REINFORCE

Kihwan Lee

Contents

1. REINFORCE

REINFORCE

REINFORECE

- Q value값을 output으로 하는 DQN과 달리 > action값을 output으로 도출 : conti-action space
- Policy Gradient Thm을 통해 목적함수의 gradient를 계산하기 쉽게 표현 : $\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{\pi_\theta}[r(\tau)] = \mathbb{E}_{\pi_\theta}\left[r(\tau) \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t \mid s_t)\right]$
- mini-batch M개의 trajectory의 샘플을 찾은 후에 평균으로 대체 : MCMC Sampling
- Total reward r(τ)를 discounted return Gt로 대체하여 앞으로 얻을 reward만 고려

=> REINFORCE (Monte Carlo Policy Gradient)

Repeat (1) ~ (3)

(1) Execute M trajectories
(each starting in state s and executing (stochastic) policy π_θ)

(2) Approximate the gradient of the objective function $J(\theta)$

$$g_\theta := \frac{1}{M} \sum_{i=1}^{M} \left(\sum_{t=0}^{T-1} G_t^{(i)} \nabla_\theta \log \pi_\theta(a_t^{(i)} \mid s_t^{(i)}) \right) \approx \nabla_\theta J(\theta)$$

(3) Update policy (network parameters) to maximize $J(\theta)$

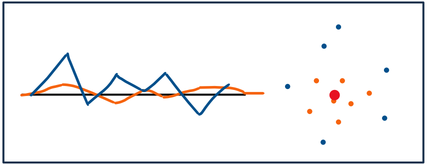
$$\theta := \theta + \alpha g_\theta \approx \theta + \alpha \nabla_\theta J(\theta)$$

2. REINFORCE with baseline

REINFORCE with baseline

- **목표**

  1) 안정적인 학습을 위한 variance를 줄이기
  2) 평균을 제대로 잡게 bias를 줄이기



- **Problem**

  Monte Carlo의 경우 data를 생성할 때, 즉 trajectory를 만들 때 (… St at rt+1 St+1 …) 모든 곳에서 무작위성이 발생 (ε-greedy)

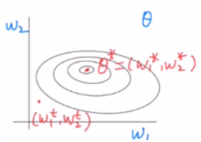
  => 이로 인한 variance가 너무 크다.

REINFORCE with baseline

REINFORECE with Baseline

앞에서 언급한 MC의 높은 variance 문제로 인해 기존 REINFORECE 목적함수에 baseline을 빼준다.

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{\pi_\theta}[r(\tau)] = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T-1} \underline{(G_t - b(s_t))}\ \underline{\nabla_\theta \log \pi_\theta(a_t \mid s_t)}\right]$$



* 고려해야 할 점 : b(st)를 빼줌으로 expectation이 바뀌어서는 안 된다. (bias 발생)

=> b(st) 은 action과 관계 없는 값이라면 다 적합!

$$\because\ \mathbb{E}_\pi\left[\sum_t b(s_t) \nabla \log \pi(a_t \mid s_t)\right] = \int \sum_t \pi(a_t \mid s_t)\, b(s_t)\, \nabla \log \pi(a_t \mid s_t)\, d\tau = \int \sum_t b(s_t)\, \nabla \pi(a_t \mid s_t)\, d\tau = 0$$

$$\text{since } \sum_{a_t} b(s_t) \nabla \pi(a_t \mid s_t) = b(s_t) \nabla \sum_{a_t} \pi(a_t \mid s_t) = b(s_t) \nabla 1 = 0.$$

* Good baseline : the state-value of current state V(s) : $V(s) = \mathbb{E}_{\pi_\theta}[G_t \mid S_t = s]$.

=> Keeping the gradient unbiased

REINFORCE with baseline

REINFORECE with Baseline

기존 REINFORCE에서는 policy에 대한 network만 존재

=> 이제는 baseline으로 사용되는 State value function과 관련한 network가 추가적으로 필요

Initialize state-value $V(s;\phi)$ and policy $\pi(a \mid s;\theta)$ randomly
Hyperparameters: stepsizes $\alpha > 0$, $\beta > 0$
for episode $= 1, M$ **do**
    Generate an episode $s_0, a_0, r_1, s_1, \ldots, s_{T-1}, a_{T-1}, r_T$, following $\pi(\cdot \mid \cdot;\theta)$
    for $t = 0, T-1$ **do**
        $G_t \leftarrow$ return from step t
        $\delta \leftarrow G_t - V(s_t;\phi)$
        $\phi \leftarrow \phi + \beta\delta\nabla_\phi V(s_t;\phi)$
        $\theta \leftarrow \theta + \alpha\gamma^t\delta\nabla_\theta \log\pi(a_t \mid s_t;\theta)$
    end
end

minimizing $L(\phi) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T-1}(G_t - V(s_t;\phi))^2\right]$

$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T-1}(G_t - V(s_t;\phi))\nabla_\theta \log\pi_\theta(a_t \mid s_t)\right]$