Lecture 11: Policy Optimization II

Ziyu Shao

School of Information Science and Technology
ShanghaiTech University

May 07, 2025

Outline

1. Policy Gradient IV: Policy Gradient Theorem
2. Policy Gradient V: Entropy Regulation
3. Policy Gradient VI: Off-Policy Policy Gradient
4. Reading
5. References

Outline

1. Policy Gradient IV: Policy Gradient Theorem
2. Policy Gradient V: Entropy Regulation
3. Policy Gradient VI: Off-Policy Policy Gradient
4. Reading
5. References

Object Function in Policy Optimization

- Now we consider continuing & infinite horizon case
- Objective: Given a policy approximator $\pi_\theta(a|s)$ with parameter θ, find the best θ to maximize $J(\theta)$
- Metric 1: average state value, where state distribution $d(s)$ is independent of policy $\pi_\theta(a|s)$

$$J_{avV}(\theta) = \sum_{s \in \mathcal{S}} d(s) V^{\pi_\theta}(s) = \sum_{s \in \mathcal{S}} d(s) V^{\pi}(s)$$

- Choice of $d(s)$:
  - Uniform distribution: $d(s) = \frac{1}{|\mathcal{S}|}, \forall s.$
  - Fixed initial state s_0: $d(s_0) = 1$, $d(s) = 0, \forall s \neq s_0$. Then we denote $J_1(\theta)$ as follows:

$$J_1(\theta) = V^{\pi}(s_0).$$

Object Function in Policy Optimization

- Metric 1: average state value, state distribution $d(s)$ depends on policy $\pi(\pi_\theta(a|s))$, e.g. stationary distribution $d^\pi(s)$ under policy π.

$$J_{avV}(\theta) = \sum_{s\in\mathcal{S}} d^\pi(s) V^\pi(s)$$

- Suppose an agent collect rewards $\{R_{t+1}, t \geq 0\}$ by following the policy π, then

$$\begin{aligned}
&\lim_{n\to\infty} E\left[\sum_{t=0}^{n} \gamma^t R_{t+1}\right] = E\left[\sum_{t=0}^{\infty} \gamma^t R_{t+1}\right] \\
&= \sum_{s\in\mathcal{S}} d^\pi(s) E\left[\sum_{t=0}^{\infty} \gamma^t R_{t+1} | S_0 = s\right] = \sum_{s\in\mathcal{S}} d^\pi(s) V^\pi(s) \\
&= J_{avV}(\theta).
\end{aligned}$$

Object Function in Policy Optimization

- Metric 2: average one-step reward (or average reward per time-step), where state distribution $d(s)$ is stationary distribution $d^\pi(s)$ under policy π

$$J_{avR}(\theta) = \sum_{s \in \mathcal{S}} d^\pi(s) r^\pi(s)$$

where

$$r^\pi(s) = \sum_{a \in \mathcal{A}} \pi_\theta(a|s) r(s,a).$$

- Suppose an agent collect rewards $\{R_{t+1}, t \geq 0\}$ by following the policy π, then

$$J_{avR}(\theta) = \lim_{n \to \infty} \frac{1}{n} E\left[\sum_{t=0}^{n-1} R_{t+1}\right].$$

Proof of Metric 2

- First, for any state s_0:

$$\lim_{n\to\infty}\frac{1}{n}E\left[\sum_{t=0}^{n-1}R_{t+1}|S_0=s_0\right]=\lim_{n\to\infty}\frac{1}{n}\sum_{t=0}^{n-1}E[R_{t+1}|S_0=s_0]$$
$$=\lim_{n\to\infty}E[R_{n+1}|S_0=s_0]=\lim_{t\to\infty}E[R_{t+1}|S_0=s_0]$$

- Cesaro Means Theory: if $a_k \to a$, let $b_k = \frac{1}{k}\sum_{i=0}^{k-1} a_i$, then $b_k \to a$.

Proof of Metric 2

- Second, by LOTE:

$$\begin{aligned} E\left[R_{t+1} \mid S_0 = s_0\right] &= \sum_{s \in \mathcal{S}} E\left[R_{t+1} \mid S_t = s, S_0 = s_0\right] P(S_t = s \mid S_0 = s_0) \\ &= \sum_{s \in \mathcal{S}} E\left[R_{t+1} \mid S_t = s, S_0 = s_0\right] p^{(t)}(s_0, s) \\ &= \sum_{s \in \mathcal{S}} E\left[R_{t+1} \mid S_t = s\right] p^{(t)}(s_0, s) \\ &= \sum_{s \in \mathcal{S}} r^{\pi}(s) p^{(t)}(s_0, s) \end{aligned}$$

- where by LOTE

$$\begin{aligned} E\left[R_{t+1} \mid S_t = s\right] &= \sum_{a \in \mathcal{A}} E\left[R_{t+1} \mid A_t = a, S_t = s\right] P(A_t = a \mid S_t = s) \\ &= \sum_{a \in \mathcal{A}} r(s, a) \pi_\theta(a \mid s) = r^{\pi}(s) \end{aligned}$$

Proof of Metric 2

- Third:

$$\lim_{n\to\infty}\frac{1}{n}E\left[\sum_{t=0}^{n-1}R_{t+1}|S_0=s\right]=\lim_{t\to\infty}E\left[R_{t+1}|S_0=s_0\right]$$
$$=\lim_{t\to\infty}\sum_{s\in\mathcal{S}}r^{\pi}(s)p^{(t)}(s_0,s)=\sum_{s\in\mathcal{S}}r^{\pi}(s)\lim_{t\to\infty}p^{(t)}(s_0,s)$$
$$=\sum_{s\in\mathcal{S}}r^{\pi}(s)d^{\pi}(s)=J_{avR}(\theta)$$

- Then

$$\lim_{n\to\infty}\frac{1}{n}E\left[\sum_{t=0}^{n-1}R_{t+1}\right]=\lim_{n\to\infty}\frac{1}{n}\sum_{s\in\mathcal{S}}d^{\pi}(s)E\left[\sum_{t=0}^{n-1}R_{t+1}|S_0=s\right]$$
$$=\sum_{s\in\mathcal{S}}d^{\pi}(s)\lim_{n\to\infty}\frac{1}{n}E\left[\sum_{t=0}^{n-1}R_{t+1}|S_0=s\right]$$
$$=\sum_{s\in\mathcal{S}}d^{\pi}(s)J_{avR}(\theta)=J_{avR}(\theta)$$

Policy Gradient Theorem

- The policy gradient theorem generalizes the likelihood ratio approach to multi-step MDPs
- Replaces instantaneous reward r with long-term value $Q^{\pi}(s, a)$
- Policy gradient theorem applies to the above metrics

Theorem

For any differentiable policy $\pi_\theta(a|s)$, for any of the policy objective functions $J(\theta) = J_1(\theta), J_{avR}(\theta)$, or $\frac{1}{1-\gamma} J_{avV}(\theta)$, the policy gradient is

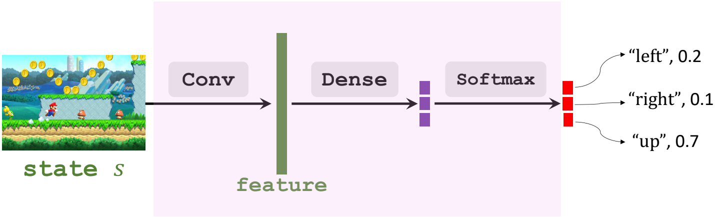
$$\nabla_\theta J(\theta) = \mathbb{E}_{S \sim d^{\pi}, A \sim \pi_\theta}[\nabla_\theta \log \pi_\theta(A|S) Q^{\pi}(S, A)]$$

Policy Network

- Stochastic gradient update:

$$\theta \leftarrow \theta + \alpha \cdot Q^{\pi}(s, a) \cdot \nabla_{\theta} \log \pi_{\theta}(a|s)$$

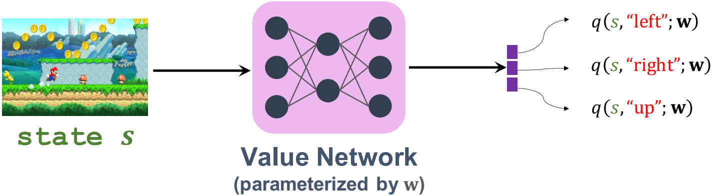
- Policy network $\pi_\theta(a|s)$: use a neural network to approximate policy $\pi(a|s)$



Policy Network
(parameterized by $\boldsymbol{\theta}$)

Value Network

- Given policy π, how to estimate $Q^\pi(s, a)$?
- Value Network $q(s, a; \mathbf{w})$!



Actor-Critic Algorithm

- Critic(Value Network): updates the value function parameters $\mathbf{w}$ and depending on the algorithm it could be action-value $q(s, a; \mathbf{w})$ or state-value $v(s; \mathbf{w})$
- Actor(Policy Network): updates the policy parameters θ for $\pi_\theta(a|s)$, in the direction suggested by the critic.

Simple Actor-Critic Policy Gradient Algorithm

given current state s_t, policy(value) network parameters $\theta(\mathbf{w})$, policy π_θ, and functions $\{q(s_t, a; \mathbf{w}), a \in \mathcal{A}\}$

1. Sample action $a_t \sim \pi_\theta(\cdot|s_t)$, perform action a_t and obtain reward r_{t+1} and new state s_{t+1}.
2. Sample action $a'_{t+1} \sim \pi_\theta(\cdot|s_{t+1})$ and collect the training data: quintuple $(s_t, a_t, r_{t+1}, s_{t+1}, a'_{t+1})$
3. Compute the TD target and TD error

$$
\begin{aligned}
y_t &= r_{t+1} + \gamma q(s_{t+1}, a'_{t+1}; \mathbf{w}) \\
\delta_t &= y_t - q(s_t, a_t; \mathbf{w})
\end{aligned}
$$

4. Update the parameters of value network as follows:

$$
\mathbf{w} \leftarrow \mathbf{w} + \alpha_{\mathbf{w}} \cdot \delta_t \cdot \nabla_{\mathbf{w}} q(s_t, a_t; \mathbf{w})
$$

5. Update the parameters of policy network as follows:

$$
\theta \leftarrow \theta + \alpha_\theta \cdot q(s_t, a_t; \mathbf{w}) \cdot \nabla_\theta \log \pi_\theta(a_t|s_t)
$$

Reduce Variance using Baselines

Theorem

For any differentiable policy $\pi_\theta(a|s)$, for any baseline b that does not depend on action, and for any of the policy objective functions $J(\theta) = J_1(\theta), J_{avR}(\theta)$, or $\frac{1}{1-\gamma} J_{avV}(\theta)$, the policy gradient is

$$\begin{aligned}\nabla_\theta J(\theta) &= \mathbb{E}_{S\sim d^\pi, A\sim\pi_\theta}[Q^\pi(S, A) \cdot \nabla_\theta \log \pi_\theta(A|S)] \\ &= \mathbb{E}_{S\sim d^\pi, A\sim\pi_\theta}[(Q^\pi(S, A) - b) \cdot \nabla_\theta \log \pi_\theta(A|S)]\end{aligned}$$

- The key of proof is to show

$$\mathbb{E}_{S\sim d^\pi, A\sim\pi_\theta}[b \cdot \nabla_\theta \log \pi_\theta(A|S)] = 0$$

Proof

- First, given any state s, we have

$$
\begin{aligned}
\mathbb{E}_{A\sim\pi_\theta(\cdot|s)}\left[b\cdot\nabla_\theta\log\pi_\theta(A|s)\right] &= b\cdot\mathbb{E}_{A\sim\pi_\theta(\cdot|s)}\left[\nabla_\theta\log\pi_\theta(A|s)\right]\\
&= b\cdot\left(\sum_{a\in\mathcal{A}}\pi_\theta(a|s)\nabla_\theta\log\pi_\theta(a|s)\right)\\
&= b\cdot\left(\sum_{a\in\mathcal{A}}\pi_\theta(a|s)\frac{1}{\pi_\theta(a|s)}\nabla_\theta\pi_\theta(a|s)\right)\\
&= b\cdot\left(\sum_{a\in\mathcal{A}}\nabla_\theta\pi_\theta(a|s)\right)\\
&= b\cdot\nabla_\theta\left(\sum_{a\in\mathcal{A}}\pi_\theta(a|s)\right)\\
&= b\cdot\nabla_\theta 1 = b\cdot 0 = 0.
\end{aligned}
$$

Proof

- Second, for any state S, now we have

$$\mathbb{E}_{A\sim\pi_\theta(\cdot|S)}\left[b\cdot\nabla_\theta\log\pi_\theta(A|S)\right]=0.$$

- Thus

$$\begin{aligned}
&\mathbb{E}_{S\sim d^\pi, A\sim\pi_\theta}[b\cdot\nabla_\theta\log\pi_\theta(A|S)]\\
=&\ \mathbb{E}_S\left[\mathbb{E}_{A\sim\pi_\theta(\cdot|S)}\left[b\cdot\nabla_\theta\log\pi_\theta(A|S)\right]\right]\\
=&\ \mathbb{E}_S\left[0\right]\\
=&\ 0
\end{aligned}$$

Advantage Actor-Critic Algorithm

- Stochastic gradient update with baseline b:

$$\theta \leftarrow \theta + \alpha \cdot (Q^\pi(s,a) - b) \cdot \nabla_\theta \log \pi_\theta(a|s)$$

- Since baseline b that does not depend on action, we choose $b = b_s = V^\pi(s)$ given state s:

$$\theta \leftarrow \theta + \alpha \cdot (Q^\pi(s,a) - V^\pi(s)) \cdot \nabla_\theta \log \pi_\theta(a|s)$$

- Advantage function $A^\pi(s,a)$:(relative measure of the importance of each action):

$$A^\pi(s,a) = Q^\pi(s,a) - V^\pi(s)$$

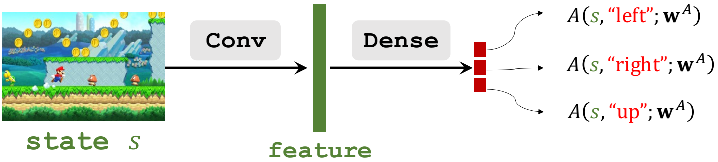
$$E_{A\sim\pi(\cdot|s)}[A^\pi(s,A)] = \sum_a Q^\pi(s,a) - V^\pi(s) = 0$$

- Stochastic gradient update:

$$\theta \leftarrow \theta + \alpha \cdot A^\pi(s,a) \cdot \nabla_\theta \log \pi_\theta(a|s)$$

Advantage Actor-Critic Algorithm: Option 1

- Use a neural network $A(s, a; \mathbf{w}^A)$ to directly estimate $A^\pi(s, a)$
- Or two neural networks to estimate $Q^\pi(s, a)$ and $V^\pi(s)$ individually



Advantage Actor-Critic Algorithm: Option 2

- Bellman Expectation Equation:

$$Q^{\pi}(s_t, a_t) = E_{S_{t+1}\sim p(\cdot|s_t,a_t)}[R_{t+1} + \gamma V^{\pi}(S_{t+1})].$$

- Suppose we know $(s_t, a_t, r_{t+1}, s_{t+1})$
- Unbiased estimation of $Q^{\pi}(s_t, a_t)$:

$$Q^{\pi}(s_t, a_t) \approx r_{t+1} + \gamma V^{\pi}(s_{t+1})$$

- Thus estimation of $A^{\pi}(s_t, a_t)$ is

$$A^{\pi}(s_t, a_t) \approx r_{t+1} + \gamma V^{\pi}(s_{t+1}) - V^{\pi}(s_t)$$

- Use a neural network $\nu(s; \mathbf{w})$ to estimate $V^{\pi}(s)$

Advantage Actor-Critic Algorithm: Option 2

- Thus estimation of $A^\pi(s_t, a_t)$ is

$$A^\pi(s_t, a_t) \approx r_{t+1} + \gamma V^\pi(s_{t+1}) - V^\pi(s_t)$$

- Use a neural network $\nu(s; \mathbf{w})$ to estimate $V^\pi(s)$:

$$A^\pi(s_t, a_t) \approx r_{t+1} + \gamma \nu(s_{t+1}; \mathbf{w}) - \nu(s_t; \mathbf{w})$$

- TD target and TD error for policy network

$$\begin{aligned} y_t &= r_{t+1} + \gamma \nu(s_{t+1}; \mathbf{w}) \\ \delta_t &= y_t - \nu(s_t; \mathbf{w}) \end{aligned}$$

- Thus $A^\pi(s_t, a_t) \approx \delta_t$

Advantage Actor-Critic Algorithm: Option 2

- Bellman Expectation Equation:

$$V^{\pi}(s_t) = E_{A_t \sim \pi_\theta(\cdot|s_t), S_{t+1} \sim p(\cdot|s_t, A_t)}[R_{t+1} + \gamma V^{\pi}(S_{t+1})].$$

- Suppose we know $(s_t, a_t, r_{t+1}, s_{t+1})$
- Unbiased estimation of $V^{\pi}(s_t)$:

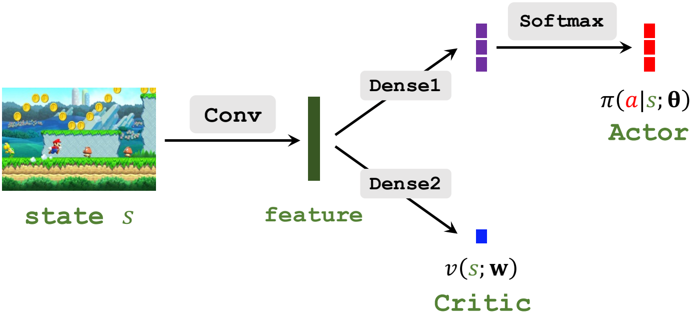
$$V^{\pi}(s_t) \approx r_{t+1} + \gamma V^{\pi}(s_{t+1})$$

- Use a neural network $\nu(s; \mathbf{w})$ to estimate $V^{\pi}(s)$
- TD target and TD error for value network

$$y_t = r_{t+1} + \gamma \nu(s_{t+1}; \mathbf{w})$$
$$\delta_t = y_t - \nu(s_t; \mathbf{w})$$

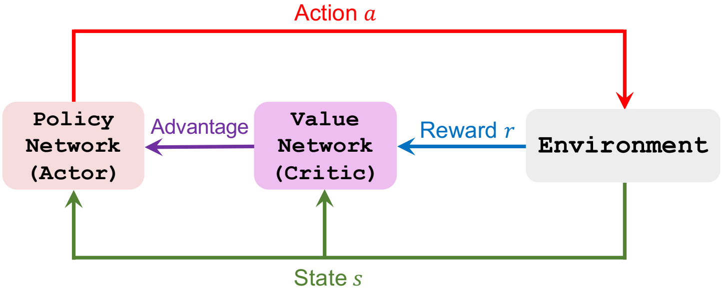
Advantage Actor-Critic Algorithm: Option 2

- Corresponding algorithm is A2C



Advantage Actor-Critic Algorithm: Option 2

- A2C



A2C(Advantage Actor-Critic Algorithm)

given current state s_t, policy network parameters θ, value network parameters $\mathbf{w}$

1. Sample action $a_t \sim \pi_\theta(\cdot|s_t)$, perform action a_t and obtain reward r_{t+1} and new state s_{t+1}.
2. Obtain $\nu(s_t; \mathbf{w})$ and $\nu(s_{t+1}; \mathbf{w})$ from value network
3. Compute the TD target and TD error

$$y_t = r_{t+1} + \gamma \cdot \nu(s_{t+1}; \mathbf{w})$$
$$\delta_t = y_t - \nu(s_t; \mathbf{w})$$

4. Update the parameters of value network as follows:

$$\mathbf{w} \leftarrow \mathbf{w} + \alpha_{\mathbf{w}} \cdot \delta_t \cdot \nabla_{\mathbf{w}} \nu(s_t; \mathbf{w})$$

5. Update the parameters of policy network as follows:

$$\theta \leftarrow \theta + \alpha_\theta \cdot \delta_t \cdot \nabla_\theta \log \pi_\theta(a_t|s_t)$$

A2C with Target Network

given s_t, parameters θ & $\mathbf{w}$, target network parameters $\mathbf{w}^{-1}$

1. Sample action $a_t \sim \pi_\theta(\cdot|s_t)$, perform action a_t and obtain reward r_{t+1} and new state s_{t+1}.
2. Obtain $\nu(s_t; \mathbf{w})$ from value network & $\nu(s_{t+1}; \mathbf{w}^{-1})$ from target network
3. Compute the TD target and TD error

$$y_t = r_{t+1} + \gamma \cdot \nu(s_{t+1}; \mathbf{w}^{-1})$$
$$\delta_t = y_t - \nu(s_t; \mathbf{w})$$

4. Update the parameters of value network as follows:

$$\mathbf{w} \leftarrow \mathbf{w} + \alpha_{\mathbf{w}} \cdot \delta_t \cdot \nabla_{\mathbf{w}} \nu(s_t; \mathbf{w})$$

5. Update the parameters of policy network as follows:

$$\theta \leftarrow \theta + \alpha_\theta \cdot \delta_t \cdot \nabla_\theta \log \pi_\theta(a_t|s_t)$$

6. Every C times update the parameters of policy network as follows:

$$\mathbf{w}^{-1} \leftarrow (1 - \tau) \cdot \mathbf{w}^{-1} + \tau \cdot \mathbf{w}$$

Outline

1. Policy Gradient IV: Policy Gradient Theorem
2. Policy Gradient V: Entropy Regulation
3. Policy Gradient VI: Off-Policy Policy Gradient
4. Reading
5. References

Entropy Regulation

- Encourage exploration and improve robustness
- Given θ and state s, policy $\pi_\theta(a|s)$, $a \in \mathcal{A}$ is a distribution with the corresponding entropy

$$H(s;\theta) = -\sum_{a\in\mathcal{A}} \pi_\theta(a|s) \log \pi_\theta(a|s).$$

- Now with entropy regulation, our policy optimization problem is:

$$\max_\theta J(\theta) + \lambda \cdot E_S\left[H(S;\theta)\right].$$

- $\lambda > 0$ is a hyper-parameter (temperature)
- Previous policy gradient is

$$\nabla_\theta J(\theta) = \mathbb{E}_{S\sim d^\pi, A\sim\pi_\theta(\cdot|S)}[Q^\pi(S,A) \cdot \nabla_\theta \log \pi_\theta(A|S)]$$

- New policy gradient is

$$\begin{aligned}&\nabla_\theta\left[J(\theta) + \lambda \cdot E_S\left[H(S;\theta)\right]\right]\\ =&E_{S\sim d^\pi, A\sim\pi_\theta(\cdot|S)}\left[(Q^\pi(S,A) - \lambda \cdot \log \pi_\theta(A|S) - \lambda) \cdot \nabla_\theta \log \pi_\theta(A|S)\right]\end{aligned}$$

Proof

- First, given any state s,

$$
\begin{aligned}
\frac{\partial H(s;\theta)}{\partial \theta} &= -\sum_{a\in\mathcal{A}} \frac{\partial\left(\pi_\theta(a|s)\log\pi_\theta(a|s)\right)}{\partial\theta} \\
&= -\sum_{a\in\mathcal{A}} \left[\frac{\partial \pi_\theta(a|s)}{\partial\theta}\cdot\log\pi_\theta(a|s) + \pi_\theta(a|s)\cdot\frac{\partial\log\pi_\theta(a|s)}{\partial\theta}\right] \\
&= -\sum_{a\in\mathcal{A}} \left[\pi_\theta(a|s)\cdot\frac{\partial\log\pi_\theta(a|s)}{\partial\theta}\cdot\log\pi_\theta(a|s) + \pi_\theta(a|s)\cdot\frac{\partial\log\pi_\theta(a|s)}{\partial\theta}\right] \\
&= -\sum_{a\in\mathcal{A}} \pi_\theta(a|s)\cdot\frac{\partial\log\pi_\theta(a|s)}{\partial\theta}\cdot\left[\log\pi_\theta(a|s)+1\right] \\
&= -E_{A\sim\pi_\theta(\cdot|s)}\left[\left(\log\pi_\theta(A|s)+1\right)\cdot\frac{\partial\log\pi_\theta(A|s)}{\partial\theta}\right]
\end{aligned}
$$

Proof

- Therefore,

$$\nabla_\theta H(S;\theta) = -E_{A\sim\pi_\theta(\cdot|S)}\left[(\log \pi_\theta(A|S) + 1)\cdot \nabla_\theta \log \pi_\theta(A|S)\right]$$

- Then we have,

$$\nabla_\theta \left[\lambda \cdot E_{S\sim d^\pi}\left[H(S;\theta)\right]\right] = \lambda \cdot E_{S\sim d^\pi}\left[\nabla_\theta H(S;\theta)\right]$$
$$= E_{S\sim d^\pi}\left[E_{A\sim\pi_\theta(\cdot|S)}\left[-\lambda\cdot(\log \pi_\theta(A|S) + 1)\cdot \nabla_\theta \log \pi_\theta(A|S)\right]\right]$$

- Then,

$$\nabla_\theta \left[J(\theta) + \lambda \cdot E_S\left[H(S;\theta)\right]\right] = \nabla_\theta J(\theta) + \nabla_\theta\left[\lambda \cdot E_S\left[H(S;\theta)\right]\right]$$
$$= E_{S,A\sim\pi_\theta(\cdot|S)}\left[Q^\pi(S,A)\cdot \nabla_\theta \log \pi_\theta(A|S)\right]$$
$$+ E_S\left[E_{A\sim\pi_\theta(\cdot|S)}\left[-\lambda\cdot(\log \pi_\theta(A|S) + 1)\cdot \nabla_\theta \log \pi_\theta(A|S)\right]\right]$$
$$= E_{S\sim d^\pi, A\sim\pi_\theta(\cdot|S)}\left[(Q^\pi(S,A) - \lambda\cdot \log \pi_\theta(A|S) - \lambda)\cdot \nabla_\theta \log \pi_\theta(A|S)\right]$$

Entropy Regulation

- Stochastic gradient update:

$$\theta \leftarrow \theta + \alpha \cdot (Q^{\pi}(s,a) - \lambda \cdot \log \pi_\theta(a|s) - \lambda) \cdot \nabla_\theta \log \pi_\theta(a|s)$$

- With baseline $b = b_s = V^{\pi}(s)$ given state s:

$$\theta \leftarrow \theta + \alpha \cdot (Q^{\pi}(s,a) - V^{\pi}(s) - \lambda \cdot \log \pi_\theta(a|s) - \lambda) \cdot \nabla_\theta \log \pi_\theta(a|s)$$

  i.e.,

$$\theta \leftarrow \theta + \alpha \cdot (A^{\pi}(s,a) - \lambda \cdot \log \pi_\theta(a|s) - \lambda) \cdot \nabla_\theta \log \pi_\theta(a|s)$$

A2C with Entropy Regulation

given current state s_t, policy network parameters θ, value network parameters $\mathbf{w}$, hyper-parameter λ

1. Sample action $a_t \sim \pi_\theta(\cdot|s_t)$, perform action a_t and obtain reward r_{t+1} and new state s_{t+1}.
2. Obtain $\nu(s_t; \mathbf{w})$ and $\nu(s_{t+1}; \mathbf{w})$ from value network
3. Compute the TD target and TD error

$$y_t = r_{t+1} + \gamma \cdot \nu(s_{t+1}; \mathbf{w})$$
$$\delta_t = y_t - \nu(s_t; \mathbf{w})$$

4. Update the parameters of value network as follows:

$$\mathbf{w} \leftarrow \mathbf{w} + \alpha_{\mathbf{w}} \cdot \delta_t \cdot \nabla_{\mathbf{w}} \nu(s_t; \mathbf{w})$$

5. Update the parameters of policy network as follows:

$$\theta \leftarrow \theta + \alpha_\theta \cdot (\delta_t - \lambda \cdot \log \pi_\theta(a_t|s_t) - \lambda) \cdot \nabla_\theta \log \pi_\theta(a_t|s_t)$$

Outline

1. Policy Gradient IV: Policy Gradient Theorem
2. Policy Gradient V: Entropy Regulation
3. Policy Gradient VI: Off-Policy Policy Gradient
4. Reading
5. References

On-Policy & Off-Policy RL

- Until now, policy gradient and actor-critic methods are on-policy
- Training samples are collected according to the target policy — the very same policy that we try to optimize for:

$$\nabla_\theta J(\theta) = \mathbb{E}_{S \sim d^\pi, A \sim \pi_\theta(\cdot|S)}[Q^\pi(S, A) \cdot \nabla_\theta \log \pi_\theta(A|S)]$$

- However, on-policy RL has a low sample efficiency
- The off-policy approach does not require full trajectories and can reuse any past episodes ("experience replay") for much better sample efficiency.
- The sample collection follows a behavior policy different from the target policy, bringing better exploration.

Off-Policy Learning using Importance Sampling

- The behavior policy $\beta(a|s)$ for collecting samples is a known policy (predefined just like a hyperparameter)
- Object function sums up the reward over the state distribution defined by this behavior policy

$$\begin{aligned} J(\theta) &= \sum_{s\in\mathcal{S}} d^{\beta}(s) V^{\pi}(s) \\ &= \sum_{s\in\mathcal{S}} d^{\beta}(s) \sum_{a\in\mathcal{A}} Q^{\pi}(s,a)\pi_{\theta}(a|s) \\ &= \mathbb{E}_{S\sim d^{\beta}}\Big[\sum_{a\in\mathcal{A}} Q^{\pi}(S,a)\pi_{\theta}(a|S)\Big] \end{aligned}$$

- $d^{\beta}(s) = \lim_{t\to\infty} P(S_t = s|S_0, \beta)$ is the stationary distribution of the behavior policy $\beta(a|s)$
- $Q^{\pi}(s,a)$ is the action-value function estimated with regard to the target policy π (not the behavior policy!)

Off-Policy Learning using Importance Sampling

- The gradient of $J(\theta)$ is

$$\begin{aligned}
\nabla_\theta J(\theta) &= \nabla_\theta \mathbb{E}_{S\sim d^\beta}\Big[\sum_{a\in\mathcal{A}} Q^\pi(S,a)\pi_\theta(a|S)\Big] \\
&= \mathbb{E}_{S\sim d^\beta}\Big[\sum_{a\in\mathcal{A}} \big(Q^\pi(S,a)\nabla_\theta\pi_\theta(a|S) + \pi_\theta(a|S)\nabla_\theta Q^\pi(S,a)\big)\Big] \\
&\overset{(i)}{\approx} \mathbb{E}_{S\sim d^\beta}\Big[\sum_{a\in\mathcal{A}} Q^\pi(S,a)\nabla_\theta\pi_\theta(a|S)\Big] \\
&= \mathbb{E}_{S\sim d^\beta}\Big[\sum_{a\in\mathcal{A}} \beta(a|S)\frac{\pi_\theta(a|S)}{\beta(a|S)} Q^\pi(S,a)\frac{\nabla_\theta\pi_\theta(a|S)}{\pi_\theta(a|S)}\Big] \\
&= \mathbb{E}_{S\sim d^\beta, A\sim\beta(\cdot|S)}\Big[\frac{\pi_\theta(A|S)}{\beta(A|S)} Q^\pi(S,A)\nabla_\theta \ln \pi_\theta(A|S)\Big]
\end{aligned}$$

- Ignore the red part, we still guarantee the policy improvement and eventually achieve the true local minimum. (proved in Degris, White & Sutton, 2012)
- The blue part is the importance weight

Off-Policy Learning with Baselines

- To further reduce the variance, we choose the baseline $b = b_s = V^\pi(s)$ for any state s
- The gradient of $J(\theta)$ does not change

$$
\begin{aligned}
\nabla_\theta J(\theta) &= \mathbb{E}_{S\sim d^\beta, A\sim\beta(\cdot|S)}\left[\frac{\pi_\theta(A|S)}{\beta(A|S)}(Q^\pi(S,A) - V^\pi(S))\nabla_\theta \ln \pi_\theta(A|S)\right] \\
&= \mathbb{E}_{S\sim d^\beta, A\sim\beta(\cdot|S)}\left[\frac{\pi_\theta(A|S)}{\beta(A|S)}A^\pi(S,A)\nabla_\theta \ln \pi_\theta(A|S)\right]
\end{aligned}
$$

- Stochastic gradient update with samples a, s from behavior policy β:

$$
\theta \leftarrow \theta + \alpha \cdot \frac{\pi_\theta(a|s)}{\beta(a|s)} \cdot A^\pi(s,a) \cdot \nabla_\theta \log \pi_\theta(a|s)
$$

Outline

1. Policy Gradient IV: Policy Gradient Theorem
2. Policy Gradient V: Entropy Regulation
3. Policy Gradient VI: Off-Policy Policy Gradient
4. Reading
5. References

Proof of Policy-Gradient Theorem

- The reward function is defined as $J(\theta) = J_1(\theta) = V^{\pi}(s_0)$
- $d^{\pi}(s)$ is the stationary distribution of Markov chain for π_θ (on-policy state distribution under policy π)
- For simplicity, the parameter θ would be omitted for the policy π_θ when the policy is present in the subscript of other functions

Proof of Policy-Gradient Theorem

- First start with the derivative of the state value function:

$$
\begin{aligned}
&\nabla_\theta V^\pi(s) \\
=&\nabla_\theta \Big(\sum_{a \in \mathcal{A}} \pi_\theta(a|s) Q^\pi(s,a) \Big) \\
=&\sum_{a \in \mathcal{A}} \Big(\nabla_\theta \pi_\theta(a|s) Q^\pi(s,a) + \pi_\theta(a|s) \nabla_\theta Q^\pi(s,a) \Big) \\
=&\sum_{a \in \mathcal{A}} \Big(\nabla_\theta \pi_\theta(a|s) Q^\pi(s,a) + \pi_\theta(a|s) \nabla_\theta \sum_{s',r} P(s',r|s,a)(r + V^\pi(s')) \Big) \\
=&\sum_{a \in \mathcal{A}} \Big(\nabla_\theta \pi_\theta(a|s) Q^\pi(s,a) + \pi_\theta(a|s) \sum_{s',r} P(s',r|s,a) \nabla_\theta V^\pi(s') \Big) \\
=&\sum_{a \in \mathcal{A}} \Big(\nabla_\theta \pi_\theta(a|s) Q^\pi(s,a) + \pi_\theta(a|s) \sum_{s'} P(s'|s,a) \nabla_\theta V^\pi(s') \Big)
\end{aligned}
$$

Proof of Policy-Gradient Theorem

- Now we have a nice recursive equation

$$\nabla_\theta V^\pi(s) = \sum_{a \in \mathcal{A}} \Big(\nabla_\theta \pi_\theta(a|s) Q^\pi(s,a) + \pi_\theta(a|s) \sum_{s'} P(s'|s,a) \nabla_\theta V^\pi(s') \Big)$$

- Consider the following visitation sequence and label the probability of transitioning from state s to state x with policy π_θ after k steps as $\rho^\pi(s \to x, k)$:

$$s \xrightarrow{a \sim \pi_\theta(.|s)} s' \xrightarrow{a \sim \pi_\theta(.|s')} s'' \xrightarrow{a \sim \pi_\theta(.|s'')} \dots$$

- when $k = 0$, $\rho^\pi(s \to s, k = 0) = 1$
- when $k = 1$, we scan through all possible actions and sum up the transition probabilities to the target state:

$$\rho^\pi(s \to s', k = 1) = \sum_a \pi_\theta(a|s) P(s'|s,a)$$

Proof of Policy-Gradient Theorem

- Imagine that the goal is to go from state s to x after $k+1$ steps while following policy π_θ.
- We can first travel from s to a middle point s' (any state can be a middle point, $s' \in \mathcal{S}$)
- After k steps and then go to the final state x during the last step. In this way, we are able to update the visitation probability recursively:

$$\rho^\pi(s \to x, k+1) = \sum_{s'} \rho^\pi(s \to s', k)\rho^\pi(s' \to x, 1).$$

- This is the Chapman - Kolmogorov equation!
- To simplify the maths, we denote

$$\phi(s) = \sum_{a \in \mathcal{A}} \nabla_\theta \pi_\theta(a|s) Q^\pi(s, a)$$

Proof of Policy-Gradient Theorem

- Then we go back to unroll the recursive equation of $\nabla_\theta V^\pi(s)$

$$
\begin{aligned}
&\nabla_\theta V^\pi(s) \\
=&\phi(s) + \sum_a \pi_\theta(a|s) \sum_{s'} P(s'|s,a) \nabla_\theta V^\pi(s') \\
=&\phi(s) + \sum_{s'} \sum_a \pi_\theta(a|s) P(s'|s,a) \nabla_\theta V^\pi(s') \\
=&\phi(s) + \sum_{s'} \rho^\pi(s \to s', 1) \nabla_\theta V^\pi(s') \\
=&\phi(s) + \sum_{s'} \rho^\pi(s \to s', 1) \sum_{a \in \mathcal{A}} \Big(\nabla_\theta \pi_\theta(a|s') Q^\pi(s', a) \\
&+ \pi_\theta(a|s') \sum_{s''} P(s''|s', a) \nabla_\theta V^\pi(s'') \Big)
\end{aligned}
$$

Proof of Policy-Gradient Theorem

- Then we go back to unroll the recursive equation of $\nabla_\theta V^\pi(s)$

$$
\begin{aligned}
&\nabla_\theta V^\pi(s) \\
=&\phi(s) + \sum_{s'} \rho^\pi(s \to s', 1)[\phi(s') + \sum_{s''} \rho^\pi(s' \to s'', 1) \nabla_\theta V^\pi(s'')] \\
=&\phi(s) + \sum_{s'} \rho^\pi(s \to s', 1)\phi(s') + \sum_{s''} \rho^\pi(s \to s'', 2) \nabla_\theta V^\pi(s'') \\
=&\phi(s) + \sum_{s'} \rho^\pi(s \to s', 1)\phi(s') + \sum_{s''} \rho^\pi(s \to s'', 2) \phi(s'') \\
&+ \sum_{s'''} \rho^\pi(s \to s''', 3) \nabla_\theta V^\pi(s''') \\
=&\ldots; \text{ Repeatedly unrolling the part of } \nabla_\theta V^\pi(.) \\
=&\sum_{x \in \mathcal{S}} \sum_{k=0}^{\infty} \rho^\pi(s \to x, k)\phi(x)
\end{aligned}
$$

Proof of Policy-Gradient Theorem

- Then we have

$$
\begin{aligned}
\nabla_\theta J(\theta) &= \nabla_\theta J_1(\theta) = \nabla_\theta V^\pi(s_0) \\
&= \sum_{s\in\mathcal{S}} \sum_{k=0}^{\infty} \rho^\pi(s_0 \to s, k)\phi(s) = \sum_{s\in\mathcal{S}} d^\pi(s)\phi(s) \\
&= \sum_{s} d^\pi(s) \sum_{a} \nabla_\theta \pi_\theta(a|s) Q^\pi(s,a) \\
&= \sum_{s\in\mathcal{S}} d^\pi(s) \sum_{a\in\mathcal{A}} \pi_\theta(a|s) Q^\pi(s,a) \frac{\nabla_\theta \pi_\theta(a|s)}{\pi_\theta(a|s)} \\
&= \sum_{s\in\mathcal{S}} d^\pi(s) \sum_{a\in\mathcal{A}} \pi_\theta(a|s) Q^\pi(s,a) \nabla_\theta \log \pi_\theta(a|s) \\
&= \mathbb{E}_{S\sim d^\pi, A\sim\pi_\theta(\cdot|S)}[Q^\pi(S,A)\cdot\nabla_\theta \log \pi_\theta(A|S)]
\end{aligned}
$$

Outline

1 Policy Gradient IV: Policy Gradient Theorem

2 Policy Gradient V: Entropy Regulation

3 Policy Gradient VI: Off-Policy Policy Gradient

4 Reading

5 References

Main References

- Reinforcement Learning: An Introduction (second edition), R. Sutton & A. Barto, 2018.
- RL course slides from Richard Sutton, University of Alberta.
- RL course slides from David Silver, University College London.
- RL course slides from Sergey Levine, UC Berkeley
- RL course slides from Shusen Wang