COMP6247: Reinforcement and Online Learning

Mahesan Niranjan

School of Electronics and Computer Science
University of Southampton

Week 12: Policy Gradients

Spring Semester 2020/21

Policy Gradients

- Policy search: Directly optimize policy π_θ, by a parameterized function approximator.
- Return $R(\tau)$ for trajectory τ: $s_0, a_0, r_1, s_1, a_1, r_2, ..., s_{T-1}, a_{T-1}, r_T$
- Maximize an objective function:

$$J(\theta) \;=\; E_{\tau \sim \rho(\theta)} \left[\gamma^t r(s_t, a_t, s_{t+1}) \right]$$

- Likelihood of trajectory:

$$\begin{aligned} \rho_\theta(\tau) &= p_\theta(s_0, a_0, r_1, s_1, a_1, r_2, ..., s_{T-1}, a_{T-1}, r_T) \\ &= p_0(s_0) \prod_{t=0}^{T} \pi_\theta(s_t | a_t)\, p(s_{t+1} | s_t, a_t) \end{aligned}$$

- Objective Function

$$J(\theta) \;=\; \int_\tau \rho_\theta(\tau) R(\tau) d\tau$$

Policy Gradient (cont'd)

- Monte Carlo Approximation

$$
\begin{aligned}
J(\theta) &= \int_\tau \rho_\theta(\tau) R(\tau) d\tau \\
&= \frac{1}{N} \sum_{i=1}^{N} R(\tau_i)
\end{aligned}
$$

- Sample trajectories according to their likelihood and average the returns
- Policy Gradient

$$\nabla_\theta J(\theta) = \frac{\partial J(\theta)}{\partial \theta}$$

- Update

$$\theta \leftarrow \theta - \eta \nabla_\theta J(\theta)$$

Policy Gradient (cont'd)

- Derivative of objective function w.r.t. policy parameters θ

$$
\begin{aligned}
\nabla_\theta J(\theta) &= \nabla_\theta \int \rho_\theta(\tau) R(\tau) d\tau \\
&= \int_\tau \left(\nabla_\theta \rho_\theta(\tau)\right) R(\tau) d\tau
\end{aligned}
$$

- A trick: $\frac{d \log f(x)}{dx} = \frac{f'(x)}{f(x)}$
- Policy gradient along single trajectory:

$$\nabla_\theta \int \rho_\theta(\tau) = \rho_\theta(\tau) \nabla_\theta \log \rho_\theta(\tau)$$

- We now have

$$
\begin{aligned}
\nabla_\theta J(\theta) &= \int_\tau \rho_\theta(\tau) \nabla_\theta \log \rho_\theta(\tau) R(\tau) d\tau \\
&= E_{\tau \sim \rho(\tau)} \left[\log \rho_\theta(\tau) R(\tau)\right]
\end{aligned}
$$

REINFORCE

- Expand further...

$$\log \rho_t heta(\tau) \;=\; \log p_0(s_0) \;+\; \sum_{t=0}^{T} \log \pi_\theta(s_t, a_t) \;+\; \sum t = 0^T \log p(s_{t+1}|s_t, a_t)$$

- Two of the above terms do not depend on θ

$$\nabla_\theta \log \rho_\theta(\tau) = \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(s_t, a_t)$$

- Gradient of interest is independent of dynamics of MDP, leading to REINFORCE :
  - Sample N trajectories from π_θ; observe $R(\tau_i)$
  - Gradient update

$$\begin{aligned} \nabla_\theta J(\theta) &= \frac{1}{N} \sum_{i=1}^{N} \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(s_t, a_t) R(\tau_i) \\ \theta &\leftarrow \theta - \nabla_\theta J(\theta) \end{aligned}$$

What we have not covered!

- Algorithmic issues with REINFORCE
  - Variance reduction – baseline
  - Policy gradient theorem
  - Estimating gradient from n lookahead/ bootstrap
- Actor-Critic methods
  - Parameterized policy as actor: $\pi_\theta(a_t|s_t)$
  - Value function approximator as critic: $v_\alpha(s_t, a_t)$