Unsupervised Exploration and Offline Reinforcement Learning

SURF-2024-0183

Supervisor: Chunchuan Lyu
Students: Chengyang Du , Xu Chen , Zirui Zhu , Gong Chen , Ruobing Li , Jiutian Chang

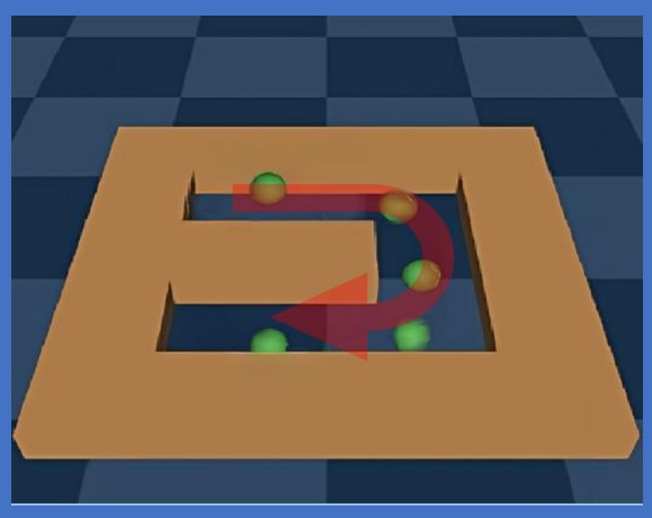
Abstract

Imagine teaching a robot to navigate a maze without providing step-by-step instructions or rewards for each correct move. This research project explores innovative ways to enable machines to learn and make decisions independently. By allowing the machine to explore its environment freely, it can gather valuable information and experiences without constant human guidance. Additionally, we will investigate methods that allow the machine to learn from previously collected data, reducing the need for continuous real-time interaction during the learning process. The goal is to develop intelligent systems that can explore effectively in complex environments, even when direct feedback is limited.

Introduction

We investigate the effectiveness of unsupervised exploration by comparing its performances under different settings. During the unsupervised exploration phase, agents explore the environment without task-specific rewards (no red ball), gathering trajectories for future offline learning. In the offline learning phase, agents learned from these trajectories with post-added rewards.

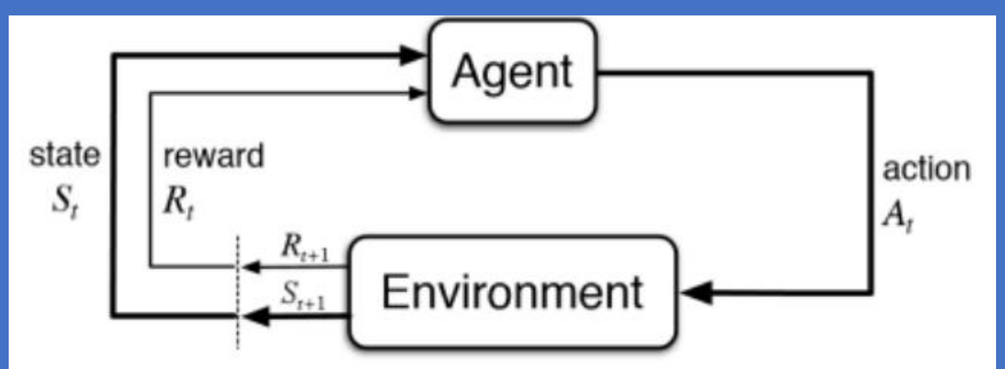
Three exploration strategies are investigated: Pure random exploration, Random Network Distillation (RND), and Proximal Policy Optimization (PPO). RND is employed to encourage visiting novel states while avoiding revisiting. PPO is used to set an upper bound on the effectiveness of unsupervised exploration. Implicit Q-Learning (IQL) and Hindsight Experience Replay (HER) are used for offline learning.



Unsupervised Exploration: green point explore the environment without knowing the goal

Supervised task: green point try catch red ball

Preliminaries



Random network distillation *Use exploration bonus to encourage agent to explore novel states*

$$i_t = \|\hat{f}(s_{t+1}) - f(s_{t+1})\|^2$$

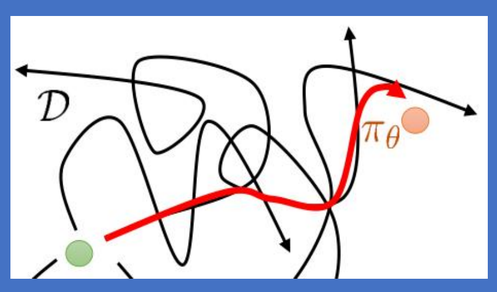
Proximal policy optimization *A highly data-efficient policy gradient method.*

$$L^{CLIP}(\theta) = \hat{\mathbb{E}}_t\left[\min(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)\right]$$

Hindsight Experience Replay *Use future achieved goals as target goals, we lookforward 16 steps*

Implicit Q-learning $\mathcal{L}_V(\theta_V) = \mathbb{E}_{(s,a)\sim\mathcal{D}}[L_2^\tau(Q_{\bar{\theta}_Q}(s,a) - V_{\theta_V}(s))]$

A conservative strategy $\tau \in [0.5, 1)$: $L_2^\tau(x) = |\tau - \mathbb{1}(x < 0)|x^2$

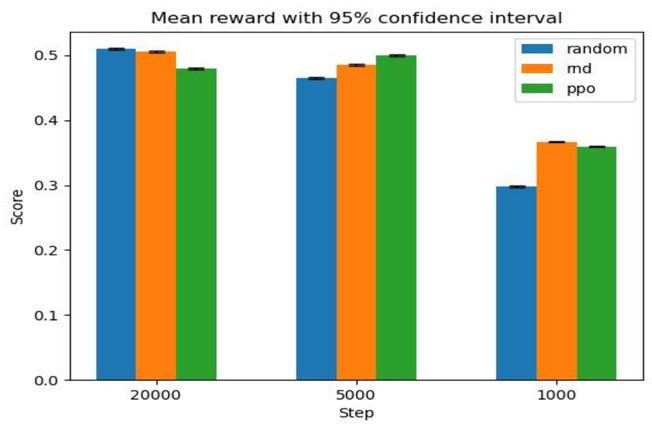


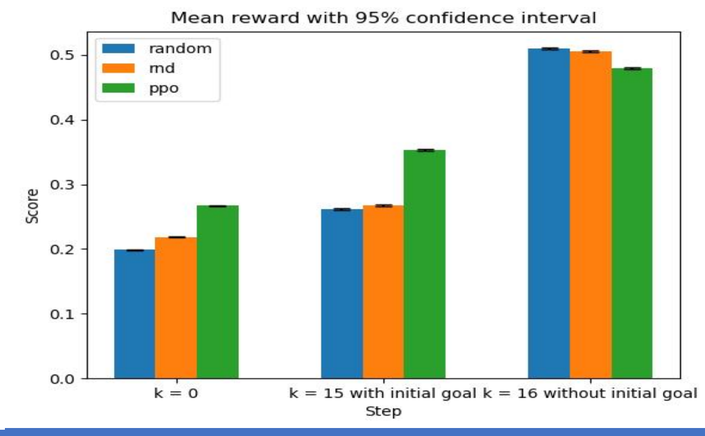
In offline reinforcement learning, the strategy is optimized on the existing data trajectory to find a more efficient path to reach the target state

Methods and Experimets

When using HER, we adjust the original trajectory with different goals. We designed two goal modes: one with k=0, where the agent is only informed of its sampled goal, and another with k=16, where next 16 states serve as achieved goals. We also experimented with different total timesteps during the exploration phase: 1000 steps, 5000 steps, and 20000 steps.

For the exploration agents, at the 20,000-step, the random agent achieved an average reward of 0.19, while the RND agent achieved an average reward of 0.23. The PPO agent achieved an average reward of 0.38. As an upper bound, the PPO agent could achieve a score of 0.62 after 100,000 timesteps of training.





- *For data collected through random and RND exploration, more data indeed helps the agents achieve convergence and better performance.*
- *At 1000 time steps exploration, random are lagging behind RND and PPO by a significant margin.*
- *The sampled goal hurts boost performance, even with achieved goals.*
- *Surprisingly, both Random and RND surpassed PPO at 20000-timesteps. The offline learning all beats the original PPO learned during exploration.*

Conclusions and Future Works

Overall, we have shown offline learning are very effective. However, we have not achieved significant gains over random exploration. This could be caused by our choice of too simplified environment or by our lack of hyperparameter tuning. In addition, we could explore the low data region in more depth.

In the future, we aim to devise novel algorithms for unsupervised exploration and evaluate them in broader settings.

Reference

[1] Andrychowicz, M., Wolski, F., Ray, A., Schneider, J., Fong, R., Welinder, P., ... & Zaremba, W. (2017). Hindsight experience replay. Advances in neural information processing systems, 30.
[2] Burda, Y., Edwards, H., Storkey, A., & Klimov, O. (2018). Exploration by random network distillation. arXiv preprint arXiv:1810.12894.
[3] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.
[4] Kostrikov, I., Nair, A., & Levine, S. (2021). Offline reinforcement learning with implicit q-learning. arXiv preprint arXiv:2110.06169.
[5] Laskin, M., Yarats, D., Liu, H., Lee, K., Zhan, A., Lu, K., ... & Abbeel, P. (2021). Urlb: Unsupervised reinforcement learning benchmark. arXiv preprint arXiv:2110.15191.