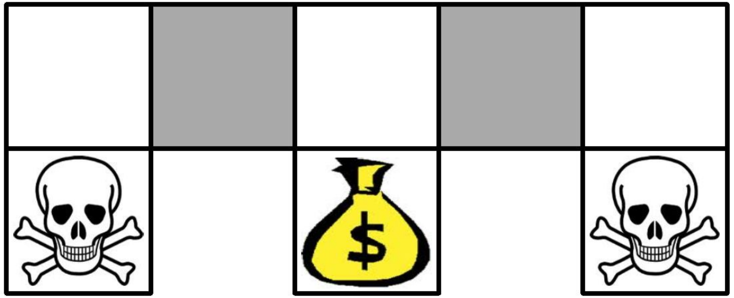
AIL 722: Reinforcement Learning

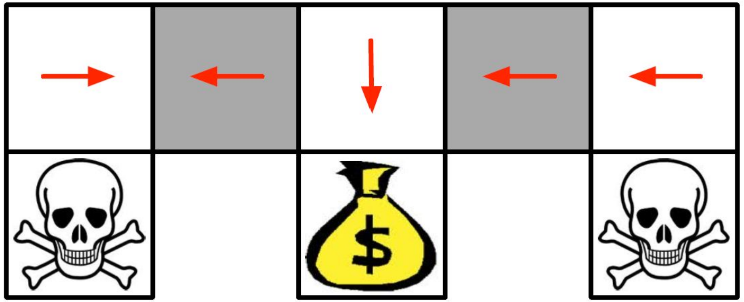
Lecture 32: Reinforce algorithm

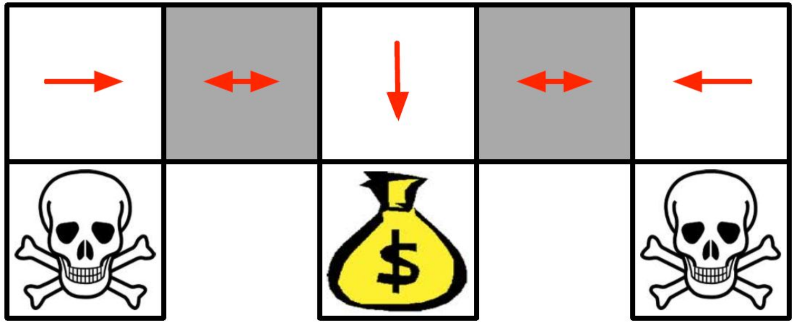
Raunak Bhattacharyya

YARDI SCHOOL OF ARTIFICIAL INTELLIGENCE
INDIAN INSTITUTE OF TECHNOLOGY DELHI

Aliased Grid World







Adapted from Hado van Hasselt

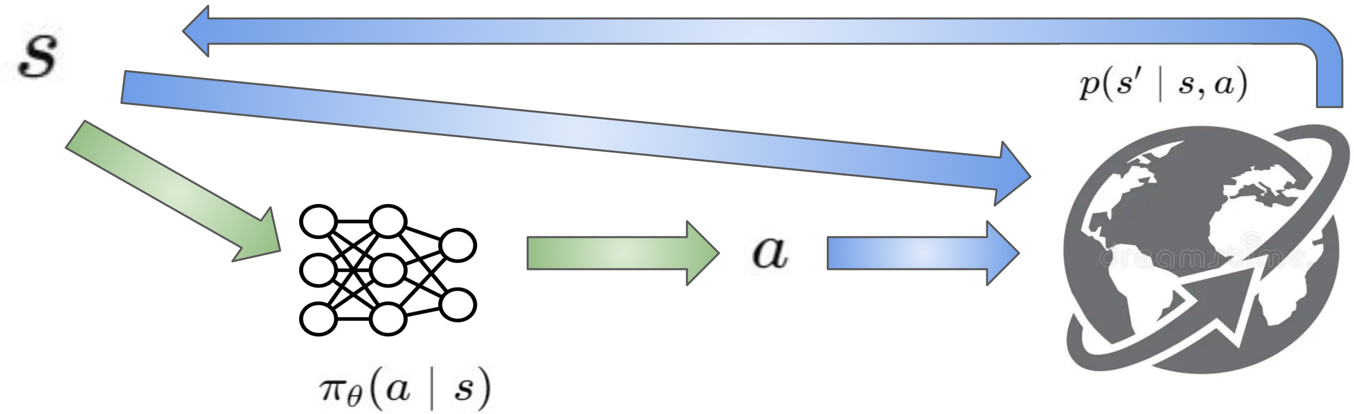
Advent of Policy Gradient Methods

- Oriented towards finding deterministic policies
- Arbitrary change in action value can cause it to be selected/not selected
    - Not converge
- Instead of approx value func and then deterministic pol, direct stochastic policy

Policy Gradient Methods for Reinforcement Learning with Function Approximation

Richard S. Sutton, David McAllester, Satinder Singh, Yishay Mansour
AT&T Labs – Research, 180 Park Avenue, Florham Park, NJ 07932

RL Objective



$$p_\theta(s_1, a_1, \ldots, s_T, a_T) = p(s_1) \prod_{t=1}^{T} \pi_\theta(a_t \mid s_t)\, p(s_{t+1} \mid s_t, a_t)$$

$p_\theta(\tau)$

$$\theta^* = \arg\max_\theta \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[\sum_t r(s_t, a_t) \right]$$

Finding the Objective Value

$$\theta^* = \arg\max_\theta \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[\sum_{t=1}^{T} r(s_t, a_t) \right]$$

$$\text{Let } r(\tau) = \sum_{t=1}^{T} r(s_t, a_t)$$

$$\text{Thus, } J(\theta) = \mathbb{E}_{\tau \sim p_\theta(\tau)}[r(\tau)]$$

$$\approx \frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{T} r(s_{i,t}, a_{i,t})$$

Take samples and estimate the expectation

Rollouts generated by running the policy

Optimising the Objective Value

$$J(\theta) = \mathbb{E}_{\tau \sim p_\theta(\tau)}[r(\tau)]$$

$$\text{i.e., } J(\theta) = \int p_\theta(\tau) r(\tau)\, d\tau$$

$$\nabla_\theta J(\theta) = \int \nabla_\theta p_\theta(\tau) r(\tau)\, d\tau$$

$$p_\theta(\tau) = p_\theta(s_1, a_1, \ldots, s_T, a_T)$$

$$p_\theta(s_1, a_1, \ldots, s_T, a_T) = p(s_1) \prod_{t=1}^{T} \pi_\theta(a_t \mid s_t)\, p(s_{t+1} \mid s_t, a_t)$$

To compute the gradient, we need to know the initial state distribution and transition probability distribution

Gradient Expression

$$\nabla_\theta J(\theta) = \int \nabla_\theta p_\theta(\tau) r(\tau)\, d\tau$$

$$r(\tau) = \sum_{t=1}^{T} r(s_t, a_t)$$

Convert the integral to an expected value, and estimate it using samples

What's the chain of thought?

$$J(\theta) = \mathbb{E}_{\tau \sim p_\theta(\tau)}[r(\tau)]$$

$$= \int p_\theta(\tau) r(\tau)\, d\tau$$

$$\nabla_\theta J(\theta) = \int \nabla_\theta p_\theta(\tau) r(\tau)\, d\tau$$

$$= \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[\nabla_\theta \log p_\theta(\tau)\, r(\tau)\right]$$

Goal: Isolate $p_\theta(\tau)$

$$\nabla_\theta p_\theta(\tau) = p_\theta(\tau) \nabla_\theta \log p_\theta(\tau)$$

$$\nabla_\theta J(\theta) = \int p_\theta(\tau) \nabla_\theta \log p_\theta(\tau) r(\tau)\, d\tau$$

$$= \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[\nabla_\theta \log p_\theta(\tau)\, r(\tau)\right]$$

Gradient Estimator

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[\nabla_\theta \log p_\theta(\tau)\, r(\tau)\right]$$

Recall

$$p_\theta(\tau) = p_\theta(s_1, a_1, \ldots, s_T, a_T)$$

$$p_\theta(s_1, a_1, \ldots, s_T, a_T) = p(s_1) \prod_{t=1}^{T} \pi_\theta(a_t \mid s_t)\, p(s_{t+1} \mid s_t, a_t)$$

$$\log p_\theta(\tau) = \log p(s_1) + \sum_{t=1}^{T} \log \pi_\theta(a_t \mid s_t) + \log p(s_{t+1} \mid s_t, a_t)$$

$$\nabla_\theta \log p_\theta(\tau) = \nabla_\theta \sum_{t=1}^{T} \log \pi_\theta(a_t \mid s_t)$$

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[\left(\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_t \mid s_t) \right) \left(\sum_{t=1}^{T} r(s_t, a_t) \right) \right]$$

Gradient Computation

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim p_\theta(\tau)}\left[\left(\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_t \mid s_t)\right)\left(\sum_{t=1}^{T} r(s_t, a_t)\right)\right]$$

$$J(\theta) = \mathbb{E}_{\tau \sim p_\theta(\tau)}\left[r(\tau)\right]$$

$$\approx \frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T} r(s_{i,t}, a_{i,t})$$

Estimating expectation

$$\nabla_\theta J(\theta) \approx \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_{i,t} \mid s_{i,t})\right)\left(\sum_{t=1}^{T} r(s_{i,t}, a_{i,t})\right)$$

No need to know the initial state distribution or transition dynamics

$$p_\theta(s_1, a_1, \ldots, s_T, a_T) = p(s_1) \prod_{t=1}^{T} \pi_\theta(a_t \mid s_t)\, p(s_{t+1} \mid s_t, a_t)$$

The Reinforce Algorithm

$$\nabla_\theta J(\theta) \approx \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{t=1}^{T}\nabla_\theta \log \pi_\theta(a_{i,t} \mid s_{i,t})\right)\left(\sum_{t=1}^{T} r(s_{i,t}, a_{i,t})\right)$$

REINFORCE: Monte-Carlo Policy-Gradient Control (episodic) for π_*

Input: a differentiable policy parameterization $\pi(a|s, \boldsymbol{\theta})$
Algorithm parameter: step size $\alpha > 0$
Initialize policy parameter $\boldsymbol{\theta} \in \mathbb{R}^{d'}$ (e.g., to $\mathbf{0}$)

Loop forever (for each episode):
    Generate an episode $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$, following $\pi(\cdot|\cdot, \boldsymbol{\theta})$
    Loop for each step of the episode $t = 0, 1, \ldots, T-1$:
        $G \leftarrow \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k$ $\qquad (G_t)$
        $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha\gamma^t G \nabla \ln \pi(A_t|S_t, \boldsymbol{\theta})$

Section 13.1, Reinforcement Learning: An Introduction, Sutton & Barto

Summary & Announcements

- Summary
  - Aliased grid world
    - Stochastic optimal policy
  - Policy gradient expression
  - Reinforce algorithm