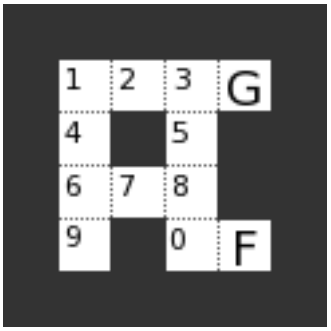
Université Paris Cité

Apprentissage par renforcement

TD MDP

Exercice 1 – Petit labyrinthe

Soit l'environnement suivant :



A tout moment, l'agent occupe une des cases de l'environnement contenant un caractère et peut effectuer une des 4 actions suivantes : Nord, Ouest, Sud, Est. Chaque action consiste à déplacer l'agent d'une case dans la direction spécifiée par le nom de l'action. Si un tel déplacement implique d'occuper une case pleine, alors l'action est sans effet.

L'agent reçoit une récompense de 1000 s'il effectue l'action qui mène à la case F, 400 s'il effectue l'action qui mène à la case G, et 0 le reste du temps. A noter que les états F et G sont terminaux.

1. Représenter cet environnement sous la forme d'un MDP
2. Soit π la politique consistant à sélectionner, dans chaque état, la première action qui provoque un changement d'état en suivant l'ordre suivant : Ouest, Sud, Est, Nord. Evaluer π avec $\gamma = 0.5$ (calculer les valeurs d'état V^{π})
3. Exécuter l'algorithme des politique itérées à partir de la politique π (avec $\gamma = 0.5$)
4. Exécuter l'algorithme des valeurs itérées dans cet environnement (avec $\gamma = 0.5$)
5. Que ce passera-t-il si γ vaut 0.9 ? Même question pour 0 et pour 1

Exercice 2 – Le lac gelé

Un frisbee a été envoyé sur un lac gelé et on souhaite le récupérer. Quelles actions exécuter pour atteindre le frisbee sans tomber dans les trous ? Attention le lac est glissant....

Cet contexte est décrit par l'environnement FrozenLake4x4 (OpenAI/Gym) :

```
S F F F
F H F H
F F F H
H F F G
```

S : état de départ
F : zone sûre
H : trou
G : position du frisbee

Les états H et G sont des état terminaux. Toute action amenant à G provoque un renforcement de +1.

Les actions possibles sont les déplacements Nord, Ouest, Sud, Est mais l'environnement est glissant : chaque exécution d'action aura au hasard l'effet de l'action prévue ou d'une des deux actions voisines (mais pas l'action opposée). Par exemple choisir l'action Nord provoquera au hasard l'effet des actions Est, Nord ou Ouest (mais pas sud).

1. Représenter cet environnement sous la forme d'un MDP
2. Calculer la politique optimale de cet environnement

Exercice 3 – Implémentation

Le fichier base.py propose une implémentation de quelques environnements modélisés sous la forme de MDP. Les classes sont décrites dans la documentation.

L'objectif de l'exercice est de prendre en main le module :

- Créer un environnement **Maze** et l'afficher
- Créer une boucle d'action pour permettre de déplacer l'agent
- Comment représenter les fonctions V et Q et les politiques ?
- Construire une politique aléatoirement, et observer cette politique dans l'environnement
- Initialiser arbitrairement des valeurs d'états et observer ces valeurs dans l'environnement

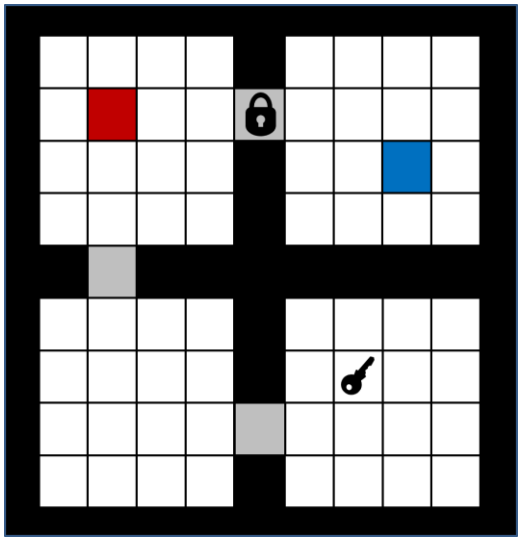
Exercice 4 – Comparaisons

Le but de l'exercice est d'implémenter les algorithmes *Policy Iteration* et *Value Iteration*, puis de les tester dans les environnements des exercices précédents.

Etapes d'implémentation (conseils) :

- Ecrire une fonction calculant Q en fonction de V
- Ecrire une fonction *greedy* à une fonction V
- Implémenter IPE
- Implémenter PI
- Implémenter VI

Comparer les performances des algorithmes (temps de calcul) dans l'environnement *FoorRooms_Key* suivant :



Dans cet environnement, l'agent démarre dans la case rouge et reçoit un renforcement de 1 s'il atteint la case bleue. Les cases grises sont des « portes ». Elles peuvent être traversées librement à moins qu'elles ne soient verrouillées (symbole cadenas). Celle du nord est verrouillée et ne peut être passée que si l'agent a auparavant ramassé la clé (il est passé par la case avec la clé).