Lecture 7: Policy gradient

7.1 Motivation

previous
- $V_{\pi_\theta}(s) \sim V_\theta(s)$
- $q_{\pi_\theta}(s,a) \sim q_\theta(s,a)$
- policy ~ ε-greedy

In this lecture.

$\pi(s,a) \leftarrow \pi_\theta(s,a) = \text{prob}(a|s,\theta)$

parameterize policy.

Advantage

David Silver
1. Better convergence
2. Effective in high-dim / continuous action space
3. can learn stochastic properties

Sutton
4. a good way to inject prior knowledge.
   <1> 领域.
   <2> 主观

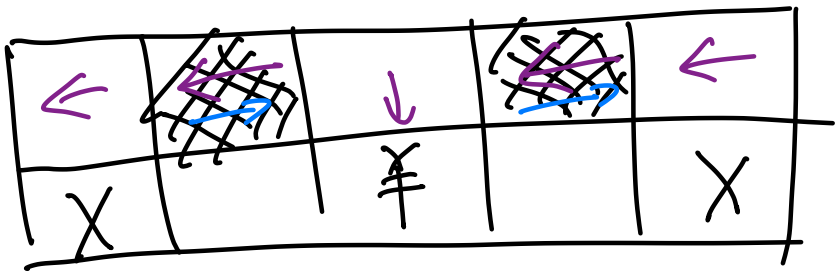
Disadvantage.
1. Typically converge to local minima
2. Evaluating a policy is often is inefficient, high variance.

Ex1. rock - paper - scissors

deterministic ⟶ exploit ⟶ fail

random (Nash equilibrium)

Ex2.



① shadow area 为什么区别

these two grids use same policy.

7.2. policy objective functions

Goal: Find π_θ (θ), find best θ

Measure quality of π_θ

① start value.

$$J_1(\theta) = V^{\pi_\theta}(s_1) = E_{\pi_\theta}(v_1)$$

② average value.

$$J_{avV}(\theta) = \sum_s d^{\pi_\theta}(s) V^{\pi_\theta}(s)$$

③ average reward per time-step.

$$J_{avR}(\theta) = \sum_s d^{\pi_\theta}(s) \sum_a \pi_\theta(s,a) R_s^a$$

$d^{\pi_\theta}(s)$ 是在 π_θ 下 θ 的稳定分布

7.2.1 Ex: one-step MDP

(1階段. $d(s)$ 与 θ 无关)

$$\nabla_\theta J_{avR}(\theta) = \sum_s d(s) \sum_a \nabla_\theta \pi_\theta(s,a) R_s^a$$

$$= \sum_s d(s) \sum \pi_\theta(s,a) \frac{\nabla_\theta \pi_\theta(s,a)}{\pi_\theta(s,a)} R_s^a$$

$$= \sum_s d(s) \sum \pi_\theta(s,a) \nabla_\theta \ln \pi_\theta(s,a) R_s^a$$

$$= E_{s,a}[\nabla_\theta \ln \pi_\theta(s,a) R_s^a]$$

$$= E_{\pi_\theta}[\nabla_\theta \ln \pi_\theta(s,a) R_s^a]$$

$$\Delta\theta = \nabla_\theta \ln \pi_\theta(s,a) R_s^a$$

Score function: $\nabla_\theta \ln \pi_\theta(s,a)$

Ex2. $\pi_\theta(s,a) \propto e^{\phi(s,a)^T\theta}$

$$\pi_\theta(s,a) = \frac{1}{\sum_a e^{\phi(s,a)^T\theta}} e^{\phi(s,a)^T\theta} \quad \text{prob}(a|s,\theta)$$

$$\nabla_\theta \ln \pi_\theta(s,a) = \nabla_\theta \ln\left(\frac{\quad}{\sum}\right)$$

$$= \nabla_\theta \ln e^{\phi^T\theta} - \frac{1}{\sum e^{\phi^T\theta}} \sum e^{\phi^T\theta} \phi$$

$$= \phi - \langle \phi \rangle$$

7.3 REINFORCE

$J(\theta) = V_{\pi_\theta}(s)$

key is to find $\nabla J(\theta)$

Policy Gradient Theorem

For J_1, J_{avR}, $\frac{1}{1-\gamma} J_{avV}$

$$\nabla_\theta J(\theta) = E_{\pi_\theta}\left(\nabla_\theta \ln \pi_\theta(S,a) \cdot Q^{\pi_\theta}(S,a)\right)$$

$\nabla_\theta \ln \pi_\theta(S,a)$: 可算 ; $Q^{\pi_\theta}(S,a)$: 可估

Prove.: $\gamma = 1$

$$\nabla V_\pi(s) = \nabla\left(\sum_a \pi(a|s)\, q_\pi(s,a)\right)$$

$$= \sum_a \nabla\pi(a|s)\cdot q_\pi(s,a) + \pi(a|s)\cdot \nabla q_\pi(s,a)$$

$$= \sum_a \nabla\pi(a|s)\cdot q_\pi(s,a) + \pi(a|s)\nabla\left(\sum_{s',r} p(s',r|s,a)(r + V_\pi(s'))\right)$$

$$= \sum_a \nabla\pi(a|s)\cdot q_\pi(s,a) + \pi(a|s)\nabla\left(\sum_{s'} p(s'|s,a) V_\pi(s')\right)$$

$$= \sum_a \nabla\pi(a|s)\cdot q_\pi(s,a) + \pi(a|s)\sum_{s'} p(s'|s,a)\nabla V_\pi(s')$$

$\sum_{s'}\sum_a \pi(a|s)\, p(s'|s,a)\cdots$

$$= \sum_a \nabla\pi(a|s)\cdot q_\pi(s,a) + \pi(a|s)\sum_{s'} p(s'|s,a)\left[\sum_{a'}\nabla\pi(a'|s')\cdot q_\pi(s',a') + \pi(a|s)\sum_{s''} p(s''|s',a')\nabla V_\pi(s'')\right]$$

$\vdots$

$$= \sum_{x \in S} \left(\sum_{k=0}^{\infty} P(s \to x, k, \pi) \right) \sum_a \nabla \pi(a|x)\, q_\pi(x, a)$$

$$= \sum_x d(x) \sum_a \nabla \pi(a|x)\, q_\pi(x, a)$$

$$\nabla J(\theta) = \nabla V_\pi(s_0)$$

$$= \sum_s d(s) \sum_a \nabla \pi(a|s)\, q_\pi(s, a)$$

$$= \sum_s d(s) \sum_a \pi(a|s) \nabla \ln \pi(a|s) \cdot q_\pi(s, a)$$

$$= E_{s,a} \left(\nabla \ln \pi(a|s) \cdot q_\pi(s, a) \right)$$

$$= E_\pi \left(\nabla \ln \pi(a|s) \cdot q_\pi(s, a) \right)$$

漂亮 !!!

Monte-Carlo Policy Gradient (REINFORCE)

- Update parameters by stochastic gradient ascent
- Using policy gradient theorem
- Using return v_t as an unbiased sample of $Q^{\pi_\theta}(s_t, a_t)$

$$\Delta\theta_t = \alpha \nabla_\theta \log \pi_\theta(s_t, a_t) v_t$$

function REINFORCE
  Initialise θ arbitrarily
  for each episode $\{s_1, a_1, r_2, ..., s_{T-1}, a_{T-1}, r_T\} \sim \pi_\theta$ **do**
    for $t = 1$ to $T - 1$ **do**
      $\theta \leftarrow \theta + \alpha \nabla_\theta \log \pi_\theta(s_t, a_t) v_t$
    end for
  end for
  return θ
end function

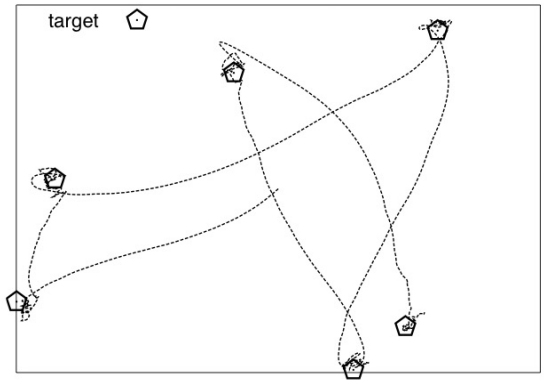
REINFORCE: Monte-Carlo Policy-Gradient Control (episodic) for π_*

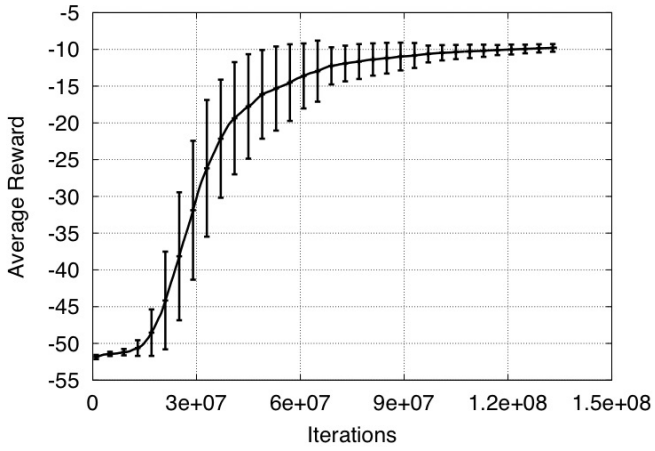
Input: a differentiable policy parameterization $\pi(a|s,\boldsymbol{\theta})$
Algorithm parameter: step size $\alpha > 0$
Initialize policy parameter $\boldsymbol{\theta} \in \mathbb{R}^{d'}$ (e.g., to $\mathbf{0}$)

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 $t = 0, 1, \ldots, T-1$:
    $G \leftarrow \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k$ (G_t)
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha \gamma^t G \nabla \ln \pi(A_t|S_t,\boldsymbol{\theta})$

$\mu \neq 1$

Puck World Example





- Continuous actions exert small force on puck
- Puck is rewarded for getting close to target
- Target location is reset every 30 seconds
- Policy is trained using variant of Monte-Carlo policy gradient

☆: MC still has high variance.

7.4 Reduce Variance.

① use a Base Line.

$$\nabla J(\theta) \propto \sum_s d(s) \sum_a \left(q_\pi(s,a) - b(s)\right) \nabla \pi(a|s,\theta)$$

$$= \sum_s d(s) \sum_a q_\pi(s,a) \nabla \pi(a|s,\theta) - \sum_s d(s) \sum_a b(s) \nabla \pi(a|s,\theta)$$

$$\sum_s d(s) b(s) \sum_a \nabla \pi(a|s,\theta)$$

$$= \sum_s d(s) b(s) \nabla \sum_a \pi(a|s,\theta)$$

$$= 0$$

REINFORCE with Baseline (episodic), for estimating $\pi_\theta \approx \pi_*$

Input: a differentiable policy parameterization $\pi(a|s,\boldsymbol{\theta})$
Input: a differentiable state-value function parameterization $\hat{v}(s,\mathbf{w})$
Algorithm parameters: step sizes $\alpha^{\boldsymbol{\theta}} > 0$, $\alpha^{\mathbf{w}} > 0$
Initialize policy parameter $\boldsymbol{\theta} \in \mathbb{R}^{d'}$ and state-value weights $\mathbf{w} \in \mathbb{R}^d$ (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 $t = 0, 1, \ldots, T-1$:
    $G \leftarrow \sum_{k=t+1}^{T} \gamma^{k-t-1} R_k$ $\quad (G_t)$
    $\delta \leftarrow G - \hat{v}(S_t,\mathbf{w})$
    $\mathbf{w} \leftarrow \mathbf{w} + \alpha^{\mathbf{w}} \delta \nabla \hat{v}(S_t,\mathbf{w})$
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha^{\boldsymbol{\theta}} \gamma^t \delta \nabla \ln \pi(A_t|S_t,\boldsymbol{\theta})$

② Using Critic

Critic : update action-value function

actor : update policy parameters

Action-Value Actor-Critic

- Simple actor-critic algorithm based on action-value critic
- Using linear value fn approx. $Q_w(s, a) = \phi(s, a)^\top w$

  Critic Updates w by linear TD(0)

  Actor Updates θ by policy gradient

function QAC

Initialise s, θ

Sample $a \sim \pi_\theta$

for each step **do**

Sample reward $r = \mathcal{R}_s^a$; sample transition $s' \sim \mathcal{P}_{s,\cdot}^a$

Sample action $a' \sim \pi_\theta(s', a')$

$\delta = r + \gamma Q_w(s', a') - Q_w(s, a)$

$\theta = \theta + \alpha \nabla_\theta \log \pi_\theta(s, a) Q_w(s, a)$

$w \leftarrow w + \beta \delta \phi(s, a)$

$a \leftarrow a', s \leftarrow s'$

end for

end function

- The policy gradient has many equivalent forms

$$
\begin{aligned}
\nabla_\theta J(\theta) &= \mathbb{E}_{\pi_\theta}\left[\nabla_\theta \log \pi_\theta(s,a)\ v_t\right] && \text{REINFORCE} \\
&= \mathbb{E}_{\pi_\theta}\left[\nabla_\theta \log \pi_\theta(s,a)\ Q^w(s,a)\right] && \text{Q Actor-Critic} \\
&= \mathbb{E}_{\pi_\theta}\left[\nabla_\theta \log \pi_\theta(s,a)\ A^w(s,a)\right] && \text{Advantage Actor-Critic} \\
&= \mathbb{E}_{\pi_\theta}\left[\nabla_\theta \log \pi_\theta(s,a)\ \delta\right] && \text{TD Actor-Critic} \\
&= \mathbb{E}_{\pi_\theta}\left[\nabla_\theta \log \pi_\theta(s,a)\ \delta e\right] && \text{TD}(\lambda)\text{ Actor-Critic}
\end{aligned}
$$

$$A^w(s,a) = Q^{\pi_\theta}(s,a) - V^{\pi_\theta}(s,a)$$

$$\delta = R + \gamma V(s',w) - V(s,w)$$