Apprentissage par renforcement: Méthode Acteur-Critique

Sami EL KATEB Thomas PAUL

1 Introduction

Acteur-critique est une méthode d'apprentissage par renforcement de type policy-gradient. Contrairement à d'autres méthodes de ce type, acteur-critique utilise une approximation de la fonction de valeur d'état $\hat{v}(s_t)$ et la différence temporelle pour apprendre la politique. Dans ce document, nous détaillerons le fonctionnement de la méthode acteur-critique, ses avantages et ses inconvénients. Puis, nous examinerons une implémentation spécifique pour le jeu de Pong et explorerons des pistes d'amélioration [1].

2 Contexte

La méthode acteur-critique se base sur deux composants principaux : l'acteur et le critique. Ces deux composants interagissent pour apprendre la meilleure politique possible pour un environnement donné.

L'acteur est responsable du choix de l'action, il représente la politique $\pi(a|s)$. L'objectif de l'acteur est d'ajuster la politique pour maximiser les récompenses.

Le critique a pour objectif d'approximer la fonction de valeur d'un état $\hat{v}(s_t)$. Celle-ci estime la récompense réduite attendue[1] à partir d'un état. Le critique utilise la différence temporelle, qui est l'écart entre l'estimation actuelle de la récompense et celle obtenue, pour mettre à jour ses estimations.

Ce type d'architecture rappelle les réseaux de neurones de la famille des GANs[2] où un générateur et un discriminateur interagissent pour s'améliorer mutuellement. Son avantages par rapport aux autres méthodes de type policy-gradient est que le critique permet de réduire la variance élevée que peuvent avoir les récompenses.

Un exemple de pseudocode pour la version de l'algorithme que nous avons utilisé est donné dans la figure ci-dessous.

Algorithm 1 Algorithme Acteur-Critique

```
1:  for each episode do
2:      Initialize s_t, R ← [ ], S ← [ ], A ← [ ]
3:      for each time step do
4:          a_t ~ π(s_t)          ▷ Action aléatoire a_t sélectionnée selon la distribution définie par l'acteur
5:          Take action a_t, observe s_{t+1} and r_{t+1}
6:          append a_t to A, r_{t+1} to R, s_{t+1} to S
7:          s_t ← s_{t+1}
8:      end for
9:      G ← discounted R
10:     advantages ← G − v̂(S)
11:     Fit π using S and A with advantages as sample weights
12:     Fit v̂ using S and G
13: end for
```

À chaque étape l'acteur retourne une distribution de probabilités en fonction de l'état. L'algorithme acteur-critique utilise alors la préférence d'actions soft-max pour trouver un équilibre entre l'exploration et l'exploitation (ligne 4). Contrairement à l'algorithme ϵ-greedy, cette technique consiste à choisir une action aléatoirement en fonction des probabilités déterminées par l'acteur. [2]

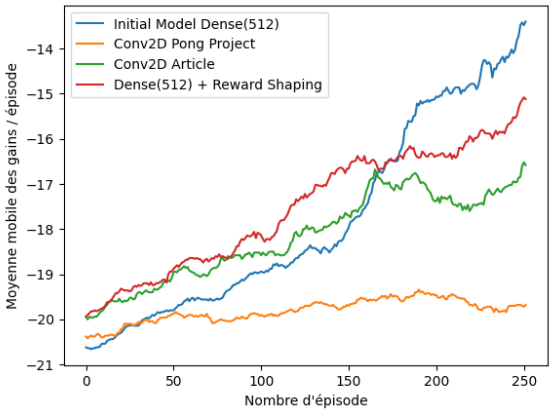
Une fois l'épisode terminé, on peut appliquer une réduction au récompenses perçues en fonction d'un facteur γ (ligne 9). Le critique évalue alors les différents états de l'épisode et retourne les valeurs prédites (ligne 10). Ces prédictions sont ensuite soustraites aux récompenses réelles pour évaluer les avantages obtenus par l'acteur (ligne 10). Les avantages peuvent ensuite être utilisés pour améliorer la politique de l'acteur et les récompenses réelles sont utilisés pour entrainer le critique (lignes 11-12).

[1] Expected Discounted Return
[2] Generative Adversarial Network

3 Application au jeu de Pong

Pour appliquer l'algorithme Acteur-Critique au jeu de Pong nous avons pris pour référence le code source de *Learning to Play with Pong*[1]. Ce repository propose une implémentation de la version Advantage Actor-Critic (A2C) que nous avons adapté à la nouvelle api de gymnasium.

Nous avons par la suite tenté plusieurs modifications dans le but d'améliorer les performances de l'algorithme. Pour les évaluer, nous avons analysé l'évolution de la moyenne mobile des gains par épisode. Chaque entrainement a été effectué sur 300 épisodes et a duré en moyenne 3 heures. Bien que le nombre d'épisode nécessaire pour vaincre l'IA soit plus élevé, nous avons choisit un nombre d'épisode nous permettant d'entrainer plusieurs modèles pour pouvoir ainsi les comparer.



Notre première modification a consisté à reprendre le même réseau de convolution que celui utilisé pour le projet Pong. Cependant, cette approche n'a pas été concluante. En effet, pour le projet Pong, nous avons utilisé un réseau convolutionnel avec un nombre restreint de paramètres. Nous avons pris cette décision car nous disposions d'un nombre limité de données labellisées et voulions diminuer le risque de surapprentissage.

Ensuite, nous avons choisit d'utiliser le réseau de convolution décrit dans l'article *Learning to Play Pong using Gradient Learning*[3]. Celui-ci a donné de meilleurs résultats que le réseau de convolution de notre projet, cependant, il n'a pas surpassé les performances de l'implémentation originale.

Enfin, nous avons ajouté du reward shaping au modèle initial, ce qui a donné de meilleurs résultats que nos autres implémentations. Pendant les premiers épisode cette implémentation est meilleure que le modèle initial. Elle est cependant surpassée par celui-ci aux alentours du 170$^{\text{ème}}$ épisode. Une explication serait que sur le long terme le modèle apprenant uniquement à chaque point marqué apprend les coups gagnant alors que le modèle avec reward shaping apprend uniquement à renvoyer la balle.

4 Avantages & Inconvénients

4.1 Avantages

La méthode Acteur-Critique combine les avantages des méthodes de type value-based (Q-Learning, Sarsa) et policy-based :

- Elle offre une meilleure convergence que les méthodes basées uniquement sur la politique, grâce à l'utilisation de la différence temporelle.
- Ne nécessite pas de données labellisés contrairement à l'approximation de la fonction de valeur par apprentissage supervisé.
- Contrairement à Q-Learning ou SARSA, elle peut fonctionner dans un espace d'état très large ou continu.

4.2 Inconvénients

Elle présente cependant quelques inconvénients :

- Nécessite moins d'épisodes pours converger mais les épisodes sont plus long.
- Nécessite d'apprendre à la fois une fonction de valeur et une politique, ce qui augmente la complexité et le temps de calcul de l'algorithme.
- Moins efficace que Q-Learning ou SARSA dans des espaces restreints et discret.

5 Discussions

Pour conclure, la méthode acteur-critique offre de nombreux avantages par rapport aux autres méthodes d'apprentissage par renforcement. Elle permet notamment d'exploiter la différence temporelle dans un espace continu.

Bien que nous ayons étudié une implémentation particulière de la méthode acteur-critique, il existe d'autres versions qu'il serait intéressant d'étudier. Par exemple, la méthode Asynchronous Advantage Actor-Critic (A3C) qui parallélise l'exécution sur plusieurs agents.

De même, il serait intéressant d'entrainer des modèles sur un plus grand nombre d'épisodes afin de confirmer nos observations.

Références

[1] Sagar Gubbi. *Learning to play Pong*. 2019. URL : https://github.com/s-gv/pong-keras/blob/master/pong-actor-critic.py (visité le 15/10/2023).

[2] Richard S. Sutton et Andrew G. Barto. *Reinforcement Learning : An Introduction*. Second. The MIT Press, 2018.

[3] Somnuk Phon-Amnuaisuk. *Learning to Play Pong using Policy Gradient Learning*. 2018. arXiv : 1807.08452 [cs.LG].