

[Nom de la société]

Q learning

Benkabilia kawtar zineb G1

BENKABLIA KAWTAR_ZINEB

[Date]

Q learning :

Principe :

En intelligence artificielle, plus précisément en apprentissage automatique, le Q-learning est une technique d'apprentissage par renforcement. Cette technique ne nécessite aucun modèle initial de l'environnement. La lettre 'Q' désigne la fonction qui mesure la qualité d'une action exécutée dans un état donné du système

Le Q-learning est un algorithme d'apprentissage par renforcement qui cherche à trouver la meilleure action à entreprendre compte tenu de l'état actuel. Il est considéré comme hors politique parce que la fonction de Q-learning apprend des actions qui sont en dehors de la politique actuelle, comme prendre des actions aléatoires, et donc une politique n'est pas nécessaire. Plus précisément, le Q-learning cherche à apprendre une politique qui maximise la récompense totale.

Création de la table Q :

C'est une matrice qui suit la forme [état, action] et qui doit être initialisé à zéro. ensuite elle sera mis à jour en stockons les q-valeurs après chaque épisode. Ce tableau de valeurs devient une table de référence pour l'agent qui sélectionne la meilleure action en fonction des valeurs de cette matrice.

```
1 import numpy as np
2
3 # Initialize q-table values to 0
4 Q = np.zeros((state_size, action_size))
```

Un agent interagit avec l'environnement de deux manières. La première consiste à utiliser la table Q comme référence et à visualiser toutes les actions possibles pour un état donné. L'agent sélectionne alors l'action basée sur la valeur maximale de ces actions (exploitation)

La deuxième manière consiste à agir de façon aléatoire (l'exploration). Au lieu de sélectionner des actions en fonction de la récompense future maximale, nous choisissons une action au hasard. Agir au hasard est important car cela permet à l'agent d'explorer et de découvrir de nouveaux états qui, autrement, pourraient ne pas être sélectionnés pendant le processus d'exploitation.

```
1 import random
2
3 # Set the percent you want to explore
4 epsilon = 0.2
5
6 if random.uniform(0, 1) < epsilon:
7     """
8     Explore: select a random action
9     """
10 else:
11     """
12     Exploit: select the action with max value
13     """
```

Mise à jour de la table q :

$Q[\text{state}, \text{action}] = Q[\text{state}, \text{action}] + lr * (\text{reward} + \gamma * \max_a(Q[\text{new_state}, a]) - Q[\text{state}, \text{action}])$

Conclusion

Dans ce petit rapport j'ai essayé de comprendre à peu près le principe de q learning, J'ai essayé d'implémenter l'algorithme mais je n'ai pas su comment et je n'ai pas compris sur quel problème on l'applique.