Démarche

Mehdi Mounsif

26 mars 2018

1 Récap : The day before

- Articles sur imitation : Third Person Imitation [4] et [5].
- Implémentation complète DDPG (target networks, experience replay et OU noise) . Résultats médiocres. A revoir.
- PPO en test : Sélection systématique de l'action ayant la plus haute probabilité. Ne tremble pas. Aurait besoin d'un peu de finesse supplémentaire pour augmenter ses performances.
- Question : Comment choisir une action continue avec PPO ?

2 IK

Progrès dans **ReacherIK** avec Behavior Cloning. Le Controlleur détermine une direction en fonction de la distance euclidienne entre le vecteur directement pointé à la cible et les directions cardinales. La direction correspond à une action qui est transmise au robot qui détermine le mouvement grâce à la jacobienne. J'ai généré 200 trajectoires avec les labels d'actions correspondants et j'ai approximé tout ceci par un réseau deux couches, 64 et 128, optimisé via ADAM. Fonctionne correctement. Il arrive que le mouvement soit saccadé mais je soupçonne le fait que les directions soient perpendiculaires, forçant ainsi le robot à gesticuler pour atteindre la cible (en effet, le mouvement est aussi saccadé dans les trajectoires.)

- DDPG/PPO en continu pourraient peut-être calculer directement la direction à fournir à la jacobienne ? (Mais si on dispose du vecteur, what's the point ?).
- Calculer et prédire la jacobienne en fonction des angles ? Possible ? Intéressant ?

Modification de ReacherIK. Ajout du solver de jacobienne. L'idée est que l'agent choisisse une action parmi n (dans ce cas, 8, correspodant aux directions cardinales) et que le vecteur soit transmis à la jacobienne pour qu'elle calcule l'incrément dans les angles.

Question : Serait-il possible d'approximer des jacobiennes plus importantes pour robotique ?

3 FloydHub

Compréhension du service de sauvegarde. Les sauvegardes sont à envoyer dans le dossier `/output/` de Floyd. D'ici, elles sont alors accessibles et téléchargeables. Bien se rappeler d'intialiser Floyd dans le dossier avec `ss`

4 PPO continuous

Pour l'apprentissage en général, écrire un nouveau plannificateur de difficulté qui place la cible dans un rayon autour de la tête. Eviter de réinitialiser la position du robot à chaque étape. Rien de neuf. Penser à lire la réponse de Wang.

5 DDPG

Pas de poursuite aujourd'hui. A revoir. D'après Anjum84 sur OpenAI forums, DDPG est simplement plus vieux que PPO. Ainsi, s'il est possible de générer des actions continues avec PPO, nul besoin de s'encombrer avec DDPG.

6 Résultats de la réunion

Intégrer plus de robotique. Permettra de publier rapidos (et d'obtenir plus de liberté).

Feuille de route : Robotique

- Prédire la Jacobienne avec NN.
- Modèle géométrique indirect. Fonctionne. Transposée de la Jacobienne itérative.
- Déplacer le robot avec ces modèles analytiques. Fonctionne sur LowReacher et Reacher
- Injecter du bruit dans les capteurs et le gérer avec RL.

Feuille de route : IA

- DDPG
- IRL : Regarder GAIL [2]
- GAN : Des progrès. MNIST. Génération Pokémons (approximative)
- Model-based : Pas étudié. Ai parcouru la conférence de Chealsea Finn sur Youtube. Sample efficient, prévoit l'environnement. A investiguer

Feuille de route : Misc

- Tester PPO sur Atari : Implémentation propre de AC, A2C, PPO à la maison. Rajouter CNN et tester sur Atari
- Tester sur SNESx9 : Détection de rectangles avec un réseau de neurones.
- Animation IK : Pas de progrès

7 DDPG

Implémentation d'une version allégée de DPPG. Comprend :

- Acteur, critique
- Target networks
- OU noise process

Manque Experience Replay. En tout cas, les résultats sont nuls. Ajouter Experience Replay et vérifier à nouveau.

8 GAIL : Generative Adversarial Imitation Learning

Permet d'apprendre une politique IRL-style sans s'encombrer de la fonction de récompense [2]. Semble, à priori, plus approprié que [1].

En pratique, les auteurs introduisent une fonction de régularisation ψ qui cherche à minimiser la distance entre ce qu'ils appellent **occupancy measure** de l'expert et celle de la politique en cours d'apprentissage. Cette notion représente la distribution des paires état-actions rencontrés suivant la politique.

On a :

$$\psi_{GA} = \begin{cases} \mathbb{E}_{\pi_E}[g(c(s,a))] \text{ si } c < 0 \\ +\infty \text{ sinon} \end{cases} \qquad where \quad g(x) = \begin{cases} -x - \log(1 - e^x) \text{ si } x < 0 \\ +\infty \text{ sinon} \end{cases} \tag{1}$$

J'ai découvert des approches plus récentes :

- Robust Imitation of Diverse Behaviours [5]
- Third Person Learning [4]
- Ainsi qu'une review : [3]

Références

[1] Finn, C., Levine, S., and Abbeel, P. Guided cost learning : Deep inverse optimal control via policy optimization. *CoRR abs/1603.00448* (2016).

[2] Ho, J., and Ermon, S. Generative adversarial imitation learning. *CoRR abs/1606.03476* (2016).

[3] Hussein, A., Gaber, M. M., Elyan, E., and Jayne, C. Imitation learning : A survey of learning methods. *ACM Comput. Surv. 50*, 2 (Apr. 2017), 21 :1–21 :35.

[4] Stadie, B. C., Abbeel, P., and Sutskever, I. Third-Person Imitation Learning. *ArXiv e-prints* (Mar. 2017).

[5] Wang, Z., Merel, J., Reed, S., Wayne, G., de Freitas, N., and Heess, N. Robust Imitation of Diverse Behaviors. *ArXiv e-prints* (July 2017).