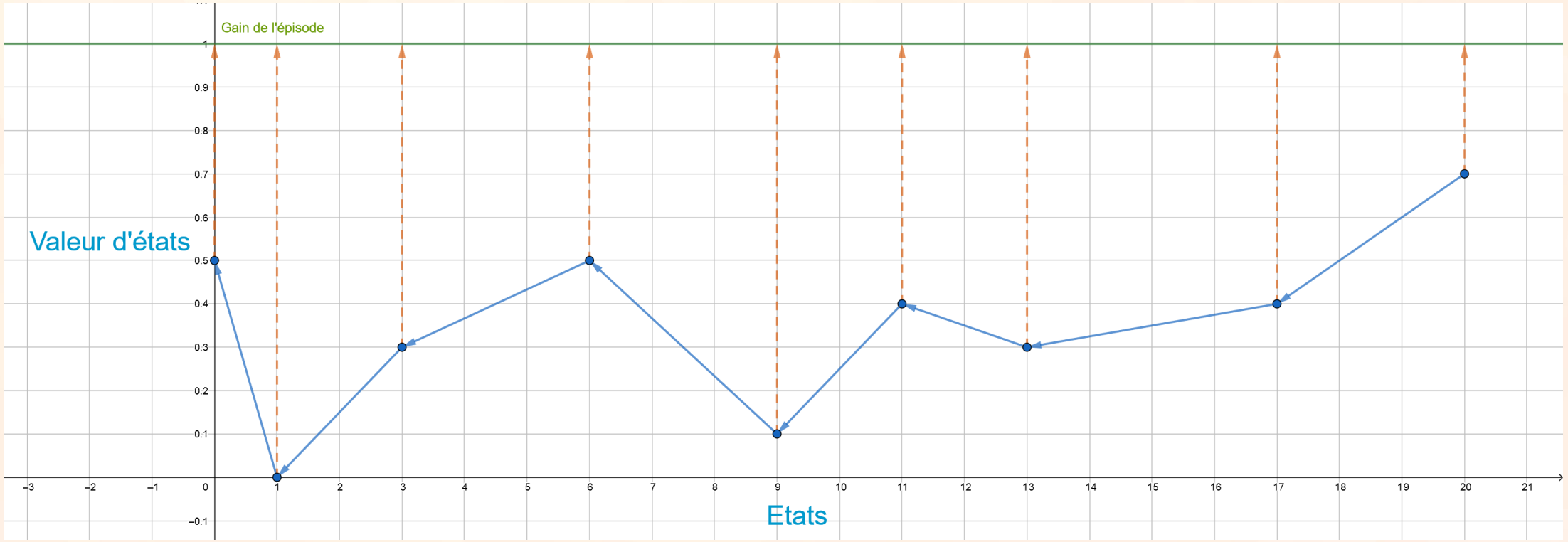
$$\widehat{\mathbb{E}}_\pi(G_t \,|S_t = s) = \frac{Somme\ des\ gains\ où\ s\ est\ observé}{Nombre\ de\ parties\ où\ s\ est\ observé} = \frac{\sum G(s)}{N(s)}$$

A la fin de chaque épisode :

$$\forall s \in \mathcal{S}, \qquad v_\pi(s) \leftarrow \frac{G}{N(s)} + \frac{N(s)-1}{N(s)} v_\pi(s)$$

$$\forall s \in \mathcal{S}, \qquad v_\pi(s) \leftarrow v_\pi(s) + \frac{1}{N(s)}(G - v_\pi(s))$$

RL – Evaluation – Monte-Carlo



$$\forall s \in \mathcal{S}, \quad v_\pi(s) \leftarrow v_\pi(s) + \frac{1}{N(s)}\left(G - v_\pi(s)\right)$$

Fonctionne en situation non-stationnaire ! $\longrightarrow$ $\alpha\left(G - v_\pi(s)\right)$

RL – Evaluation – Différence temporelle

Bootstraping:

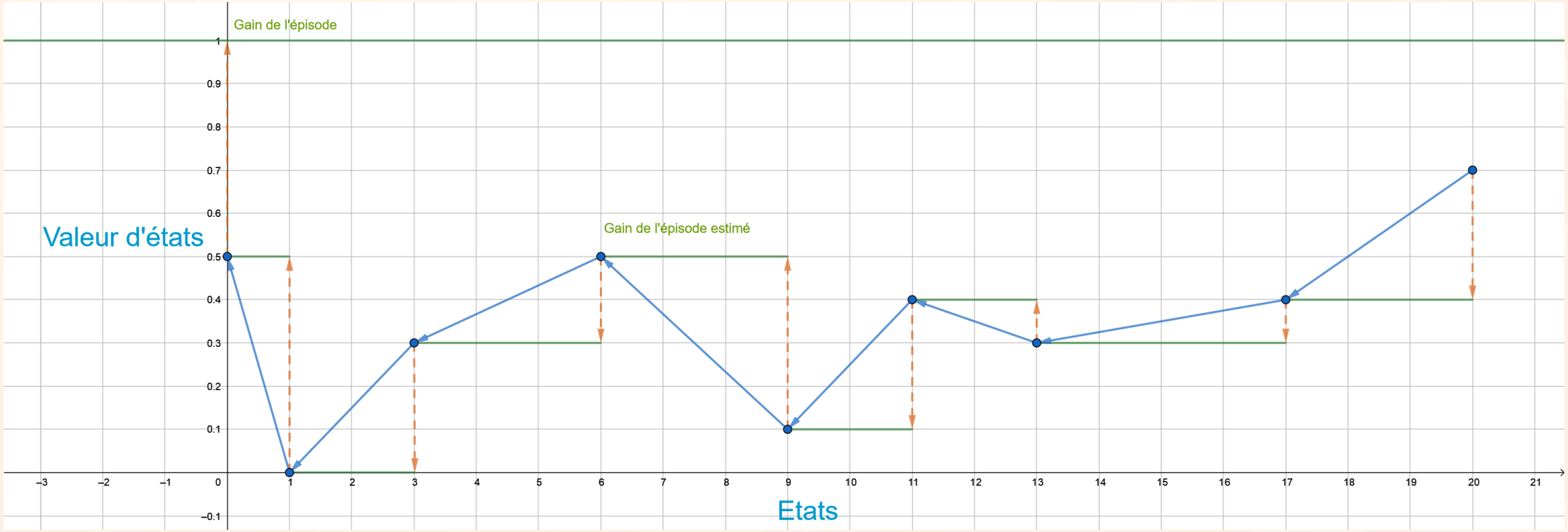
$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = R_{t+1} + \gamma G_{t+1}$$

$$G_t \simeq R_{t+1} + \gamma\, v_\pi(S_{t+1})$$

A la fin de chaque ETAPE :

$$v_\pi(S_t) \leftarrow v_\pi(S_t) + \alpha\,(R_{t+1} + \gamma\, v_\pi(S_{t+1}) - v_\pi(s))$$
$$v_\pi(S_t) \leftarrow v_\pi(S_t) + \alpha\, \delta_t$$

RL – Evaluation – Différence temporelle



$$v_\pi(S_t) \leftarrow v_\pi(S_t) + \alpha\,(R_{t+1} + \gamma\, v_\pi(S_{t+1}) - v_\pi(s))$$

Ici nulle jusqu'à la fin !

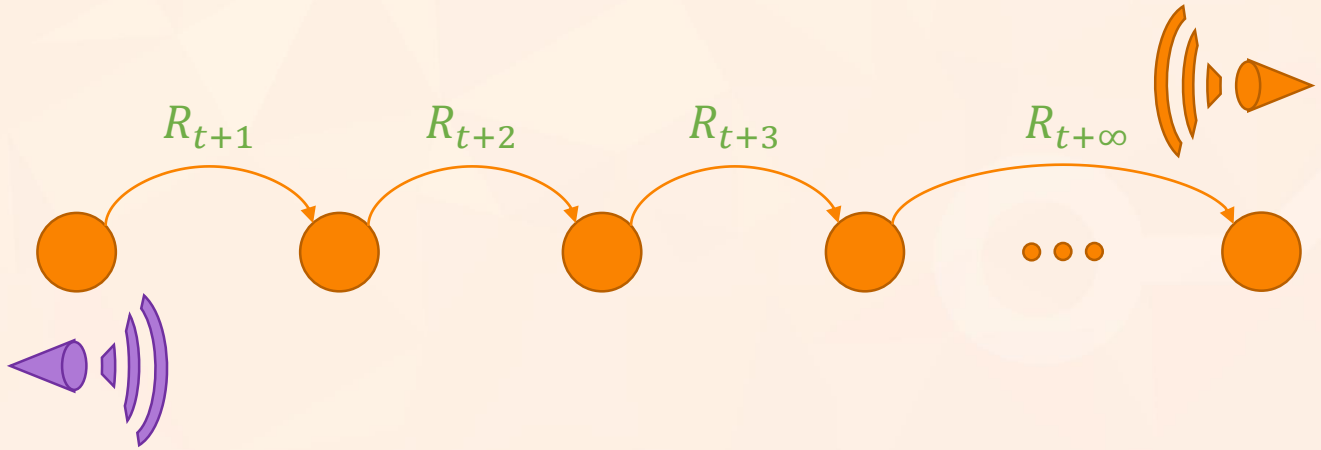
RL – Evaluation – Différence temporelle généralisée

Pourquoi ne pas aller plus loin ?

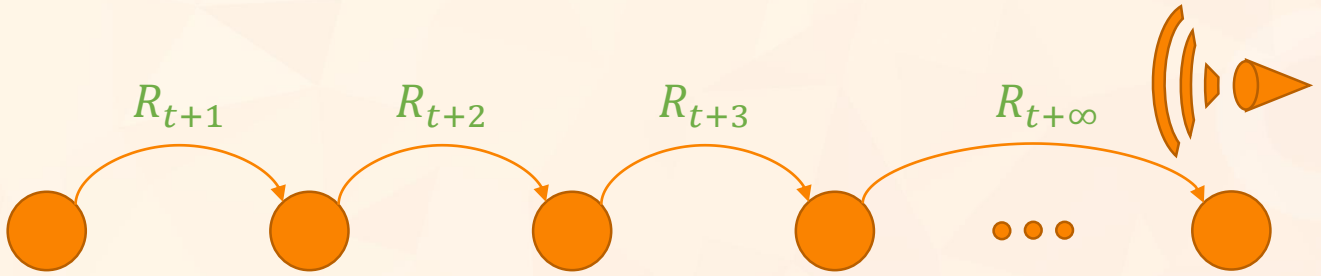
$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = R_{t+1} + \gamma R_{t+2} + \gamma^2 G_{t+2}$$

$$G_t \simeq R_{t+1} + \gamma R_{t+2} + \gamma^2 v_\pi(S_{t+2})$$

$n = 1, 2, 3, \ldots, +\infty \longleftarrow$ Monte-Carlo



RL – Evaluation – Différence temporelle généralisée



$$E_t(s) \leftarrow \gamma\lambda E_{t-1}(s) + 1_{\{s=S_t\}}$$

$$v_\pi(S_t) \leftarrow v_\pi(S_t) + \alpha\, \delta_t E_t \qquad \delta_t = (R_{t+1} + \gamma\, v_\pi(S_{t+1}) - v_\pi(s))$$

Apprentissage par renforcement

Amélioration sans modèle

RL – Amélioration avare

Relation d'ordre partielle :

$$\pi \leq \pi' \Leftrightarrow \forall s \in \mathcal{S}, v_\pi(s) \leq v_{\pi'}(s)$$

Naïvement on peut améliorer notre politique avec :

$$\pi'(s) = \underset{a \in \mathcal{A}}{argmax}\, \mathcal{R}_s^a + \sum \mathcal{P}_{ss'}^a v_\pi(s')$$

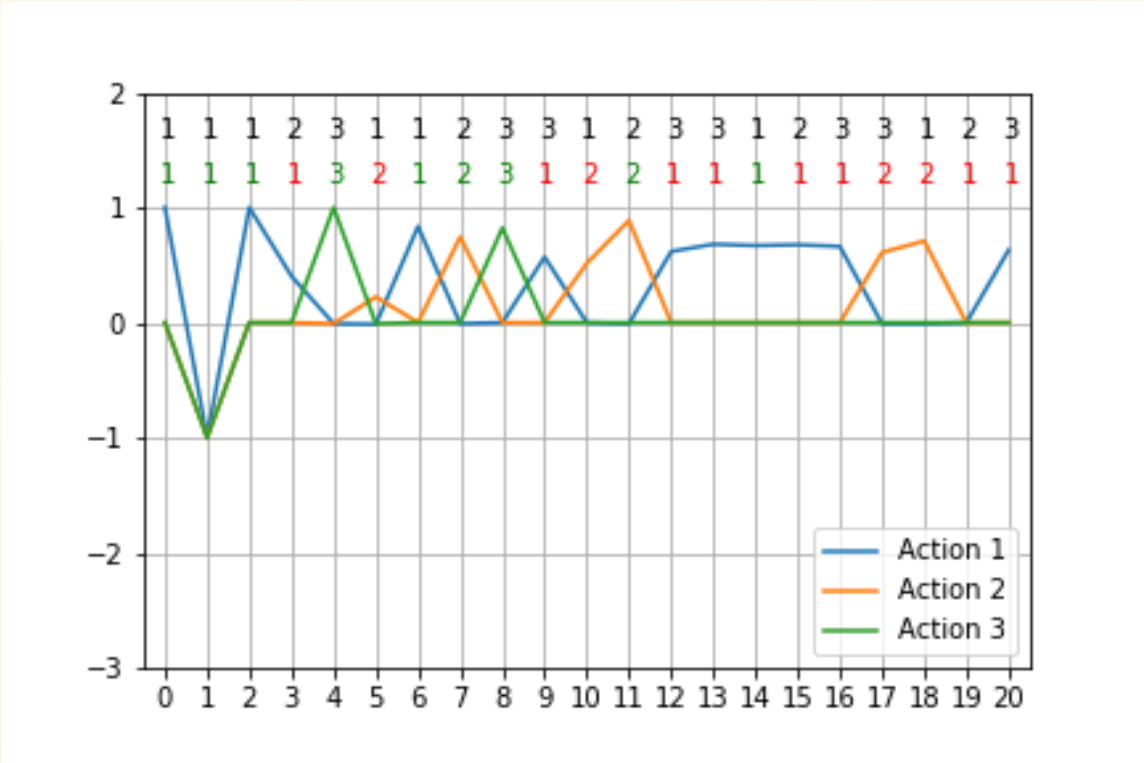
Mais on ne connait pas le modèle ! Donc on est obligé d'utiliser :

$$\pi'(s) = \underset{a \in \mathcal{A}}{argmax}\, q_\pi(s, a)$$

RL – Amélioration avare – Implémentation

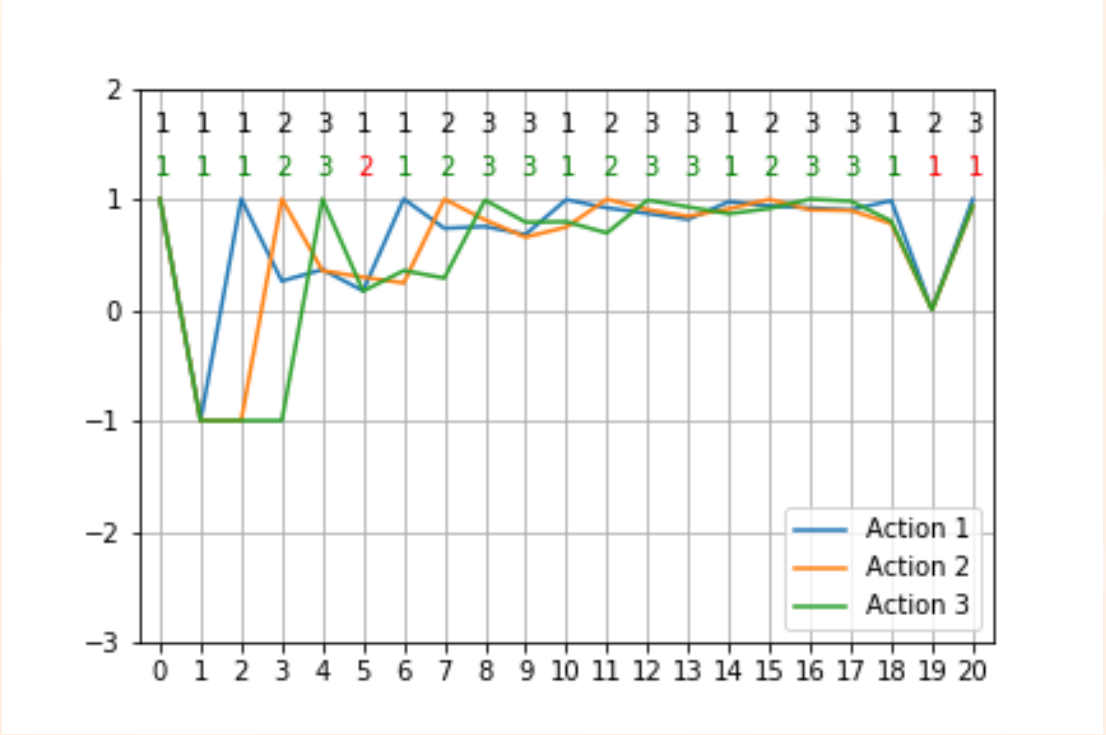
$$q_\pi(S_t, A_t) \leftarrow q_\pi(S_t, A_t) + \alpha\,(G - q_\pi(S_t, A_t))$$

$$\pi'(s) = \underset{a \in \mathcal{A}}{argmax}\; q_\pi(s, a)$$



RL – Amélioration ε-avare

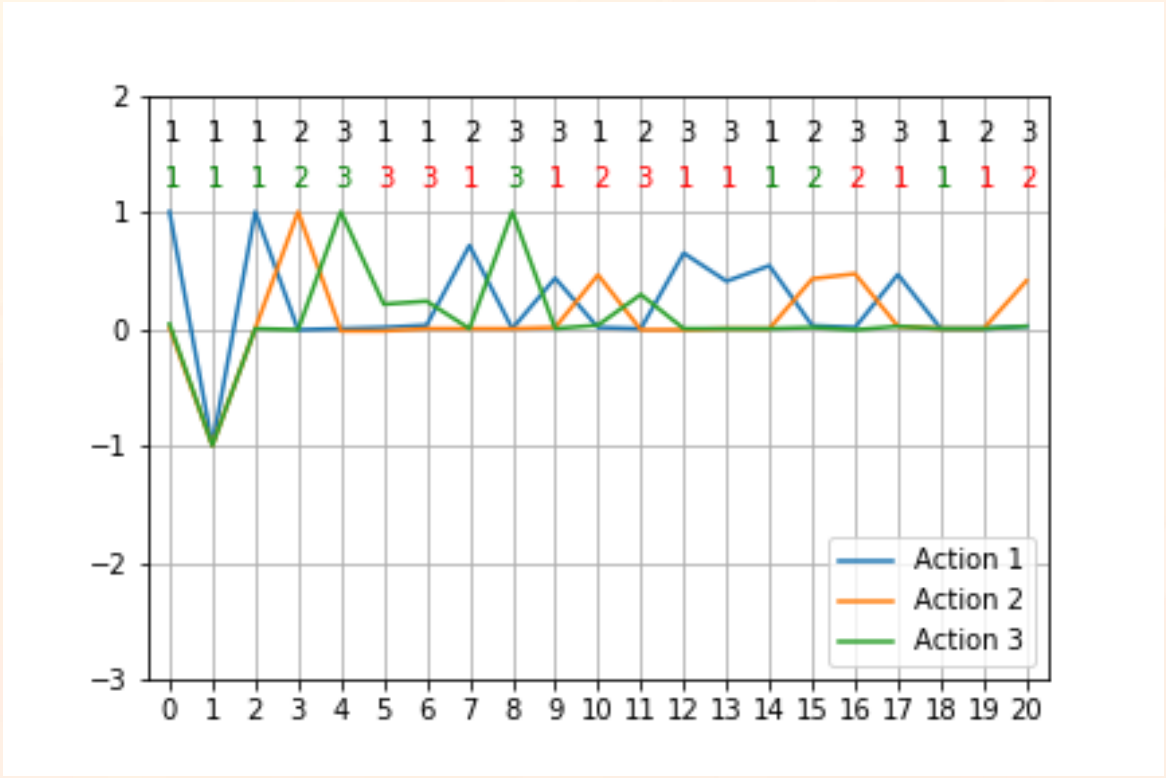
$$\pi'(s) = \begin{cases} \frac{\varepsilon}{m} + (1-\varepsilon) & si\ a = \underset{a \in \mathcal{A}}{argmax}\ q_\pi(s,a) \\ \frac{\varepsilon}{m} & sinon \end{cases}$$



RL – Amélioration – GLIE-ε-avare

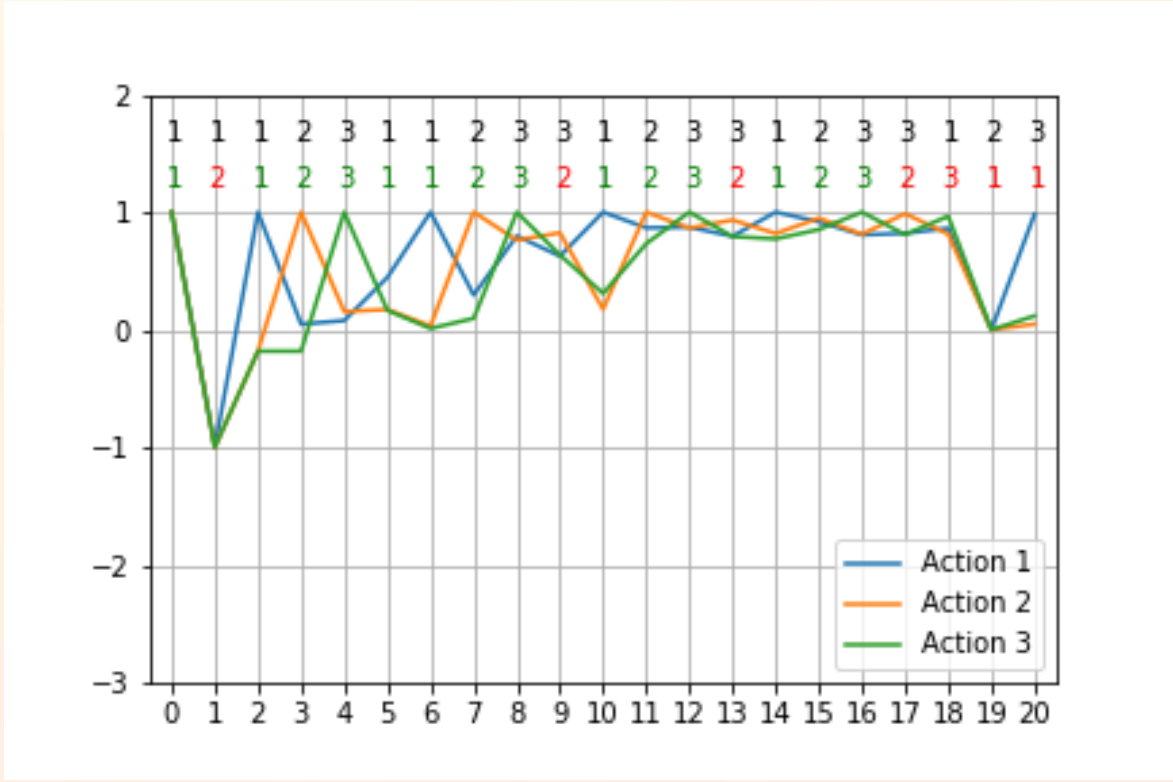
$$\forall s \in \mathcal{S}, \quad \lim_{k \to +\infty} N(s) = +\infty$$

$$\forall s \in \mathcal{S}, \quad \lim_{k \to +\infty} \pi_k(a|s) = 1_{\{a = \underset{a \in \mathcal{A}}{argmax}\ q_\pi(s,a)\}}$$



RL – Amélioration – GLIE-UCB

$$\pi'(s) = \underset{a \in \mathcal{A}}{argmax}\, q_\pi(s,a) + c\sqrt{\frac{2\ln(k)}{N(s,a)}}$$

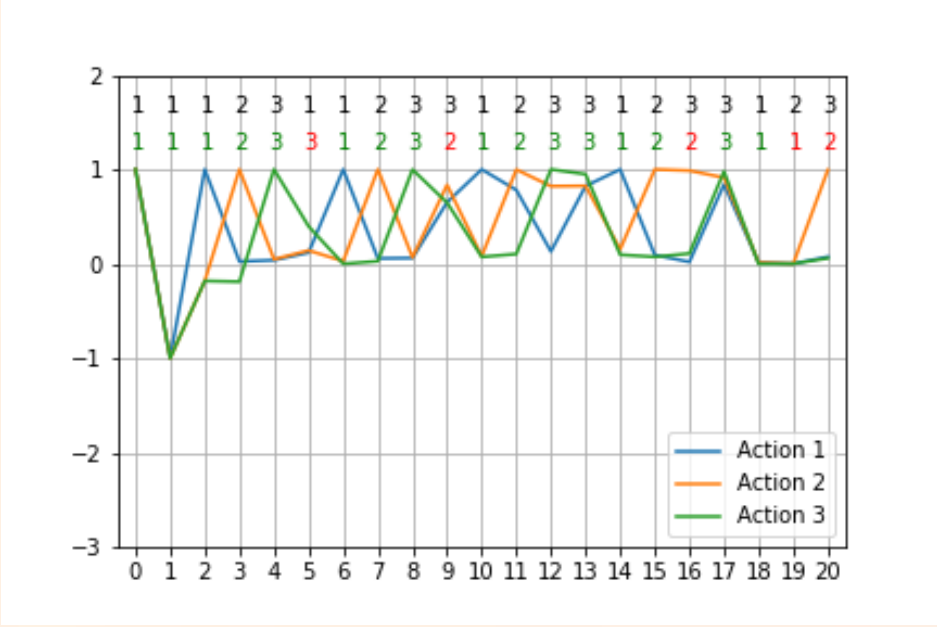


RL – SARSA

Evaluation de Q par différence temporelle !

$$q_\pi(S_t, A_t) \leftarrow q_\pi(S_t, A_t) + \alpha\,(R_{t+1} + \gamma \sum \pi(a|S_{t+1})\, q_\pi(S_{t+1}, a) - q_\pi(S_t, A_t))\, E_t$$

Nécessite beaucoup moins de parties ! (Ici 3 fois moins)



RL – Off-policy Learning

On utilise une autre politique μ qui est notre politique objectif
On utilise toujours π sur laquelle on apprend:

$$\mathbb{E}_{\mu}\big(f(X)\big) = \sum \mu(X) f(X) = \sum \pi(X) \frac{\mu(X)}{\pi(X)} f(X) = \mathbb{E}_{\pi}\left(\frac{\mu(X)}{\pi(X)} f(X)\right)$$

$$q_\pi(S_t, A_t) \leftarrow q_\pi(S_t, A_t) + \alpha \left(R_{t+1} + \gamma \underbrace{\mathbb{E}_\mu(G_t | S_{t+1} = s')}_{\text{Ce que l'on ferait avec notre politique finale}} - q_\pi(S_t, A_t)\right) E_t$$

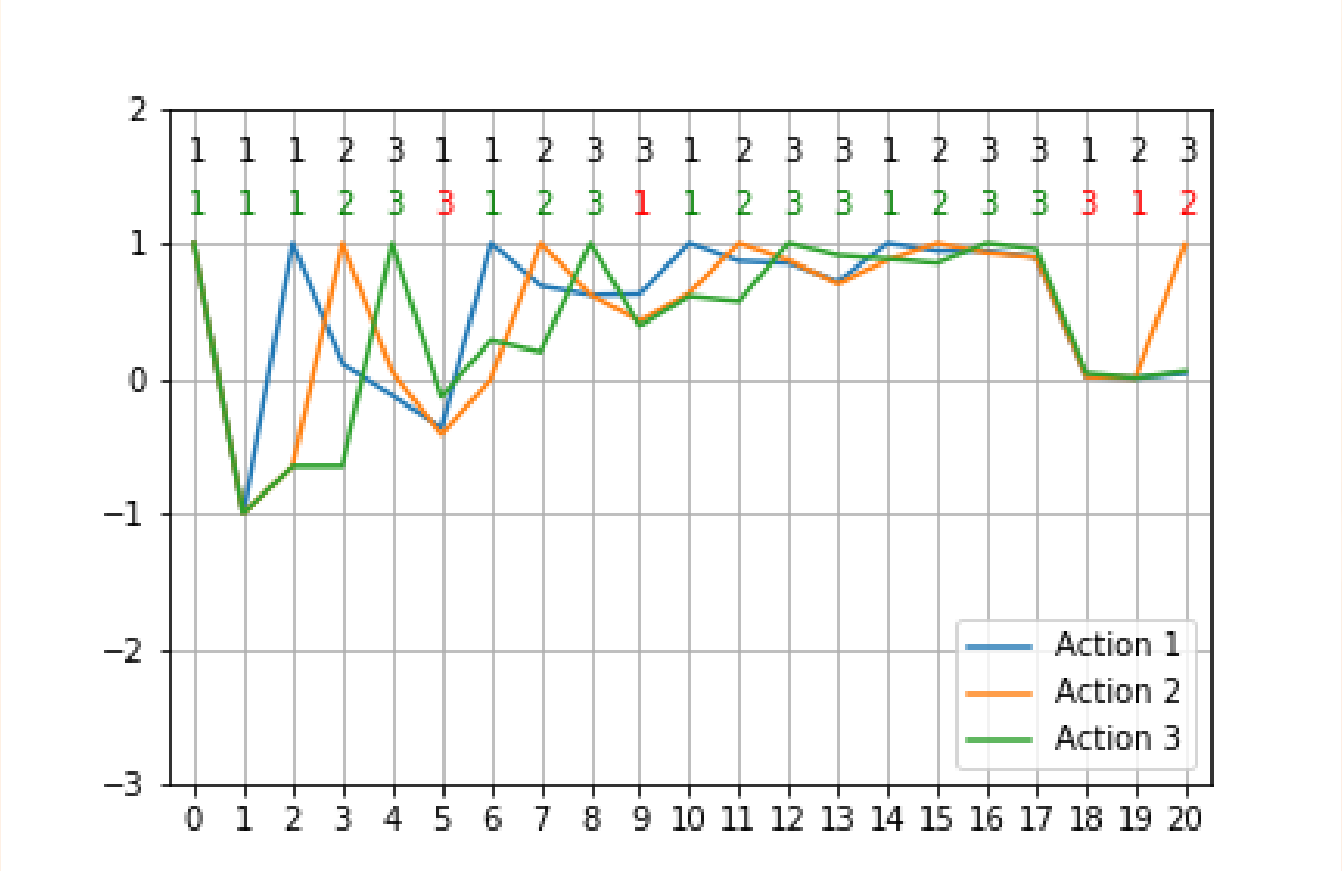
$$q_\pi(S_t, A_t) \leftarrow q_\pi(S_t, A_t) + \alpha \left(R_{t+1} + \gamma \sum \pi(a|S_{t+1}) \frac{\mu(a|S_{t+1})}{\pi(a|S_{t+1})} q_\pi(S_{t+1}, a) - q_\pi(S_t, A_t)\right) E_t$$

$$q_\pi(S_t, A_t) \leftarrow q_\pi(S_t, A_t) + \alpha \left(R_{t+1} + \gamma \sum \mu(a|S_{t+1}) q_\pi(S_{t+1}, a) - q_\pi(S_t, A_t)\right) E_t$$

RL – Q-learning

$$q_\pi(S_t, A_t) \leftarrow q_\pi(S_t, A_t) + \alpha \left(R_{t+1} + \gamma \sum greedy(a|S_{t+1})\, q_\mu(S_{t+1}, a) - q_\pi(S_t, A_t)\right) E_t$$

$$q_\pi(S_t, A_t) \leftarrow q_\pi(S_t, A_t) + \alpha \left(R_{t+1} + \gamma \max_{a \in \mathcal{A}} q_\mu(S_{t+1}, a) - q_\pi(S_t, A_t)\right) E_t$$



Merci !

Des questions ?