Lecture11: Policy Gradients for Reinforcement Learning

> Notes taken by squarezhong
> Repo address: squarezhong/SDM5008-Lecture-Notes

Lecture11: Policy Gradients for Reinforcement Learning
- Problem Formulation
- Derivation of Policy Gradient
  - Compute $\nabla_\theta U(\theta)$
  - Summary of Policy Gradient Derivation

Problem Formulation

- RL Problem: find optimal policy π^*

$$V^*(s) = \max_\pi (E_{\tau\sim\pi}(R(\tau)|S_0 = s))$$

- Parameterize policy by a parameter vector θ, denote:
  - $\pi_\theta := \pi_\theta(\cdot|s)$
  - $P_\theta(\tau)$: the likelihood of trajectory τ under policy π_θ

$$P_\theta(\tau) = P(s_0)\prod_{t=0}^{T-1}\pi_\theta(a_t|s_t)P(s_{t+1}|s_t, a_t)$$

- Reformulate the MDP problem as an optimization problem
  - assume s_0 distribution is included in trajectory likelihood $P_\theta(\tau)$
  - Utility function:

$$U(\theta) = E_{\tau\sim P_\theta(\tau)}[R(\tau)] = \sum_\tau P_\theta(\tau)R(\tau)$$

  - RL problem reduces to finding the optimal policy parameter θ

$$\max_\theta U(\theta) = \max_\theta \sum_\tau P_\theta(\tau)R(\tau)$$

    - θ: dim is high
    - $U(\theta)$: hard to compute (approximate)
  - Policy gradient use 1^{st} order gradient ascend method

Derivation of Policy Gradient

Optimize $U(\theta)$ involves evaluating $U(\theta)$

- think of τ as a random variable (in trajectory space)
- By monte carlo: $U(\theta) = \sum_\tau P_\theta(\tau)R(\tau) \approx \frac{1}{N}\sum_i R(\tau^{(i)}), \quad \tau^{(i)} \overset{\text{iid}}{\sim} P_\theta(\tau)$

Compute $\nabla_\theta U(\theta)$

$$\nabla_\theta U(\theta) = \frac{\partial}{\partial\theta}\sum_\tau P_\theta(\tau)R(\tau) = \sum_\tau R(\tau)\frac{\partial}{\partial\theta}P_\theta(\tau)$$

Note the fact that $\frac{\nabla_\theta P_\theta(\tau)}{P_\theta(\tau)} = \nabla_\theta \log(P_\theta(\tau))$

Then

$$\begin{aligned}\nabla_\theta U(\theta) &= \sum_\tau R(\tau)P_\theta(\tau)\frac{\nabla_\theta P_\theta(\tau)}{P_\theta(\tau)} = \sum_\tau R(\tau)P_\theta(\tau)\nabla_\theta \log(P_\theta(\tau)) \\ &= \underset{\tau\sim P_\theta(\tau)}{E}(R(\tau)\nabla_\theta \log(P_\theta(\tau))) \\ &\approx \frac{1}{N}\sum_\tau R(\tau^{(i)})\nabla_\theta \log(P_\theta(\tau^{(i)})) \quad \text{(monte carlo)}\end{aligned}$$

Substitute concrete expression

$$\begin{aligned}\nabla_\theta \log P_\theta(\tau) &= \nabla_\theta\left(\log P(s_0) + \sum_{t=0}^{T-1}\log \pi_\theta(a_t|s_t) + \log P(s_{t+1}|s_t, a_t)\right) \\ &= \nabla_\theta \sum_{t=0}^{T-1}\log \pi_\theta(a_t|s_t)\end{aligned}$$

Then

$$\begin{aligned}\nabla_\theta U(\theta) &= \underset{\tau\sim P_\theta(\tau)}{E}\left(R(\tau)\nabla_\theta \sum_{t=0}^{T-1}\log \pi_\theta(a_t|s_t)\right) \\ &= \frac{1}{N}\sum_i R(\tau^{(i)})\nabla_\theta \sum_{t=0}^{T-1}\log \pi_\theta(a_t^{(i)}|s_t^{(i)}) \quad \text{(monte carlo)}\end{aligned}$$

Summary of Policy Gradient Derivation

- Roll out trajectories $\tau^{(i)} \sim P_\theta(\cdot),\ i = 1, \cdots, N$
- Compute the empirical mean $\hat{g} = \frac{1}{N}\sum_i R(\tau^{(i)})\nabla_\theta \sum_{t=0}^{T-1}\log \pi_\theta(a_t^{(i)}|s_t^{(i)})$
- By Monte Carlo, we know $E(\hat{g}) = \nabla_\theta U(\theta)$
- In practice, the sample mean estimate $\hat{g}$ has a high variance and many ways can be used to reduce the variance, leading to different algorithms.