Policy Gradient Methods: Estimation

Lucas Janson and Sham Kakade

CS/Stat 184: Introduction to Reinforcement Learning

Fall 2023

Today

- ✓ Recap
- Estimation: REINFORCE
- Variance Reduction
  - Other Gradient Expressions
  - Baselines and Advantages
- Examples

Recap

Optimization Objective

- Consider a parameterize class of policies:

$$\{\pi_\theta(a \mid s) \mid \theta \in \mathbb{R}^d\}$$

(why do we make it stochastic?)

- Objective $\max_\theta J(\theta)$, where

$$J(\theta) := E_{s_0 \sim \mu}\left[V^{\pi_\theta}(s_0)\right] = E_{\tau \sim \rho_{\pi_\theta}}\left[\sum_{h=0}^{H-1} r(s_h, a_h)\right]$$

- Policy Gradient Descent:

$$\theta^{k+1} = \theta^k + \eta \nabla J(\theta^k)$$

Example Policy Parameterizations

Recall that we consider parameterized policy $\pi_\theta(\cdot \mid s) \in \Delta(A), \forall s$

1. Softmax linear Policy

Feature vector $\phi(s, a, h) \in \mathbb{R}^d$, and parameter $\theta \in \mathbb{R}^d$

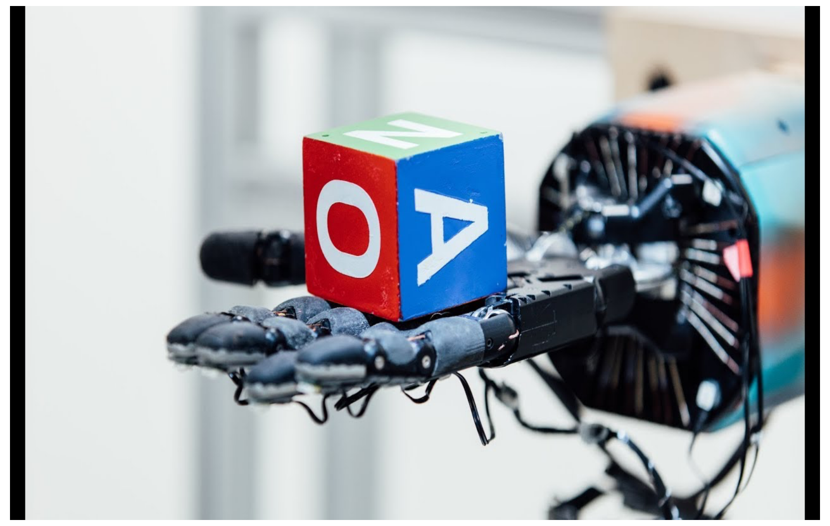
$$\pi_\theta(a \mid s, h) = \frac{\exp(\theta^\top \phi(s, a, h))}{\sum_{a'} \exp(\theta^\top \phi(s, a', h))}$$

2. Neural Policy:

Neural network $f_\theta : S \times A \times [H] \mapsto \mathbb{R}$

$$\pi_\theta(a \mid s, h) = \frac{\exp(f_\theta(s, a, h))}{\sum_{a'} \exp(f_\theta(s, a', h))}$$

Neural Policy Parameterization for “Controls”



Suppose $a \in R^k$, as it might be for a control problem.

3. Gaussian + Linear Model

- Feature vector: $\phi(s,h) \in \mathbb{R}^d$,
- Parameters: $\theta \in \mathbb{R}^{k \times d}$,
  (and maybe $\sigma \in R^+$)

- Policy: sample action from a (multivariate) Normal with mean $\theta \cdot \phi(s,h)$ and variance $\sigma^2 I$, i.e.
$$\pi_{\theta,\sigma}(\cdot \mid s,h) = \mathcal{N}\left(\theta \cdot \phi(s,h), \sigma^2 I\right)$$

4. Gaussian + Neural Model

- Neural network $g_\theta : S \times [H] \mapsto \mathbb{R}^k$
- Parameters: $\theta \in R^d$,
  (and maybe $\sigma \in R^+$)

- Policy: a (multivariate) Normal with mean $g_\theta(s)$ and variance $\sigma^2 I$, i.e.
$$\pi_{\theta,\sigma}(\cdot \mid s,h) = \mathcal{N}(g_\theta(s,h), \sigma^2 I)$$

The Likelihood Ratio Method

The Likelihood Ratio Method

- Suppose $J(\theta) = \mathbb{E}_{x \sim P_\theta}\left[f(x)\right] = \sum_x P_\theta(x) f(x)$, and our objective is $\max_\theta J(\theta)$.
- Computing $\nabla_\theta J(\theta)$ exactly may be difficult to compute (due to the sum over x).
  - Can we estimate $\nabla_\theta J(\theta)$?
  - Suppose we can: compute $f(x)$, $P_\theta(x)$, and $\nabla P_\theta(x)$ & sample $x \sim P_\theta$

sample $x_1, \ldots, x_m \sim P_\theta$

$\hat{J} = \frac{1}{m} \sum_{i=1}^{m} f(x_i)$

$G(\theta) = \mathbb{E}_{x \sim D}\left[g_\theta(x)\right]$

$x_1, \ldots, x_m \sim D$

$\hat{G}(\theta) = \frac{1}{m} \sum_{i=1}^{m} g_\theta(x_i)$

$\nabla \hat{G}(\theta) = \frac{1}{m} \sum_{i=1}^{m} \nabla g_\theta(x_i)$

The Likelihood Ratio Method

- Suppose $J(\theta) = \mathbb{E}_{x \sim P_\theta}\left[f(x)\right] = \sum_x P_\theta(x) f(x)$, and our objective is $\max_\theta J(\theta)$.
- Computing $\nabla_\theta J(\theta)$ exactly may be difficult to compute (due to the sum over x).
  - Can we estimate $\nabla_\theta J(\theta)$?
  - Suppose we can: compute $f(x)$, $P_\theta(x)$, and $\nabla P_\theta(x)$ & sample $x \sim P_\theta$
- We have that:

$$\nabla_\theta J(\theta) = \mathbb{E}_{x \sim P_\theta(x)}\left[\nabla_\theta \log P_\theta(x)\, f(x)\right]$$

The Likelihood Ratio Method

- Suppose $J(\theta) = \mathbb{E}_{x \sim P_\theta}\left[f(x)\right] = \sum_x P_\theta(x) f(x)$, and our objective is $\max_\theta J(\theta)$.
- Computing $\nabla_\theta J(\theta)$ exactly may be difficult to compute (due to the sum over x).
  - Can we estimate $\nabla_\theta J(\theta)$?
  - Suppose we can: compute $f(x)$, $P_\theta(x)$, and $\nabla P_\theta(x)$ & sample $x \sim P_\theta$
- We have that:

$$\nabla_\theta J(\theta) = \mathbb{E}_{x \sim P_\theta(x)}\left[\nabla_\theta \log P_\theta(x)\, f(x)\right]$$

Proof:

$$\begin{aligned}
\nabla_\theta J(\theta) &= \sum_x \nabla_\theta P_\theta(x) f(x) \\
&= \sum_x P_\theta(x) \frac{\nabla_\theta P_\theta(x)}{P_\theta(x)} f(x) \\
&= \sum_x P_\theta(x) \nabla_\theta \log P_\theta(x) f(x)
\end{aligned}$$

The Likelihood Ratio Method, continued

- We have:

$$\nabla_\theta J(\theta) = \mathbb{E}_{x\sim P_\theta(x)}\left[\nabla_\theta \log P_\theta(x)\, f(x)\right]$$

- An unbiased estimate is given by:

$\widehat{\nabla}_\theta J(\theta) = \nabla_\theta \log P_\theta(x) \cdot f(x)$, where $x \sim P_\theta$

- We can lower variance by draw N i.i.d. samples from P_θ and averaging:

$$\widehat{\nabla}_\theta J(\theta) = \frac{1}{N}\sum_{i=1}^{N} \nabla_\theta \log P_\theta(x_i)\, f(x_i)$$

Today:

Today

- Recap
- Estimation: REINFORCE
- Variance Reduction
  - Other Gradient Expressions
  - Baselines and Advantages
- Examples

REINFORCE: A Policy Gradient Algorithm

REINFORCE: A Policy Gradient Algorithm

- Let $\rho_\theta(\tau)$ be the probability of a trajectory $\tau = \{s_0, a_0, s_1, a_1, \ldots, s_{H-1}, a_{H-1}\}$, i.e.

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)P(s_1 \mid s_0, a_0)\ldots P(s_{H-1} \mid s_{H-2}, a_{H-2})\pi_\theta(a_{H-1} \mid s_{H-1})$$

REINFORCE: A Policy Gradient Algorithm

- Let $\rho_\theta(\tau)$ be the probability of a trajectory $\tau = \{s_0, a_0, s_1, a_1, \ldots, s_{H-1}, a_{H-1}\}$, i.e.
$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)P(s_1 \mid s_0, a_0)\ldots P(s_{H-1} \mid s_{H-2}, a_{H-2})\pi_\theta(a_{H-1} \mid s_{H-1})$$

- Let $R(\tau)$ be the cumulative reward on trajectory τ, i.e. $R(\tau) := \sum_{h=0}^{H-1} r(s_h, a_h)$

REINFORCE: A Policy Gradient Algorithm

- Let $\rho_\theta(\tau)$ be the probability of a trajectory $\tau = \{s_0, a_0, s_1, a_1, \ldots, s_{H-1}, a_{H-1}\}$, i.e.

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)P(s_1 \mid s_0, a_0)\ldots P(s_{H-1} \mid s_{H-2}, a_{H-2})\pi_\theta(a_{H-1} \mid s_{H-1})$$

- Let $R(\tau)$ be the cumulative reward on trajectory τ, i.e. $R(\tau) := \sum_{h=0}^{H-1} r(s_h, a_h)$

- Our objective function is:

$$J(\theta) = E_{\tau\sim\rho_\theta}\left[R(\tau)\right]$$

REINFORCE: A Policy Gradient Algorithm

- Let $\rho_\theta(\tau)$ be the probability of a trajectory $\tau = \{s_0, a_0, s_1, a_1, \ldots, s_{H-1}, a_{H-1}\}$, i.e.

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)P(s_1 \mid s_0, a_0)\ldots P(s_{H-1} \mid s_{H-2}, a_{H-2})\pi_\theta(a_{H-1} \mid s_{H-1})$$

- Let $R(\tau)$ be the cumulative reward on trajectory τ, i.e. $R(\tau) := \sum_{h=0}^{H-1} r(s_h, a_h)$
- Our objective function is:

$$J(\theta) \ = E_{\tau\sim\rho_\theta}\Big[R(\tau)\Big]$$

- The REINFORCE Policy Gradient expression:

$$\nabla_\theta J(\theta) := \mathbb{E}_{\tau\sim\rho_\theta}\left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h)\right) R(\tau)\right]$$

Proof

Proof

- From the likelihood ratio method, we have:

$$\nabla_\theta J(\theta) := \mathbb{E}_{\tau \sim \rho_\theta}\left[\nabla_\theta \ln \rho_\theta(\tau)\ R(\tau)\right]$$

Proof

- From the likelihood ratio method, we have:

$$\nabla_\theta J(\theta) := \mathbb{E}_{\tau \sim \rho_\theta} \left[\nabla_\theta \ln \rho_\theta(\tau) \; R(\tau) \right]$$

- We have:

$$\nabla_\theta \ln \rho_\theta(\tau) = \nabla_\theta \left(\ln \rho(s_0) + \ln \pi_\theta(a_0 \,|\, s_0) + \ln P(s_1 \,|\, s_0, a_0) + \ldots \right)$$

Proof

- From the likelihood ratio method, we have:

$$\nabla_\theta J(\theta) := \mathbb{E}_{\tau \sim \rho_\theta} \left[\nabla_\theta \ln \rho_\theta(\tau) \; R(\tau) \right]$$

- We have:

$$\nabla_\theta \ln \rho_\theta(\tau) = \nabla_\theta \Big(\ln \rho(s_0) + \ln \pi_\theta(a_0 \,|\, s_0) + \ln P(s_1 \,|\, s_0, a_0) + \ldots \Big)$$

$$= \nabla_\theta \Big(\ln \pi_\theta(a_0 \,|\, s_0) + \ln \pi_\theta(a_1 \,|\, s_1) \ldots \Big)$$

Proof

- From the likelihood ratio method, we have:

$$\nabla_\theta J(\theta) := \mathbb{E}_{\tau \sim \rho_\theta}\left[\nabla_\theta \ln \rho_\theta(\tau)\ R(\tau)\right]$$

- We have:

$$\nabla_\theta \ln \rho_\theta(\tau) = \nabla_\theta \Big(\ln \rho(s_0) + \ln \pi_\theta(a_0 \,|\, s_0) + \ln P(s_1 \,|\, s_0, a_0) + \ldots\Big)$$

$$= \nabla_\theta \Big(\ln \pi_\theta(a_0 \,|\, s_0) + \ln \pi_\theta(a_1 \,|\, s_1) \ldots\Big)$$

$$= \left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \,|\, s_h)\right)$$

Obtaining an Unbiased Gradient Estimate at θ_0

$$\nabla_\theta J(\theta) := \mathbb{E}_{\tau \sim \rho_\theta}\left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h)\right) R(\tau)\right]$$

Obtaining an Unbiased Gradient Estimate at θ_0

$$\nabla_\theta J(\theta) := \mathbb{E}_{\tau \sim \rho_\theta}\left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h)\right) R(\tau)\right]$$

1. Obtain a trajectory $\tau \sim \rho_{\theta_0}$
   (which we can do in our learning setting)

Obtaining an Unbiased Gradient Estimate at θ_0

$$\nabla_\theta J(\theta) := \mathbb{E}_{\tau \sim \rho_\theta}\left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h)\right) R(\tau)\right]$$

1. Obtain a trajectory $\tau \sim \rho_{\theta_0}$
   (which we can do in our learning setting)
2. Set:

$$\widetilde{\nabla}_\theta J(\theta_0) := \left(\sum_{h=0}^{H-1} \nabla \ln \pi_{\theta_0}(a_h \mid s_h)\right) R(\tau)$$

Obtaining an Unbiased Gradient Estimate at θ_0

$$\nabla_\theta J(\theta) := \mathbb{E}_{\tau \sim \rho_\theta}\left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h)\right) R(\tau)\right]$$

1. Obtain a trajectory $\tau \sim \rho_{\theta_0}$
   (which we can do in our learning setting)
2. Set:

$$\widetilde{\nabla}_\theta J(\theta_0) := \left(\sum_{h=0}^{H-1} \nabla \ln \pi_{\theta_0}(a_h \mid s_h)\right) R(\tau)$$

We have: $\mathbb{E}[\widetilde{\nabla}_\theta J(\theta_0)] = \nabla_\theta J(\pi_{\theta_0})$

PG with REINFORCE:

PG with REINFORCE:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$

PG with REINFORCE:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :

PG with REINFORCE:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
   1. Obtain a trajectory $\tau \sim \rho_{\theta^k}$

      Set $\widetilde{\nabla}_\theta J(\theta^k) = \sum_{h=0}^{H-1} \nabla \ln \pi_{\theta^k}(a_h | s_h) R(\tau)$

PG with REINFORCE:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
    1. Obtain a trajectory $\tau \sim \rho_{\theta^k}$

       Set $\widetilde{\nabla}_\theta J(\theta^k) = \sum_{h=0}^{H-1} \nabla \ln \pi_{\theta^k}(a_h | s_h) R(\tau)$

    2. Update: $\theta^{k+1} = \theta^k + \eta^k \widetilde{\nabla}_\theta J(\theta^k)$

The (mini-batch) PG procedure with REINFORCE

(reducing variance using batch sizes of M)

The (mini-batch) PG procedure with REINFORCE

(reducing variance using batch sizes of M)

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$

The (mini-batch) PG procedure with REINFORCE

(reducing variance using batch sizes of M)

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :

The (mini-batch) PG procedure with REINFORCE

(reducing variance using batch sizes of M)

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
    1. Init $g = 0$ and do M times:

       Obtain a trajectory $\tau \sim \rho_{\theta^k}$

       Update: $g \leftarrow g + \sum_{h=0}^{H-1} \nabla \ln \pi_{\theta^k}(a_h | s_h) R(\tau)$

The (mini-batch) PG procedure with REINFORCE

(reducing variance using batch sizes of M)

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
   1. Init $g = 0$ and do M times:

      Obtain a trajectory $\tau \sim \rho_{\theta^k}$

      Update: $g \leftarrow g + \sum_{h=0}^{H-1} \nabla \ln \pi_{\theta^k}(a_h | s_h) R(\tau)$
   2. Set $\widetilde{\nabla}_\theta J(\theta^k) := \frac{1}{M} g$

The (mini-batch) PG procedure with REINFORCE

(reducing variance using batch sizes of M)

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
   1. Init $g = 0$ and do M times:

      Obtain a trajectory $\tau \sim \rho_{\theta^k}$

      Update: $g \leftarrow g + \sum_{h=0}^{H-1} \nabla \ln \pi_{\theta^k}(a_h | s_h) R(\tau)$
   2. Set $\widetilde{\nabla}_\theta J(\theta^k) := \frac{1}{M} g$
   3. Update: $\theta^{k+1} = \theta^k + \eta^k \widetilde{\nabla}_\theta J(\theta^k)$

The (mini-batch) PG procedure with REINFORCE

(reducing variance using batch sizes of M)

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
   1. Init $g = 0$ and do M times:

      Obtain a trajectory $\tau \sim \rho_{\theta^k}$

      Update: $g \leftarrow g + \sum_{h=0}^{H-1} \nabla \ln \pi_{\theta^k}(a_h | s_h) R(\tau)$
   2. Set $\widetilde{\nabla}_\theta J(\theta^k) := \frac{1}{M} g$
   3. Update: $\theta^{k+1} = \theta^k + \eta^k \widetilde{\nabla}_\theta J(\theta^k)$

We sill have: $\mathbb{E}[\widetilde{\nabla}_\theta J(\theta^k)] = \nabla_\theta J(\theta^k)$

Today

- Recap
- Estimation: REINFORCE
- Variance Reduction
  - Other Gradient Expressions
  - Baselines and Advantages
- Examples

Other PG formulas
(that are lower variance for sampling)

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta} \left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) \right) R(\tau) \right]$$

Other PG formulas
(that are lower variance for sampling)

$$\nabla J(\theta) = \mathbb{E}_{\tau\sim\rho_\theta}\left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h)\right) R(\tau)\right]$$

$$= \mathbb{E}_{\tau\sim\rho_\theta}\left[\sum_{h=0}^{H-1}\left(\nabla_\theta \ln \pi_\theta(a_h \mid s_h) \sum_{t=h}^{H-1} r_t\right)\right]$$

Other PG formulas
(that are lower variance for sampling)

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta}\left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h)\right) R(\tau)\right]$$

$$= \mathbb{E}_{\tau \sim \rho_\theta}\left[\sum_{h=0}^{H-1}\left(\nabla_\theta \ln \pi_\theta(a_h \mid s_h) \sum_{t=h}^{H-1} r_t\right)\right]$$

$$= \mathbb{E}_{\tau \sim \rho_\theta}\left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) Q_h^{\pi_\theta}(s_h, a_h)\right]$$

Other PG formulas
(that are lower variance for sampling)

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta} \left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) \right) R(\tau) \right]$$

$$= \mathbb{E}_{\tau \sim \rho_\theta} \left[\sum_{h=0}^{H-1} \left(\nabla_\theta \ln \pi_\theta(a_h \mid s_h) \sum_{t=h}^{H-1} r_t \right) \right]$$

$$= \mathbb{E}_{\tau \sim \rho_\theta} \left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) Q_h^{\pi_\theta}(s_h, a_h) \right]$$

Intuition: Change action distribution at h only affects rewards later on…

HW: You will show these simplified version are also valid PG expressions

An improved PG procedure:

An improved PG procedure:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$

An improved PG procedure:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :

An improved PG procedure:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
    1. Obtain a trajectory $\tau \sim \rho_{\theta^k}$

       Set $\widetilde{\nabla}_\theta J(\theta^k) = \sum_{h=0}^{H-1} \left(\nabla \ln \pi_{\theta^k}(a_h | s_h) R_h(\tau) \right)$

On a trajectory τ, define:

$$R_h(\tau) = \sum_{t=h}^{H-1} r_t$$

An improved PG procedure:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
    1. Obtain a trajectory $\tau \sim \rho_{\theta^k}$

       Set $\widetilde{\nabla}_\theta J(\theta^k) = \sum_{h=0}^{H-1} \Big(\nabla \ln \pi_{\theta^k}(a_h \,|\, s_h) R_h(\tau) \Big)$
    2. Update: $\theta^{k+1} = \theta^k + \eta^k \widetilde{\nabla}_\theta J(\theta^k)$

On a trajectory τ, define:

$$R_h(\tau) = \sum_{t=h}^{H-1} r_t$$

An improved PG procedure:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
   1. Obtain a trajectory $\tau \sim \rho_{\theta^k}$

      Set $\widetilde{\nabla}_\theta J(\theta^k) = \sum_{h=0}^{H-1} \left(\nabla \ln \pi_{\theta^k}(a_h \mid s_h) R_h(\tau) \right)$
   2. Update: $\theta^{k+1} = \theta^k + \eta^k \widetilde{\nabla}_\theta J(\theta^k)$

On a trajectory τ, define:

$$R_h(\tau) = \sum_{t=h}^{H-1} r_t$$

Comments:

- We still have unbiased gradient estimates.
- Easy to use a mini-batch algorithm to reduce variance.
- Easy to compute the gradient in "one pass" over the data.

Today

- Recap
- Estimation: REINFORCE
- Variance Reduction
  - Other Gradient Expressions
  - Baselines and Advantages
- Examples

With a “baseline” function:

With a "baseline" function:

For any function only of the state, $b_h : S \to R$, we have:

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta}\left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h | s_h)\left(Q_h^{\pi_\theta}(s_h, a_h) - b_h(s_h)\right)\right]$$

$$= \mathbb{E}_{\tau \sim \rho_\theta}\left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h | s_h)\left(\sum_{t=h}^{H-1} r_t - b_h(s_h)\right)\right]$$

With a "baseline" function:

For any function only of the state, $b_h : S \to R$, we have:

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta} \left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h | s_h) \left(Q_h^{\pi_\theta}(s_h, a_h) - b_h(s_h) \right) \right]$$

$$= \mathbb{E}_{\tau \sim \rho_\theta} \left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h | s_h) \left(\sum_{t=h}^{H-1} r_t - b_h(s_h) \right) \right]$$

$E[x] = \mu$

$x \sim D$

$E[\mu - x] = 0$

$\hat{\mu} = \frac{1}{m} \sum_{i=1}^{m} x_i$

$\delta_i = \mu - x_i$

$\hat{\mu} = \frac{1}{m}$

This is (basically) the method of control variates.

$\hat{\mu} = \frac{1}{m} \sum_{i=1}^{m} (x_i + \delta_i) = \mu$

Proof:

Proof:

- We have:

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta}\left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) Q_h^{\pi_\theta}(s_h, a_h)\right]$$

$$= \sum_{h=0}^{H-1} \mathbb{E}_{s_h \sim \rho_\theta}\left[\mathbb{E}_{a_h \sim \pi(\cdot \mid s_h)}\left[\nabla_\theta \ln \pi_\theta(a_h \mid s_h) Q_h^{\pi_\theta}(s_h, a_h)\right]\right]$$

(where $s_h \sim \rho_\theta$ is a sample from the marginal state distribution at time h)

Proof:

- We have:

$$\nabla J(\theta) = \mathbb{E}_{\tau\sim\rho_\theta}\left[\sum_{h=0}^{H-1}\nabla_\theta \ln \pi_\theta(a_h\,|\,s_h)Q_h^{\pi_\theta}(s_h,a_h)\right]$$

$$= \sum_{h=0}^{H-1}\mathbb{E}_{s_h\sim\rho_\theta}\left[\mathbb{E}_{a_h\sim\pi(\cdot|s_h)}\left[\nabla_\theta \ln \pi_\theta(a_h\,|\,s_h)Q_h^{\pi_\theta}(s_h,a_h)\right]\right]$$

(where $s_h \sim \rho_\theta$ is a sample from the marginal state distribution at time h)

- To see this, first note:

$$\mathbb{E}_{x\sim P_\theta}\left[\nabla \log P_\theta(x) c\right] = c\sum_x P_\theta(x)\nabla \log P_\theta(x)$$

$$\nabla \log P_\theta(x) = \frac{\nabla P_\theta}{P_\theta}$$

$$= c\sum_x \nabla P_\theta(x) = c\nabla\sum_x P_\theta(x)$$

$$\sum_x P_\theta(x) = 1$$

$$= c\nabla 1 = 0$$

Proof:

- We have:

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta}\left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h | s_h) Q_h^{\pi_\theta}(s_h, a_h)\right]$$

$$= \sum_{h=0}^{H-1} \mathbb{E}_{s_h \sim \rho_\theta}\left[\mathbb{E}_{a_h \sim \pi(\cdot | s_h)}\left[\nabla_\theta \ln \pi_\theta(a_h | s_h) Q_h^{\pi_\theta}(s_h, a_h)\right]\right]$$

(where $s_h \sim \rho_\theta$ is a sample from the marginal state distribution at time h)

- To see this, first note:

$$\mathbb{E}_{x \sim P_\theta}\left[\nabla \log P_\theta(x) c\right] =$$

- The claims follows due to that for any constant c,

$$\mathbb{E}_{x \sim P_\theta}\left[\nabla \log P_\theta(x)\left(f(x) - c\right)\right] = \mathbb{E}_{x \sim P_\theta}\left[\nabla \log P_\theta(x) f(x)\right]$$

(M=1) PG with a Naive (constant) Baseline:

(M=1) PG with a Naive (constant) Baseline:

- Let try to use a constant (time-dependent) baseline: $b_h^{\theta} = \mathbb{E}_{\tau \sim \rho_{\theta}(\tau)} E \left[R_h(\tau) \right]$

(M=1) PG with a Naive (constant) Baseline:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$

- Let try to use a constant (time-dependent) baseline: $b_h^{\theta} = \mathbb{E}_{\tau \sim \rho_{\theta}(\tau)} E\left[R_h(\tau)\right]$

(M=1) PG with a Naive (constant) Baseline:

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :

- Let try to use a constant (time-dependent) baseline: $b_h^{\theta} = \mathbb{E}_{\tau \sim \rho_{\theta}(\tau)} E\left[R_h(\tau)\right]$

(M=1) PG with a Naive (constant) Baseline:

- Let try to use a constant (time-dependent) baseline: $b_h^{\theta} = \mathbb{E}_{\tau \sim \rho_{\theta}(\tau)} E\left[R_h(\tau)\right]$

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
   1. Sample M trajectories, $\tau_1, \ldots, \tau_M$ under π_{θ^k}. Set:
      $$\widetilde{b}_h = \frac{1}{M} \sum_{i=1}^{M} R_h(\tau_i)$$

(M=1) PG with a Naive (constant) Baseline:

- Let try to use a constant (time-dependent) baseline: $b_h^{\theta} = \mathbb{E}_{\tau \sim \rho_\theta(\tau)} E\left[R_h(\tau)\right]$

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
   1. Sample M trajectories, $\tau_1, \ldots, \tau_M$ under π_{θ^k}. Set:
      $$\widetilde{b}_h = \frac{1}{M} \sum_{i=1}^{M} R_h(\tau_i)$$
   2. Obtain a trajectory $\tau \sim \rho_{\theta^k}$
      Set $\widetilde{\nabla}_\theta J(\theta^k) = \sum_{h=0}^{H-1} \nabla \ln \pi_{\theta^k}(a_h \mid s_h)\left(R_h(\tau) - \widetilde{b}_h\right)$

(M=1) PG with a Naive (constant) Baseline:

- Let try to use a constant (time-dependent) baseline: $b_h^{\theta} = \mathbb{E}_{\tau \sim \rho_{\theta}(\tau)} E\left[R_h(\tau)\right]$

1. Initialize θ_0, parameters: $\eta_1, \eta_2, \ldots$
2. For k = 0, … :
    1. Sample M trajectories, $\tau_1, \ldots, \tau_M$ under π_{θ^k}. Set:
    $$\widetilde{b}_h = \frac{1}{M} \sum_{i=1}^{M} R_h(\tau_i)$$
    2. Obtain a trajectory $\tau \sim \rho_{\theta^k}$
    Set $\widetilde{\nabla}_{\theta} J(\theta^k) = \sum_{h=0}^{H-1} \nabla \ln \pi_{\theta^k}(a_h | s_h) \left(R_h(\tau) - \widetilde{b}_h \right)$
    3. Update: $\theta^{k+1} = \theta^k + \eta^k \widetilde{\nabla}_{\theta} J(\theta^k)$

The Advantage Function (finite horizon)

$$V_h^\pi(s) = \mathbb{E}\left[\sum_{\tau=h}^{H-1} r(s_\tau, a_\tau) \middle| s_h = s\right] \qquad Q_h^\pi(s,a) = \mathbb{E}\left[\sum_{\tau=h}^{H-1} r(s_\tau, a_\tau) \middle| (s_h, a_h) = (s,a)\right]$$

The Advantage Function (finite horizon)

$$V_h^\pi(s) = \mathbb{E}\left[\sum_{\tau=h}^{H-1} r(s_\tau, a_\tau) \middle| s_h = s\right] \qquad Q_h^\pi(s,a) = \mathbb{E}\left[\sum_{\tau=h}^{H-1} r(s_\tau, a_\tau) \middle| (s_h, a_h) = (s,a)\right]$$

- The Advantage function is defined as:

$$A_h^\pi(s,a) = Q_h^\pi(s,a) - V_h^\pi(s)$$

The Advantage Function (finite horizon)

$$V_h^\pi(s) = \mathbb{E}\left[\sum_{\tau=h}^{H-1} r(s_\tau, a_\tau) \middle| s_h = s\right] \qquad Q_h^\pi(s,a) = \mathbb{E}\left[\sum_{\tau=h}^{H-1} r(s_\tau, a_\tau) \middle| (s_h, a_h) = (s,a)\right]$$

- The Advantage function is defined as:

$$A_h^\pi(s,a) = Q_h^\pi(s,a) - V_h^\pi(s)$$

- We have that:

$$E_{a\sim\pi(\cdot|s)}\left[A_h^\pi(s,a) \middle| s, h\right] = \sum_a \pi(a\,|\,s) A_h^\pi(s,a) = \ ?? \qquad 0$$

The Advantage Function (finite horizon)

$$V_h^\pi(s) = \mathbb{E}\left[\sum_{\tau=h}^{H-1} r(s_\tau, a_\tau) \,\middle|\, s_h = s\right] \qquad Q_h^\pi(s,a) = \mathbb{E}\left[\sum_{\tau=h}^{H-1} r(s_\tau, a_\tau) \,\middle|\, (s_h, a_h) = (s, a)\right]$$

- The Advantage function is defined as:

$$A_h^\pi(s,a) = Q_h^\pi(s,a) - V_h^\pi(s)$$

- We have that:

$$E_{a\sim\pi(\cdot|s)}\left[A_h^\pi(s,a) \,\middle|\, s, h\right] = \sum_a \pi(a\,|\,s) A_h^\pi(s,a) = \;\; ??$$

$\forall a$ $\quad A^{*}_{a}(s, a)$

- What do we know about $A_h^{\pi^\star}(s,a)$?

The Advantage Function (finite horizon)

$$V_h^\pi(s) = \mathbb{E}\left[\sum_{\tau=h}^{H-1} r(s_\tau, a_\tau) \middle| s_h = s\right] \qquad Q_h^\pi(s,a) = \mathbb{E}\left[\sum_{\tau=h}^{H-1} r(s_\tau, a_\tau) \middle| (s_h, a_h) = (s,a)\right]$$

- The Advantage function is defined as:

$$A_h^\pi(s,a) = Q_h^\pi(s,a) - V_h^\pi(s)$$

- We have that:

$$E_{a\sim\pi(\cdot|s)}\left[A_h^\pi(s,a) \middle| s, h\right] = \sum_a \pi(a \mid s) A_h^\pi(s,a) = \ ??$$

- What do we know about $A_h^{\pi^\star}(s,a)$?

- For the discounted case, $A^\pi(s,a) = Q^\pi(s,a) - V^\pi(s)$

Summary:

1. REINFORCE (a direct application of the likelihood ratio method)
2. Variance Reduction: with baselines

Attendance:

bit.ly/3RcTC9T

Feedback:

bit.ly/3RHtlxy