Reinforcement Learning & Advanced Deep

M2 DAC

TME 6. Advanced Policy Gradients

Ce TME a pour objectif d'expérimenter l'approche PPO.

1 PPO Adaptative KL

Implémenter l'algorithme PPO avec coût KL adaptatif donné dans la figure ci-dessous et l'appliquer aux 3 problèmes des TP précédents (CartPole, LunarLander et GridWorld)

Algorithm 4 PPO with Adaptive KL Penalty

Input: initial policy parameters θ_0, initial KL penalty β_0, target KL-divergence δ
for $k = 0, 1, 2, \ldots$ **do**
  Collect set of partial trajectories $\mathcal{D}_k$ on policy $\pi_k = \pi(\theta_k)$
  Estimate advantages $\hat{A}_t^{\pi_k}$ using any advantage estimation algorithm
  Compute policy update

$$\theta_{k+1} = \arg\max_{\theta} \mathcal{L}_{\theta_k}(\theta) - \beta_k \bar{D}_{KL}(\theta||\theta_k)$$

  by taking K steps of minibatch SGD (via Adam)
  if $\bar{D}_{KL}(\theta_{k+1}||\theta_k) \geq 1.5\delta$ **then**
    $\beta_{k+1} = 2\beta_k$
  else if $\bar{D}_{KL}(\theta_{k+1}||\theta_k) \leq \delta/1.5$ **then**
    $\beta_{k+1} = \beta_k/2$
  end if
end for

Notons les K pas de gradients à chaque optimisation sur les trajectoires récoltées (plutôt que 1 avec les PG classiques).

Au fait d'après vous, que vaut le gradient de la KL au premier passage ?

Quels sont les hyper-paramètres dont vous aurez besoin ?

2 PPO with Clipped Objective

Implémenter l'algorithme PPO avec objectif "clippé" donné dans la figure ci-dessous.

Algorithm 5 PPO with Clipped Objective

Input: initial policy parameters θ_0, clipping threshold ϵ
for $k = 0, 1, 2, \ldots$ **do**
Collect set of partial trajectories $\mathcal{D}_k$ on policy $\pi_k = \pi(\theta_k)$
Estimate advantages $\hat{A}_t^{\pi_k}$ using any advantage estimation algorithm
Compute policy update

$$\theta_{k+1} = \arg\max_{\theta} \mathcal{L}_{\theta_k}^{CLIP}(\theta)$$

by taking K steps of minibatch SGD (via Adam), where

$$\mathcal{L}_{\theta_k}^{CLIP}(\theta) = \mathop{\mathrm{E}}_{\tau \sim \pi_k} \left[\sum_{t=0}^{T} \left[\min(r_t(\theta)\hat{A}_t^{\pi_k}, \text{clip}\left(r_t(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}_t^{\pi_k}) \right] \right]$$

end for

3 Comparaison

Comparer les performances des deux versions de PPO avec A2C et une version de PPO sans KL ni clipped objective (au moins sur Cartpole et LunarLander). Pour cela, une fois des "bons" hyper-paramètres trouvés, faire tourner les algos un certain nombre de fois (e.g., 10) et tracer les points moyens de performance (e.g., toutes les 100 trajectoires collectées).

4 Bonus: Entropie

Vous pourrez considérer une version avec coût d'entropie évitant aux politiques de converger trop rapidement vers des solutions sous-optimales.