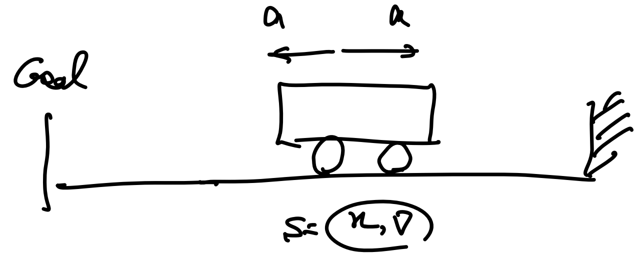
Lecture 25 - April 18, 2023

- Deep Q-Network (DQN) → Finite action
  - Deep Q-Network (DQN)
  - Double DQN
  - Prioritized DQN
  - Dueling DQN
  - Noisy-Net DQN
- Deep Policy Gradients (DPG) → large/continuous Action
  - REINFORCE
  - REINFORCE with Baseline
  - Advantage Actor Critic (A2C)
  - Deep Deterministic Policy Gradient (DDPG)

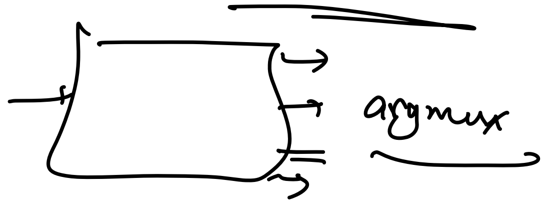
HW5 → Due April 18

TA's office hour: Wendsdays, 2pm-3pm (in-person)
Fridays, 2pm-3pm (virtual)



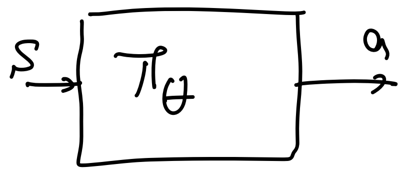
Learning policy seems to be easier than Q-Value.

$Q(s, a)$



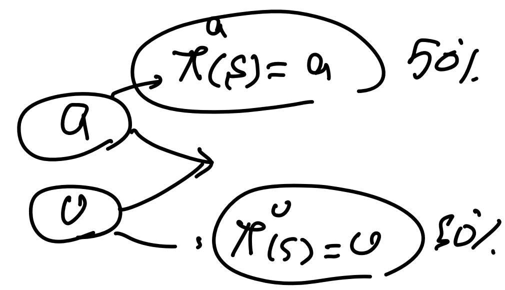
① smoothness

② Multi-Agent ⇒ Stochastic



$$\pi^*(a|s) = \begin{cases} 0.2 & a^1 \\ 0.8 & a^2 \end{cases}$$

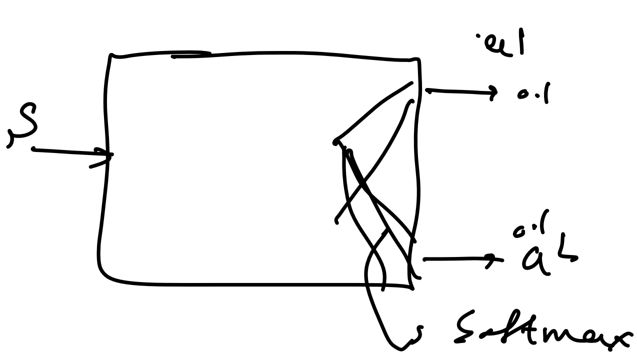
$a^1 \quad a^2$

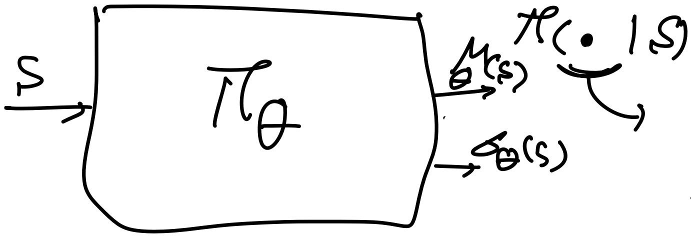


on-policy

$\pi_\theta(s) = a$

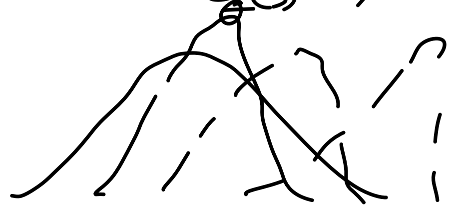
$$\pi_\theta(a|s) = \begin{cases} 0.2 & a1 \\ 0.3 & a2 \\ \vdots \end{cases}$$



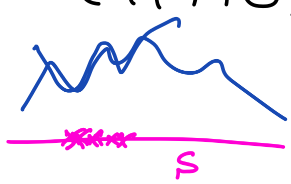


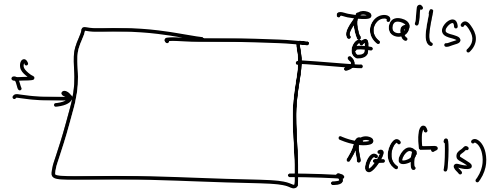
$a \in [-1 \quad 1]$

$$\pi_\theta(a|s) = \frac{1}{\sqrt{2\pi}\,\sigma_\theta(s)} \exp\left(-\frac{(a-\mu_\theta(s))^2}{2\sigma_\theta^2(s)}\right)$$



$$\pi_\theta(a|s) = \pi(a|s,\theta) = P(a_t = a \mid S_t = s, \theta)$$



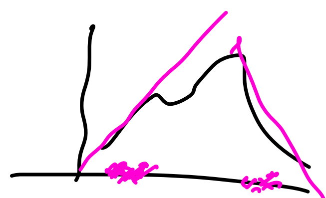


$J(\theta) = V_{\pi_\theta}(s_0)$ → all s_0 under π_θ

Goal: find θ that maximizes $J(\theta)$

$$J(\theta) = E[R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \cdots \mid a_{t:\infty} \sim \pi_\theta, s]$$

$$X = E[Q(s, \pi_\theta)$$



$$J(\theta) = \mathbb{E}_{s \sim \pi_\theta}\left[\sum_{a \in A} Q_{\pi_\theta}(s,a)\, \pi_\theta(a|s) \;\middle|\; s \right]$$

Maximizing $J(\theta) \Rightarrow \theta^* = \arg\max \underline{J(\theta)}$

$$\mathbb{E}_{\pi_\theta}\left[\sum_a \pi_\theta(a|s)\, Q_{\pi_\theta}(s,a)\right]$$

$$\overset{\text{new}}{\theta} \leftarrow \overset{\text{old}}{\theta} + \alpha \nabla_\theta J(\theta)$$

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{a \in A} Q_{\pi_\theta}(s,a)\, \nabla_\theta \pi(a|s,\theta) \right]$$

$$\propto \underbrace{\sum_s \mu(s)}_{\text{Distribution of state } s \text{ under policy } \pi_\theta} \sum_{a \in A} Q_{\pi_\theta}(s,a)\, \nabla_\theta \pi(a|s,\theta)$$

REINFORCE

$$\nabla J(\theta) \propto \sum_s \mu(s) \sum_a Q_{\pi_\theta}(s,a) \nabla_\theta \pi(a|s,\theta)$$

$$= E_{\pi_\theta}\left[\sum_a Q_{\pi_\theta}(s,a) \nabla_\theta \pi(a|s,\theta)\right]$$

$$= E_{\pi_\theta}\left[\sum_a \pi(a|s,\theta)\, Q_{\pi_\theta}(s,a)\, \frac{\nabla_\theta (a|s,\theta)}{\pi(a|s,\theta)}\right]$$

$$= E_{\pi_\theta}\left[\sum_a \pi(a|s_t,\theta)\, Q_{\pi_\theta}(s_t,a)\, \nabla_\theta \ln \pi_\theta(a|s_t,\theta)\right]$$

$$\nabla J(\theta) = E_{\pi_\theta}\left[Q_{\pi_\theta}(s_t, a_t)\, \nabla_\theta \ln \pi_\theta(a_t|s_t,\theta)\right]$$

$a_t \sim \pi(a|s_t,\theta)$

REINForce Trick

$$\nabla \ln x = \frac{\nabla x}{x}$$

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

Modification of Algorithm 13.3 of "Reinforcement Learning: An Introduction, Second Edition".

π_θ

$s_0, a_0, s_1, r_1 \rightarrow G_1 = r_1 + \gamma r_2 + \gamma^2 r_3 + \cdots \gamma^{T-1} r_T$

$s_1, a_1, s_2, r_2 \rightarrow G_2 = r_2 + \gamma r_3 + \cdots$

$\vdots$

$s_{T-1}, a_{T-1}, s_T, r_T \rightarrow G_T = r_T$

$$\nabla J(\theta) = E_{\pi_\theta}\left[Q_{\pi_\theta}(s_t, a_t) \nabla_\theta \ln \pi_\theta(a_t | s_t, \theta) \right]$$

$$\approx \frac{1}{N} \sum_{i=1}^{N} G_t^i \nabla_\theta \ln (a_i | s_i, \theta)$$

$$= \frac{1}{T} \left[G_1 \nabla_\theta \ln (a_0 | s_0, \theta) + G_2 \nabla_\theta \ln (a \cdots) \right)$$

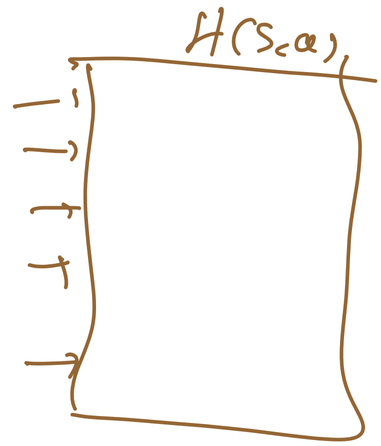
$$\theta \leftarrow \theta + \alpha \nabla J(\theta)$$

Simpler Function Approximation:

Actor critic

$$\pi(a|s) = \frac{e^{H(s,a)}}{\sum_{a'} e^{H(s,a')}}$$

$H(s,a) \rightarrow$ pref



$$H(s,a) = \theta^T \Phi(s,a)$$

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta)$$

$$\nabla_\theta J(\theta) = E\left[Q_{\pi_\theta}(s_t, a_t) \nabla_\theta \ln \pi_\theta(a_t | s_t, \theta) \right]$$

$$\pi_\theta(a|s) = \frac{e^{\theta^T \Phi(s,a)}}{\sum_{a'} e^{\theta^T \Phi(s,a')}}$$

$$\nabla_\theta \ln \pi_\theta(a_t | s_t, \theta) = \Phi(s_t, a_t) - \sum_{a'} \pi(a' | s_t, \theta) \Phi(s_t, a')$$

II Gaussian Function App

$$\pi(a|s,\theta) = \frac{1}{\sigma(s,\theta)\sqrt{2\pi}} \exp\left(-\frac{(a-\mu(s,\theta))^2}{2\sigma(s,\theta)^2}\right)$$

$$\mu(s,\theta) = \theta_\mu^T \Phi_\mu(s) \qquad \theta = \begin{bmatrix}\theta_\mu \\ \theta_\sigma\end{bmatrix}$$

$$\sigma(s,\theta) = \theta_\sigma^T \phi_\sigma(s)$$

$$\nabla_\theta \ln \pi(a|s_t,\theta) = \quad \checkmark$$

REINFORCE with Baseline

$$\nabla_\theta J(\theta) \propto \sum_s \mu(s) \sum_a Q^\pi(s,a) \nabla_\theta \pi(a|s,\theta)$$

$$= \sum_s \mu(s) \sum_a \left(Q^\pi(s,a) - b(s)\right) \nabla_\theta \pi(a|s,\theta)$$

why? $\rightarrow \sum_a b(s) \nabla_\theta \pi(a|s,\theta) = b(s) \underbrace{\sum_a \nabla_\theta \pi(a|s,\theta)}_{\underbrace{\nabla_\theta \underbrace{\sum_a \pi(a|s,\theta)}_{1}}_{0}}$

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

$$\pi_\theta \left\{ \begin{array}{ll} s_0, a_0, r_1, s_1 & G_1 \\ s_1, a_1, r_2, s_2 & G_2 \\ \vdots & \\ s_{T-1}, a_{T-1}, r_T, s_T & G_T \end{array} \right\}$$

$$V(s, w) = V_w(s)$$

$$\pi_\theta(a \mid s)$$

$$L(w) = E\left[\left(G_t - V_w(s_t)\right)^2\right]$$

Gradient Decent

$$w \leftarrow w - \frac{1}{2}\alpha \nabla_w L(w)$$

$$w \leftarrow w + \alpha \left(G_t - V_w(s_t)\right) \nabla V_w(s_t)$$

$$J(\theta) \longrightarrow$$

$$\theta \leftarrow \theta + \alpha \underline{\nabla J(\theta)}$$

$$\left(G_t - V_w(s_t)\right) \nabla_\theta \ln \pi_\theta(a \mid s)$$

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

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

Modification of Algorithm 13.4 of "Reinforcement Learning: An Introduction, Second Edition"

REINFORCE $\rightarrow$ MC

TD

$$\theta_{t+1} = \theta_t + \alpha \left(\underbrace{G_t}_{R_{t+1} + \gamma \hat{V}(S_{t+1}, w)} - \hat{V}(S_t, w) \right) \nabla_\theta \ln \pi_\theta(a_t|S_t)$$

A2C

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

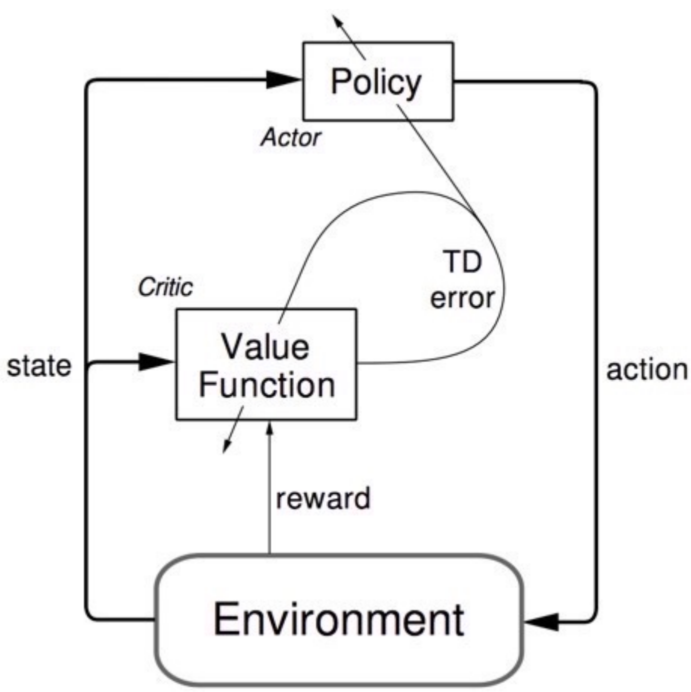
Input: a differentiable policy parameterization $\pi(a|s, \boldsymbol{\theta})$
Input: a differentiable state-value function parameterization $\hat{v}(s,\mathbf{w})$
Algorithm parameters: step sizes $\alpha^{\boldsymbol{\theta}} > 0$, $\alpha^{\mathbf{w}} > 0$
Initialize policy parameter $\boldsymbol{\theta}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

Modification of Algorithm 13.4 of "Reinforcement Learning: An Introduction, Second Edition"

One-step Actor–Critic (episodic), for estimating $\pi_{\boldsymbol{\theta}} \approx \pi_*$

Input: a differentiable policy parameterization $\pi(a|s, \boldsymbol{\theta})$
Input: a differentiable state-value function parameterization $\hat{v}(s,\mathbf{w})$
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):
  Initialize S (first state of episode)
  $I \leftarrow 1$
  Loop while S is not terminal (for each time step):
    $A \sim \pi(\cdot|S, \boldsymbol{\theta})$
    Take action A, observe S', R
    $\delta \leftarrow R + \gamma \hat{v}(S',\mathbf{w}) - \hat{v}(S,\mathbf{w})$ (if S' is terminal, then $\hat{v}(S',\mathbf{w}) \doteq 0$)
    $\mathbf{w} \leftarrow \mathbf{w} + \alpha^{\mathbf{w}} I \delta \nabla \hat{v}(S,\mathbf{w})$
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha^{\boldsymbol{\theta}} I \delta \nabla \ln \pi(A|S, \boldsymbol{\theta})$
    $I \leftarrow \gamma I$
    $S \leftarrow S'$



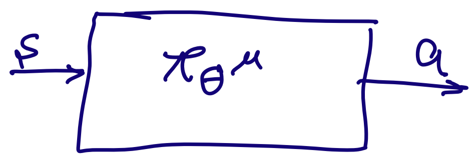
Deep Deterministic Policy Gradient (DDPG)

Q: Q-Network

Q': Target Network

θ^{μ}: Deterministic Policy Net

$\theta^{\mu'}$: Target Policy Net



D

Policy Network θ^{μ}

s_0, a_0, s_1, r_1

s_1, a_1, s_2, r_2

⋮

Update Q Net

minibatch B ⇒

s_0, a_0, s_1, r_1 $\quad y_1 = r_1 + \gamma \max Q'(s,a)$ ...

Target Net

$$w \leftarrow w + \alpha \frac{1}{B} \sum_{s,a,s',r \in B} \left(r + \gamma Q'(s', \mu'(s')) - Q(s,a) \right) \nabla Q(s,a)$$

$$J(\theta) = E[Q(s,a) \mid s_t = s, a_t = \mu(s)]$$

$$\nabla_\theta J(\theta) \approx \nabla_a Q_w(s,a) \nabla_{\theta^\mu} \mu_{\theta^\mu}(s)$$

Algorithm 1 DDPG algorithm

Randomly initialize critic network $Q(s, a|\theta^Q)$ and actor $\mu(s|\theta^\mu)$ with weights θ^Q and θ^μ.
Initialize target network Q' and μ' with weights $\theta^{Q'} \leftarrow \theta^Q$, $\theta^{\mu'} \leftarrow \theta^\mu$
Initialize replay buffer R
for episode = 1, M **do**
  Initialize a random process $\mathcal{N}$ for action exploration
  Receive initial observation state s_1
  for t = 1, T **do**
    Select action $a_t = \mu(s_t|\theta^\mu) + \mathcal{N}_t$ according to the current policy and exploration noise
    Execute action a_t and observe reward r_t and observe new state s_{t+1}
    Store transition (s_t, a_t, r_t, s_{t+1}) in R
    Sample a random minibatch of N transitions (s_i, a_i, r_i, s_{i+1}) from R
    Set $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$
    Update critic by minimizing the loss: $L = \frac{1}{N}\sum_i (y_i - Q(s_i, a_i|\theta^Q))^2$
    Update the actor policy using the sampled policy gradient:

$$\nabla_{\theta^\mu} J \approx \frac{1}{N} \sum_i \nabla_a Q(s, a|\theta^Q)|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu} \mu(s|\theta^\mu)|_{s_i}$$

    Update the target networks:

$$\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}$$

$$\theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}$$

  end for
end for

Algorithm 1 of the paper "Continuous Control with Deep Reinforcement Learning" by Timothy P. Lillicrap et al