C'est quoi PPO?

Introduction

PPO, ou **Proximal Policy Optimization**, est un algorithme utilisé pour entraîner des agents à apprendre à prendre de bonnes décisions. Il fait partie de la famille des algorithmes d'**apprentissage par renforcement (AR)**.

1 Imagine une salle d'entraînement pour robots

PPO, c'est comme un coach d'entraînement pour un robot qui doit apprendre à réussir une tâche, comme **jouer au jeu CartPole** (garder un bâton en équilibre sur un chariot). L'algorithme observe ce que fait le robot, lui donne des **récompenses** ou des **punitions** selon ses actions, et ajuste ses stratégies pour qu'il s'améliore.

2 Le cœur de PPO : L'apprentissage des stratégies

Un agent (robot) a une **stratégie**, aussi appelée **policy**. Cette stratégie décide :

"Que dois-je faire dans cette situation?"

Exemple :

Dans CartPole, la stratégie répond à la question:

"Si le bâton penche à droite, devrais-je déplacer le chariot à droite ou à gauche?"

PPO entraîne cette stratégie pour qu'elle devienne **meilleure** au fil du temps.

3 Pourquoi PPO est utile?

Dans l'apprentissage par renforcement, on peut facilement tomber dans deux pièges:

1. **Changer trop vite la stratégie**
    - Si on ajuste la stratégie de l'agent de manière brutale, il risque de devenir instable (un peu comme un débutant qui change constamment de technique de jeu).

- **Solution de PPO :** Progresser lentement, par petits ajustements.

2. **Explorer les mauvaises actions trop souvent**

   - Si l'agent passe trop de temps à essayer des actions inutiles (par exemple, déplacer le chariot alors que le bâton est stable), il apprend moins efficacement.
   - **Solution de PPO :** Se concentrer sur ce qui marche déjà, mais sans trop ignorer les nouvelles actions.

4 Comment fonctionne PPO? (Version simplifiée)

PPO suit **3 grandes étapes** pour entraîner l'agent :

Étape 1 : Faire jouer l'agent

PPO commence par **laisser l'agent jouer dans l'environnement** (comme un débutant). L'agent teste différentes actions et reçoit des **récompenses** ou des **punitions**.

Exemple :

- Si le bâton tombe, il reçoit une punition.
- Si le bâton reste en équilibre longtemps, il reçoit une récompense.

Étape 2 : Comparer les actions

Une fois qu'il a joué, PPO regarde chaque action que l'agent a faite et se pose cette question:

"Était-ce une bonne idée?"

Pour cela, PPO compare deux choses:

- **L'ancienne stratégie :** Ce que l'agent pensait être une bonne action avant.
- **La nouvelle stratégie :** Ce que PPO pense être la meilleure action maintenant.

Étape 3 : Ajuster doucement

PPO ajuste la stratégie de l'agent en suivant cette règle:

"Ne change pas trop vite!"

Pourquoi? Si on change tout d'un coup, l'agent peut devenir instable. PPO utilise une limite appelée **clip** pour s'assurer que les ajustements sont **progressifs**.

5 Une analogie pour PPO

Imagine que tu apprends à jouer au basket.

1. **Phase 1 : Essais et erreurs**
    - Tu tires le ballon et vois si tu marques ou non.
    - Si tu rates, tu te dis: "Ok, je dois ajuster ma force ou mon angle."
2. **Phase 2 : Ajustement lent**
    - Au lieu de changer complètement ta technique, tu fais de petits ajustements :
        - "Je vais viser un peu plus haut."
        - "Je vais tirer avec un peu plus de force."
    - Cela évite de perdre le peu de progrès que tu as déjà fait.

PPO fait exactement ça pour entraîner un agent.

6 Pourquoi PPO est génial?

1. **Stable et efficace** L'agent progresse sans devenir instable.
2. **Simple à utiliser** PPO est facile à programmer (il est populaire pour cette raison).
3. **Utilisé partout** PPO est utilisé dans des jeux, des simulations, et même des applications du monde réel (robots, trading, etc.).

Résumé

- PPO est un algorithme d'**apprentissage par renforcement**.
- Il entraîne un agent à apprendre **doucement et efficacement** à jouer à un jeu ou à réussir une tâche.
- PPO ajuste la stratégie de l'agent petit à petit pour éviter les erreurs et les changements brutaux.