CS 594 Modern Reinforcement Learning

Lecture 4: Actor-Critic Methods

REINFORCE

- $\nabla J(\theta) = \sum_s \eta(s) \sum_a \pi(a|s,\theta) q_\pi(s,a) \nabla \ln(\pi(a|s,\theta))$

- Sample a trajectory
- For each time t calculate the Monte Carlo Estimate:
  - $g_t = \sum_{k=t}^{T-1} \gamma^{k-t} r_{t+1}$
- Calculate gradient estimate:
  - $\widehat{\nabla J(\theta)} = \sum_{t=0}^{T} \gamma^t \mathrm{g_t} \nabla \ln(\pi(a_t|s_t,\theta))$
- Update:
  - $\theta_{t+1} = \theta_t + \alpha \nabla \widehat{J(\theta_t)}$

Baselines

- $\nabla J(\theta) = \sum_s \eta(s) \sum_a \big(\nabla\pi(a|s,\theta)\big) q_\pi(s,a)$
- $\sum_a \big(\nabla\pi(a|s,\theta)\big)$
- $\sum_a \big(\nabla\pi(a|s,\theta)\big)=0$
- $\nabla J(\theta) = \sum_s \eta(s) \sum_a \big(\nabla\pi(a|s,\theta)\big)(q_\pi(s,a) - b(s))$

Which Baseline?

- $Var(X) = E[X^2] - (E[X])^2$
- P(X=+1)=0.5, P(X=0)=0.5
- How to minimize $Var\big(X - B(X)\big)$?
- How to minimize $Var(Q(S,A) - B(S))$?
- Common choice: $b(s) = \hat{v}(s)$

Which Baseline?

- Common choice: $b(s) = \hat{v}(s)$

- $Var(X - Y) = Var(X) + Var(Y) - Cov(X, Y)$
- $Cov(X, Y) = E[(X - E[X])(Y - E[Y])] = E[XY] - E[X]E[Y]$

- Want Y to be (positively) correlated with X

- Simple and works well, but not optimal

Advantage Function

- $\nabla J(\theta) = \sum_s \eta(s) \sum_a \big(\nabla \pi(a|s,\theta)\big)\big(q_\pi(s,a) - v_\pi(s)\big)$

- $adv_\pi(s,a) = q_\pi(s,a) - v_\pi(s)$

- Lots of nice properties / intuitions / uses
  - Want our gradients to make actions with positive advantages more likely
  - Policy improvement steps look for positive advantages
  - Form similar to TD-error

REINFORCE with Baseline

- $\nabla J(\theta) = \sum_s \eta(s) \sum_a \pi(a|s,\theta)(q_\pi(s,a) - \hat{v}(s)) \nabla \ln(\pi(a|s,\theta))$

- Sample a trajectory
- For each time t calculate the Monte Carlo Estimate with Baseline:
  - $\delta_t = \left(\sum_{k=t}^{T-1} \gamma^{k-t} r_{t+1}\right) - \hat{v}(s)$
- Calculate gradient estimate:
  - $\widehat{\nabla J(\theta)} = \sum_{t=0}^{T} \gamma^t \, \delta_\mathrm{t} \nabla \ln(\pi(a_t|s_t,\theta))$
- Update:
  - $\theta_{t+1} = \theta_t + \alpha \nabla \widehat{J(\theta_t)}$

Gradients for Value Functions

- $VE(W) = \sum_s \mu(s)[v_\pi(s) - \hat{v}(s,w)]^2$
- $\nabla VE(W) = \sum_s \mu(s) 2[v_\pi(s) - \hat{v}(s,w)][-\nabla \hat{v}(s,w)]$
- $w_{t+1} = w_t + \alpha_t[U_t - \hat{v}(s_t, w_t)] \nabla \hat{v}(s_t, w_t)$
- $w_{t+1} = w_t + \alpha_t[g_t - \hat{v}(s_t, w_t)] \nabla \hat{v}(s_t, w_t)$
- $w_{t+1} = w_t + \alpha_t[r_{t+1} + \gamma\, \hat{v}(s_{t+1}, w_t) - \hat{v}(s_t, w_t)] \nabla \hat{v}(s_t, w_t)$

REINFORCE with Baseline

- Sample a trajectory
- For each time t calculate the Monte Carlo Estimate with Baseline:
  - $\delta_t = \left(\sum_{k=t}^{T-1} \gamma^{k-t} r_{t+1}\right) - \hat{v}(s)$
- Calculate gradient estimates:
  - $\widehat{\nabla J(\theta)} = \sum_{t=0}^{T} \gamma^t \, \delta_{\mathrm{t}} \nabla \ln(\pi(a_t|s_t, \theta)$
  - $\widehat{\nabla V(w)} = \sum_{t=0}^{T} \gamma^t \, \delta_{\mathrm{t}} \nabla \hat{v}(s, w)$
- Update:
  - $\theta_{t+1} = \theta_t + \alpha^\theta \widehat{\nabla J(\theta_t)}$
  - $w_{t+1} = w_t + \alpha^w \widehat{\nabla V(w_t)}$

Use of the Baseline

- Value function baseline reduces variance of Monte Carlo estimates in REINFORCE

- How else have we reduced the variance of Monte Carlo?

- Actor-Critic: we can use the value function to put in TD estimates

One-step Actor Critic

While not terminal:

- Sample a from $\pi(a, \theta)$
- Take a. Observe s', r
- $\delta_t = r + \gamma\hat{v}(s', w) - \hat{v}(s, w)$
- $w_{t+1} = w_t + \alpha^w \delta_\mathrm{t} \nabla \hat{v}(s, w)$
- $\theta_{t+1} = \theta_t + \alpha^\theta \gamma^t \delta_\mathrm{t} \nabla \ln(\pi(a|s, \theta)$

Summary

- Can learn everything via stochastic gradient methods
  - Value functions
  - Parameterized Policies
    - Even handles continuous action spaces
- Two flavors
  - Monte Carlo (Policy Gradient Methods)
  - TD (Actor Critic Methods)
- Managing Variance – Baselines