Generative Adversarial Imitation Learning(GAIL)

1. Train experts
    a. `python3 run_ppo.py`
    b. 전문가 행동을 만들기 위해서 (학습).
        i. Env-CartPole-V0
        ii. 2 개의 Network 생성 =(Policy, Old Policy)(action 을 예측함)
            1. Policy_net, Value_net 2 개 만들고, action stochastic(Policy_net 사용)하게 함
        iii. 2 개의 Network ->를 PPO_network 로 넣음(new, old)
        iv. PPO 는 2 개의 학습네트워크의 ratio of Probability 가 계산에 필요
            1. 기본 트레이닝 세팅
            2. Old, New -> action_probability 를 가져옴

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

- θ is the policy parameter
- $\hat{E}_t$ denotes the empirical expectation over timesteps
- r_t is the ratio of the probability under the new and old policies, respectively
- $\hat{A}_t$ is the estimated advantage at time t
- ε is a hyperparameter, usually 0.1 or 0.2

            3.
            4. 수행
            5. 각 Network 의 Policy -> entropy 계산 , Value_prob -> MSE 계산
            6. Training Algorithm
                a. Policy network (action from action_prob, value_prob, reward, state 를 에피소드가 끝날 때 까지 모음)
                b. (Reward + gamma * V_next - V) -> advantage term
                c. PPO training
            7. Action probability 는 학습은 ->PPO
2. Sample Trajectory 만들기 -> GAIL 를 수행하기 위해서 data 를 생성
    a. `python3 sample_trajectory.py`
        i. 기존에 PPO 에 학습된 agent 를 불러서 cartpole 를 환경에서 action 과 state 를 가지며 각각을 list 로 저장
        ii. Trajectory directory 에 저장
3. GAIL 학습
    a. Python3 run_gail 실행

π_E	Expert Policy
$H(\pi)$	γ-discounted casual entropy of policy π
c(s,a)	cost for state s and action a
ψ	Regularizer
ψ^*	Convex conjugate of ψ

i.
ii. Policy net (old, new) -> PPO -> Discriminator(env) -> train(PPO)
iii. Discriminator -> Learning agent(not cost)
iv. Expert 와 학습되는 Agent 의 state_action 을 가져옴

$$\mathrm{RL} \circ \mathrm{IRL}_{\psi}(\pi_E) = \arg\min_{\pi \in \Pi} -H(\pi) + \psi^*(\rho_\pi - \rho_{\pi_E})$$

1. 여기서 convex conjugate 부분을 Discriminator 학습네트워크로 표현

v. 각 Agent 를 네트워크를 만들고 Probability 를 가져옴 ->전문가의 A, S 와 현재 Agent 의 A, S 를 가져와서 전문가와 유사하게 샘플링함

vi. $$\psi^*_{\mathrm{GA}}(\rho_\pi - \rho_{\pi_E}) = \max_{D \in (0,1)^{\mathcal{S} \times \mathcal{A}}} \mathbb{E}_\pi[\log(D(s,a))] + \mathbb{E}_{\pi_E}[\log(1 - D(s,a))]$$

GAN term 을 Loss function 을 만듬

vii. Reward 를 Agent 의 reward 로 가짐
1. Because D(expert|a,s) = 1-D(agent|a,s)

viii. GAIL 로 부터 만들어진 Reward 를 가지고, PPO 의 advantage term(gaes)를 만든이후 PPO 학습