Cours *Apprentissage par renforcement*

(2e année Master *Mathématiques Vision Apprentissage)*

Rémi Munos

SequeL project (Sequential Learning), INRIA Lille - Nord Europe

remi.munos@inria.fr

Page web du cours : *http ://sequel.futurs.inria.fr/munos/master-mva*

Support de cours : transparents +

- Livre "Processus décisionnels de Markov et intelligence artificielle", Hermès, 2008. Disponible à : *http ://sequel.futurs.inria.fr/munos/pdmia-tome-1.pdf* et *pdmia-tome-2.pdf*.
- Livre introductif : Sutton et Barto, Reinforcement Learning : An Introduction, 1998. Disponible en ligne : http ://www-anw.cs.umass.edu/~rich/book/the-book.html
- Livre pour approfondir : Neuro-Dynamic Programming, Bertsekas et Tsitsiklis, 1996. http ://web.mit.edu/jnt/www/ndp.html

Plan du cours :

1. Introduction à l'apprentissage par renforcement
2. Introduction à la programmation dynamique, le cas discret
   (a) Processus de décision markovien
   (b) Algorithmes de programmation dynamique
3. Algorithmes d'Apprentissage par renforcement
   (a) Méthodes de Monte-Carlo, algorithmes stochastiques
4. Dilemme exploration / exploitation et algorithmes de bandit
5. Approximation de fonction et apprentissage supervisé
6. Programmation dynamique avec approximation
7. Recherche directe d'une politique paramétrée
8. (Contrôle optimal, le cas continu)

Introduction à l'apprentissage par renforcement (A/R)

- Acquisition automatisée de compétences pour la prise de décisions (**actions** ou **contrôle**) en milieu complexe et incertain.
- Apprendre par l'"expérience" une stratégie comportementale (appelée **politique**) en fonction des échecs ou succès constatés (les **renforcements** ou **récompenses**).
- **Exemples** : jeu du chaud-froid, apprentissage sensori-moteur, jeu d'échecs, robotique mobile autonome, planification, ...
- **Applications** : robotique autonome, économie, recherche opérationnelle, jeux, ...

Naissance du domaine :

Rencontre début 1980 :

- **Neurosciences computationnelles**. Renforcement des poids synaptiques des transmissions neuronales (règle de Hebbs, modèles de Rescorla et Wagner dans les années 60, 70).
  Renforcement = corrélations activités neuronales.
- **Psychologie expérimentale**. Modèles de conditionnement animal : renforcement de comportement menant à une satisfaction (recherches initiées vers 1900 par Pavlov, Skinner et le courant béhavioriste).
  Renforcement = satisfaction, plaisir ou inconfort, douleur.

Cadre mathématique adéquat :

Programmation dynamique de Bellman (années 50, 60), en théorie du contrôle optimal. Renforcement = critère à maximiser.

Conditionnement animal

Psychologie expérimentale. Thorndike (1911) :

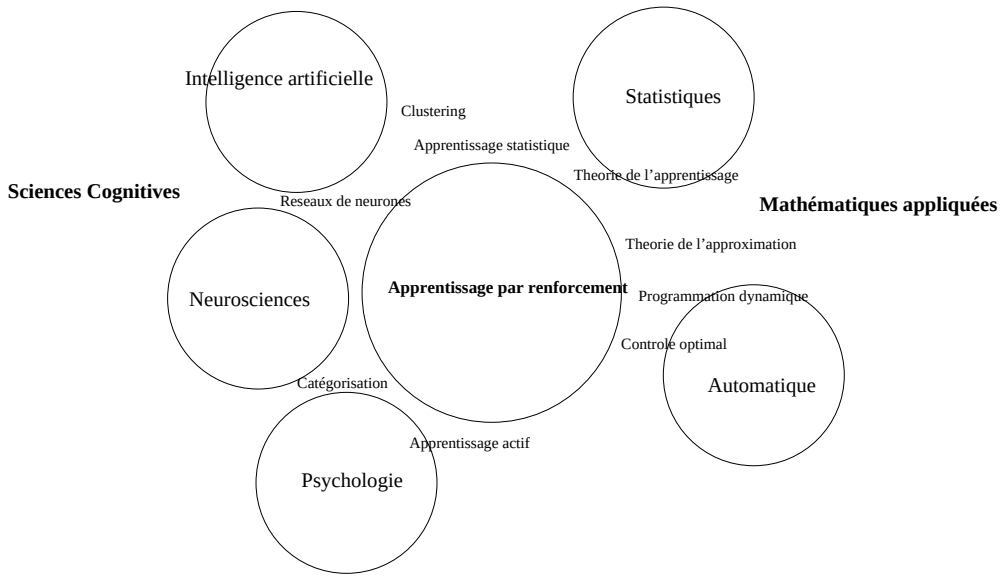
> "Of several responses made to the same situation, those which are accompanied or closely followed by satisfaction to the animal will, other things being equal, be more firmly connected with the situation, so that, when it recurs, they will be more likely to recur ; those which are accompanied or closely followed by discomfort to the animal will, other things being equal, have their connections with that situation weakened, so that, when it recurs, they will be less likely to occur. The greater the satisfaction or discomfort, the greater the strengthening or weakening of the bond"

Loi des effets.

Les débuts de l'A/R computationnel

- Shannon 1950 : Programming a computer for playing chess.
- Minsky 1954 : Theory of Neural-Analog Reinforcement Systems.
- Samuel 1959 : Studies in machine learning using the game of checkers.
- Michie 1961 : Trial and error. -> joueur de tic-tac-toe.
- Michie et Chambers 1968 : Adaptive control -> pendule inversé.
- Widrow, Gupta, Maitra 1973 : Punish/reward : learning with a critic in adaptive threshold systems -> règles neuronales.
- Sutton 1978 : Théories d'apprentissage animal : règles dirigées par des modifications dans prédictions temporelles successives.
- Barto, Sutton, Anderson 1983 : règles neuronales Actor-Critic pour le pendule inversé.
- Sutton 1984 : Temporal Credit Assignment in Reinforcement Learning.
- Klopf 1988 : A neuronal model of classical conditioning.
- Watkins 1989 : Q-learning.
- Tesauro 1992 : TD-Gammon

Un domaine pluridisciplinaire



L'apprentissage, c'est quoi ?

Règle de modification des paramètres d'un modèle, en fonction de données observées.

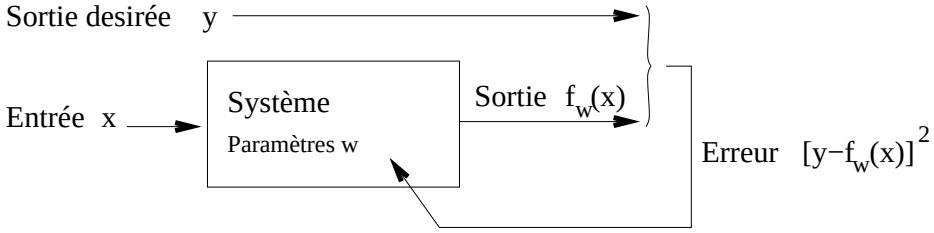
Inspiration biologique : soit au niveau des modèles (ex : réseaux de neurones), soit au niveau des comportements.

Différentes formes d'apprentissage :

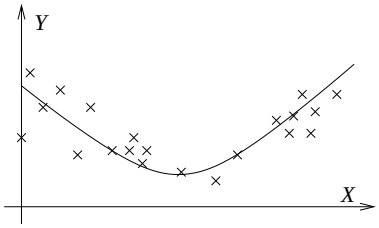
	Supervisé	Non-supervisé	Par renforcement
Qu'est ce qu'on apprend ?	relations	structures	loi d'action
Info pour l'apprentissage	sortie désirée	rien	récompense
Forme d'apprentissage	par instruction	par observation	par évaluation
Loi d'apprentissage	gradient	auto-organisation	différence temporelle

Apprentissage supervisé

Apprentissage par instruction *: Information disponible pour l'apprentissage :* la sortie désirée.



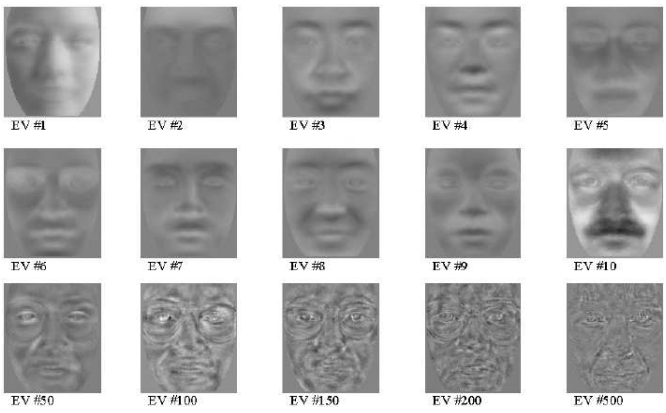
Problème d'approximation de fonction.



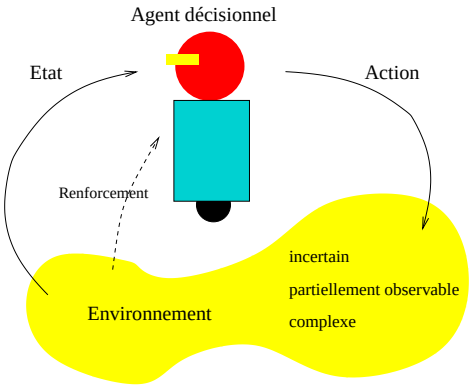
Exemples : Réseaux neuronaux, Support Vector Machines, Régression linéaire, Décomposition sur bases de cosinus, ...

Apprentissage non-supervisé

- Clustering, découverte de structure dans des données, data-mining.
- Estimation de densités, Réseaux bayésiens
- Cartes de Kohonen, ...
- Analyse en composantes principales, exemple : **eigenfaces** :



Apprentissage par renforcement



Information disponible pour l'apprentissage : le renforcement.

Les dynamiques sont aléatoires et partiellement inconnues.

Objectif : maximiser l'espérance de gain.

Apprentissage par évaluation

L'environnement :

- Déterministe ou stochastique (ex : backgammon)
- Hostile (ex : jeu d'échecs) ou non (ex : jeu Tétris)
- Partiellement observable (ex : robotique mobile)
- Connu ou inconnu (ex : vélo) de l'agent décisionnel

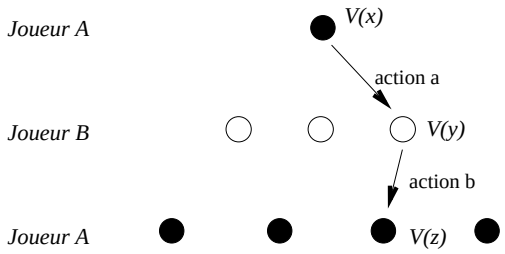
Le renforcement :

- retardé dans le temps –> problème du "credit-assignment" : quelles actions doivent être accréditées pour un renforcement obtenu au terme d'une séquence de décisions ?

Comment sacrifier petit gain à court terme pour privilégier meilleur gain à long terme ?

Fonction valeur :

- Evaluation de chaque état si l'agent agit optimalement.
- Evaluation par recherche en profondeur. Ex : jeu à 2 joueurs à somme nulle :

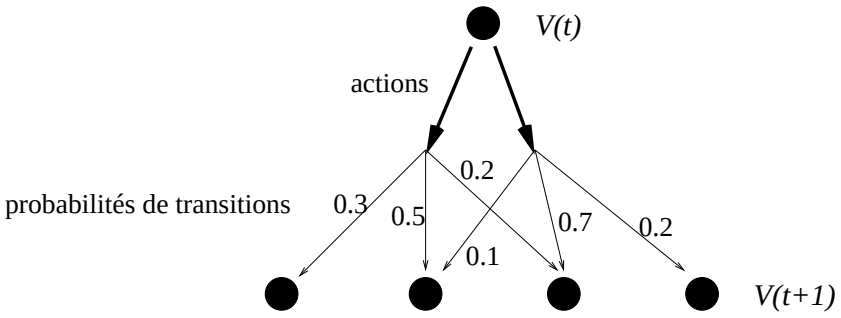


$$V(x) = \max_a \left(r(x,a) + \min_b [r(y,b) + V(z)] \right)$$

- Peut être améliorée par recherche en profondeur : si $\tilde{V}$ est une approximation de V, alors $\max_a \left(r(x,a) + \min_b [r(y,b) + \tilde{V}(z)] \right)$ est une meilleure approximation.

Fonction valeur

Ex : environnement stochastique, sans adversaire :



- Si V est connue, **en moyenne, pas de surprise** : si je choisis l'action optimale,

$$\mathbb{E}[V(t+1) + r(t) - V(t)] = 0$$

- Comment apprendre la fonction valeur ? **Grâce à la surprise** : si $\tilde{V}(t+1) + r(t) - \tilde{V}(t) > 0$ alors il n'y a pas cohérence de l'évaluation ($\tilde{V}(t)$ était sous-évaluée).

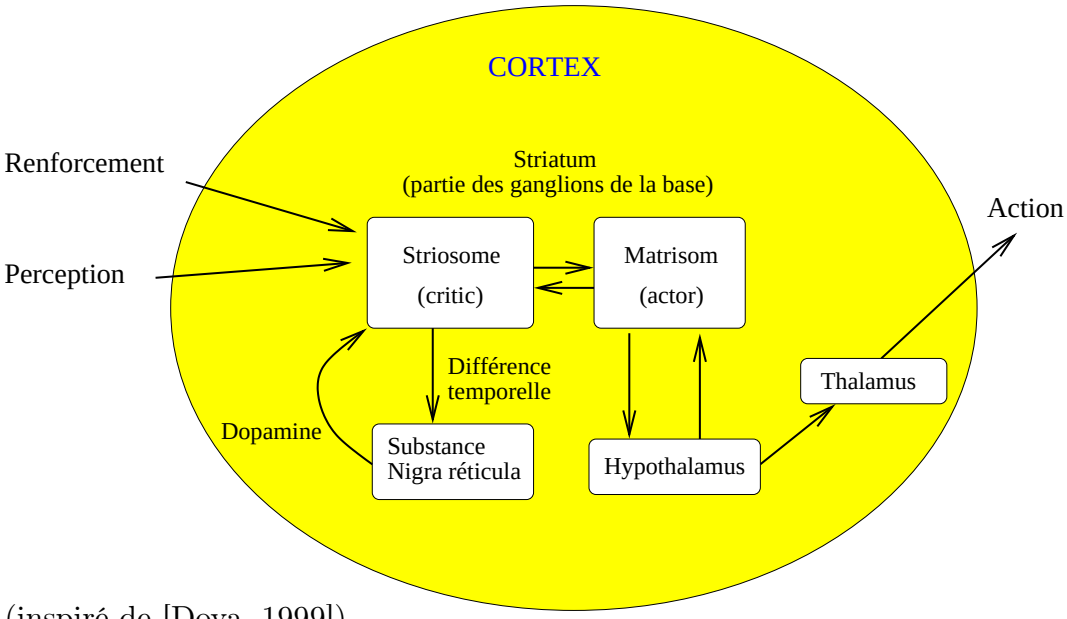
Fonction valeur

– Permet d'anticiper les récompenses à venir.
– Si elle est connue, permet de choisir, à chaque instant, la meilleure action –> maximiser localement la fonction valeur revient à maximiser le renforcement à long terme.

Lien avec les neurosciences :

– *Théorie des émotions*. Lien entre juste appréciation des émotions en fonction de la situation vécue et capacités de prises de décisions adéquates [Damasio, L'erreur de Descartes, la raison des émotions, 2001].
– neurotransmetteurs du renforcement : dopamine –> surprise.

Modèle des ganglions de la base



(inspiré de [Doya, 1999])

Quelques problématiques de l'A/R

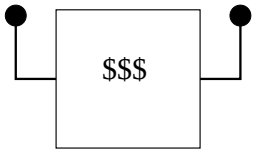
A/R = méthodes visant à résoudre de manière adaptative un problème de contrôle optimal stochastique sous, au moins, une des deux problématiques suivantes :

1. Les dynamique d'état ou les récompenses sont partiellement inconnues : (Ex : vélo) 2 approches possibles :
   - **A/R indirect** : apprentissage préalable d'un modèle des dynamiques (forme d'apprentissage supervisé),
   - **A/R direct** : agir directement sans cette étape de modélisation (par exemple si les dynamiques d'état sont complexes alors que le contrôleur est simple).
2. La grande complexité du problème interdit sa résolution exacte. –> nécessaire **utilisation d'approximations** : problème de la généralisation (ex : le programme TD-gammon).

Dilemme Exploration / Exploitation

Exploiter (agir en maximisant) selon la connaissance actuelle, ou **explorer** (améliorer notre connaissance du domaine).

Exemple : **Le bandit à plusieurs bras**



- A chaque instant, sélectionne un bras a, reçoit récompense r_t : variable aléatoire de loi inconnue (une loi pour chaque bras).
- Récompenses déjà reçues : $6\$, 7\$, 4\$, 7\$$ pour le bras gauche, $10\$, 0\$$ pour le bras droit. Quel bras choisir ?
- Il ne faut jamais s'arrêter d'explorer, mais il faut explorer de moins en moins fréquemment.
- Algorithmes "optimistes dans l'incertain" [Lai et Robbins, 1985]

Choix de l'action

Supposons qu'on estime les "qualités" des actions $Q_k(a) = \frac{1}{k_a} \sum_{t=1}^{k_a} r(t,a)$ (moyenne empirique des récompenses reçues)

– *Exploitation* : choix de l'action "**gloutonne**" à l'instant k

$$\arg\max_a Q_k(a)$$

– *Exploration* : choix stochastique, selon certaines probabilités :

$$p(a) = \frac{e^{\frac{1}{T}Q_k(a)}}{\sum_b e^{\frac{1}{T}Q_k(b)}}$$

(distribution de Boltzman) T = température.

Autre exemple : choix UCB (Upper Confidence Bound) [Auer et al. 2002] :

$$\arg\max_a \left[Q_k(a) + \sqrt{\frac{\log k}{k_a}} \right]$$

Problème non-stationnaire

Les lois changent lentement au cours du temps.

Règle incrémentale de mise à jour des qualités :

$$Q_k(a) = (1-\gamma) r(k,a) + \gamma Q_{k-1}(a)$$

moyenne avec poids à décroissance exponentielle des récompenses reçues :

$$Q_k(a) = (1-\gamma) \sum_{t=0}^{k} \gamma^{k-t} r(t,a)$$

Et si les récompenses sont retardées, et les lois changent selon l'état ?

–> algorithme du **Q-learning** (Watkins, 1989).

Quelques applications :

- TD-Gammon. [Tesauro 1992-1995] : jeu de backgammon. Produit le meilleur joueur mondial !
- KnightCap [Baxter et al. 1998] : jeu d'échec ($\simeq$2500 ELO)
- Robotique : jongleurs, balanciers, acrobots, ... [Schaal et Atkeson, 1994]
- Robotique mobile, navigation : robot guide au musée Smithonian [Thrun et al., 1999], ...
- Commande d'une batterie d'ascenseurs [Crites et Barto, 1996],
- Routage de paquets [Boyan et Littman, 1993],
- Ordonnancement de tâches [Zhang et Dietterich, 1995],
- Maintenance de machines [Mahadevan et al., 1997],
- Contrôle de la circulation dans une ville (feux rouge), ...
- On verra aussi (mais est-ce vraiment de l'A/R ?)
  - Jeu de poker (algos de minimisation de regret)
  - jeu de go (UCT et algos exploratoires pour la recherche arborescente)

Introduction à la programmation dynamique, le cas discret

Références bibliographiques :

Livres :

- Bertsekas et Tsitsiklis : *Neuro Dynamic Programming*, 1996.
- Puterman : Markov Decision Problems, 1994.
- Livre PDMIA, chapitre 1.

Chaîne de Markov

– Système dynamique à temps discret $(x_t)_{t \in I\!N} \in X$. Toute l'information pertinente pour la prédiction du futur est contenue dans l'état présent (**propriété de Markov**) :

$$P(x_{t+1} = x \,|\, x_t, x_{t-1}, \ldots, x_0) = P(x_{t+1} = x \,|\, x_t)$$

– On peut alors définir des *probabilités de transition* : $p(y|x) = P(x_{t+1} = y | x_t = x)$.

Lorsque les transitions dépendent d'actions, on parle de **Processus de Décision Markovien**. On définit alors les probabilités de transition $p(y|x, a)$.

– Exemple : système dynamique (positions, vitesses, ...), Tétris, ...

Processus de Décision Markovien

Bellman 1957, Howard 1960, Dubins et Savage 1965, Fleming et Rishel 1975, Bertsekas 1987, Puterman 1994.

Défini par $(\mathcal{T}, X, A, p, r)$:

– $\mathcal{T}$: instants de décision. Discet $\mathcal{T} = \{t_0, t_1, t_2, \ldots\}$ ou continu. Intervalles : constants ou pas, dépendant de l'état, aléatoire.
– X : espace d'état. Discret ou continu.
– A : espace d'actions (ou décisions, ou contrôles)
– $p(y|x, a)$: probabilités de transition d'un état $x \in X$ à $y \in X$ lorsque l'action $a \in A$ est choisie,
– $r(x, a)$: récompense obtenue en choisissant l'action $a \in A$ en $x \in X$ (peut être une variable aléatoire).

Politique

Règle de décision : π_t détermine, à l'instant t, une loi d'action en tout état :

– déterministe : $\pi_t : X \to A$,

$$\pi_t(x) = \text{ action choisie en } x,$$

– stochastique : $\pi_t : X \times A \to I\!R$,

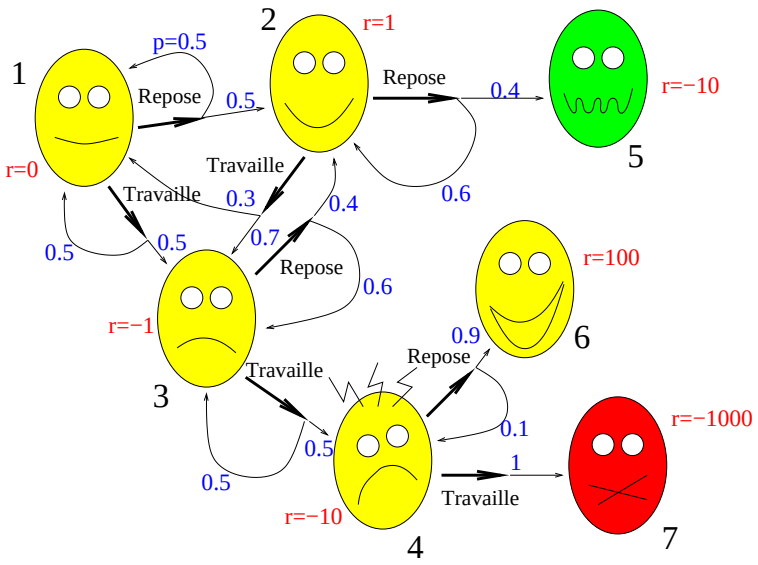
$$\pi_t(a|x) = \text{ probabilité de choisir } a \text{ en } x.$$

Politique (ou **stratégie**, ou **plan**) : $\pi = (\pi_0, \pi_1, \pi_2, \dots,)$

Politique *stationnaire* ou *Markovienne* : $\pi = (\pi, \pi, \pi, \dots)$.

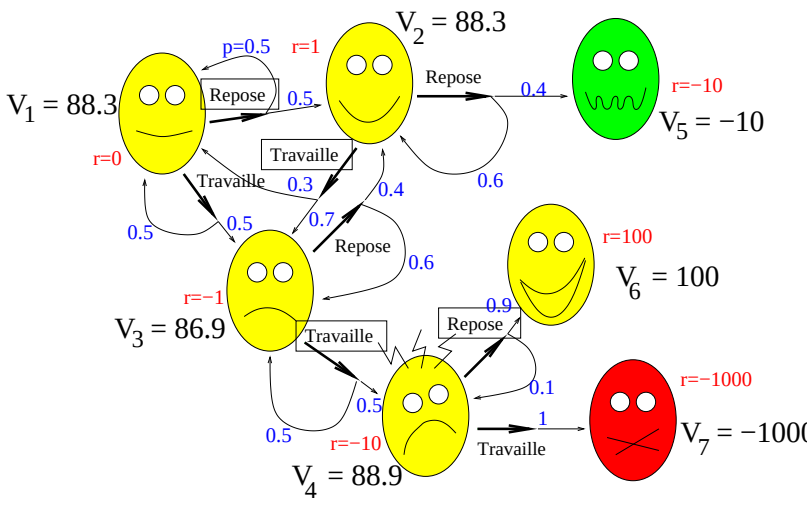
–> Pour une politique $\pi = (\pi_0, \pi_1, \dots)$ donnée, la séquence d'états $(x_t)_{t \geq 0}$ devient une chaîne de Markov de probabilités de transition :
$p(y|x) = p(y|x, \pi_t(x))$.

Exemple : le dilemme de l'étudiant MVA



Il cherche à maximiser la somme des récompenses !

Solution de l'étudiant MVA



$V_5 = -10$, $V_6 = 100$, $V_7 = -1000$, $V_4 = -10 + 0.9V_6 + 0.1V_4 \simeq 88.9$.
$V_3 = -1 + 0.5V_4 + 0.5V_3 \simeq 86.9$. $V_2 = 1 + 0.7V_3 + 0.3V_1$ et
$V_1 = \max\{0.5V_2 + 0.5V_1, 0.5V_3 + 0.5V_1\}$, soit : $V_1 = V_2 = 88.3$.

Evaluation du cours

Mini-projets : par binômes : travail à partir d'un article (une liste sera proposée sur le site bientôt, ou tout autre article de votre choix avec mon accord)

- Lecture, compréhension de l'algorithme,
- Application à un problème approprié de votre choix,
- Mise en oeuvre et réalisation informatique (langage de votre choix),
- Rapport + soutenance.

Possibilité de sujet adapté à vos centres d'intérêt (+ théorique, + biologique, ...)