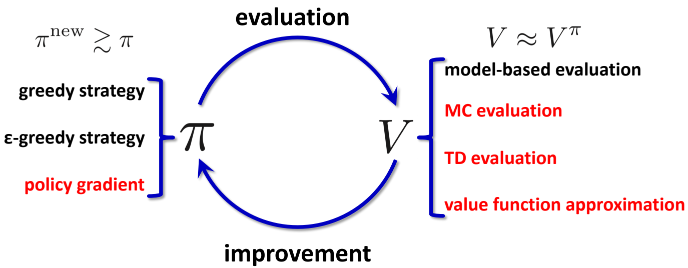
Algorithmic and Theoretical Foundations of RL

Policy Optimization I

Ke Wei
School of Data Science
Fudan University

Table of Contents

Value-Based RL vs Policy-Based RL



- Value-based RL: Learn optimal values and policy is implicitly inferred;
- Policy-based RL: Parametrize policy and conduct search in policy space.

Policy-Based RL

Consider a policy parameterization (which is essentially about how to represent a distribution) such that :

$$\pi_\theta(\cdot|s) \text{ defines a probability distribution on } \mathcal{A}.$$

Note that once θ is given, policy is determined.

Goal: Search for best θ subject to certain performance measure.

Typical advantages of policy-based methods include:

- Better convergence properties
- Effective in high dimensional or continuous action spaces
- Can learn stochastic policies

Policy Parameterizations

- Discrete action space
  - Simplex parameterization

$$\pi_\theta(a|s) = \theta_{s,a} \quad \text{subject to} \quad \theta_{s,a} \geq 0 \text{ and } \sum_a \theta_{s,a} = 1.$$

  - Softmax parameterization

$$\pi_\theta(a|s) = \frac{\exp(f_\theta(s,a))}{\sum_{a' \in \mathcal{A}} \exp(f_\theta(s,a'))}.$$

- Continuous action space: Gaussian parameterization

$$\pi_\theta(\cdot|s) \text{ is the pdf of } \mathcal{N}(\mu_\theta(s), \sigma_\theta^2(s)).$$

Policy Optimization

Consider average state value with initial distribution μ as performance measure:

$$V^{\pi_\theta}(\mu) = \mathbb{E}_{s_0 \sim \mu}\left[V^{\pi_\theta}(s_0)\right] = \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}}\left[r(\tau)\right],$$

where given $\tau = (s_t, a_t, r_t)_{t=0}^{\infty}$,

$$P_\mu^{\pi_\theta}(\tau) = \mu(s_0) \prod_{t=0}^{\infty} \pi_\theta(a_t|s_t) P(s_{t+1}|s_t, a_t) \quad \text{and} \quad r(\tau) = \sum_{t=0}^{\infty} \gamma^t r_t.$$

It is natural to formulate RL as

$$\theta^* = \operatorname*{argmax}_\theta V^{\pi_\theta}(\mu).$$

- Initial state distribution can be for example Dirac delta distribution, uniform distribution, or stationary distribution under policy π_θ.

For simplicity, we only discuss the case where sate and action spaces are discrete.

Alternative Expression of State Value and Visitation Measure

Theorem 1 (Expression of State Value in Terms of Visitation Measure)

For any policy π, there holds

$$V^{\pi}(\mu) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{\mu}^{\pi}}\mathbb{E}_{a\sim\pi(\cdot|s)}\mathbb{E}_{s'\sim P(\cdot|s,a)}\left[r(s,a,s')\right],$$

where

$$d_{\mu}^{\pi}(s) = \mathbb{E}_{s_0\sim\mu}\left[d_{s_0}^{\pi}(s)\right] = \mathbb{E}_{s_0\sim\mu}\left[(1-\gamma)\sum_{t=0}^{\infty}\gamma^{t}P(s_t = s|s_0,\pi)\right]$$

is discounted state visitation measure under policy π and initial distribution μ.

Proof of Theorem 1

$$
\begin{aligned}
V^{\pi}(s_0) &= \mathbb{E}_{\tau \sim P^{\pi}_{s_0}} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t, s_{t+1}) \right] \\
&= \sum_{t=0}^{\infty} \mathbb{E}_{\tau \sim P^{\pi}_{s_0}} \left[\gamma^t r(s_t, a_t, s_{t+1}) \right] \\
&= \sum_{t=0}^{\infty} \sum_{s} \sum_{a} \sum_{s'} \gamma^t \cdot P(s_t = s | s_0, \pi) \pi(a|s) P(s'|s,a) r(s,a,s') \\
&= \sum_{s} \sum_{a} \sum_{s'} \left(\sum_{t=0}^{\infty} \gamma^t \cdot P(s_t = s | s_0, \pi) \right) \pi(a|s) P(s'|s,a) r(s,a,s') \\
&= \frac{1}{1-\gamma} \sum_{s} \sum_{a} \sum_{s'} d^{\pi}_{s_0}(s) \pi(a|s) P(s'|s,a) r(s,a,s').
\end{aligned}
$$

Expression for $\mathbb{E}_{s_0 \sim \mu}\left[V^{\pi}(s_0)\right]$ can be obtained directly by averaging over $s_0 \sim \mu$.

More on Visitation Measure

Lemma 1

The visitation measure can be expressed in the following matrix form

$$d_\mu^\pi = (1-\gamma)(I - \gamma (P^\pi)^T)^{-1}\mu,$$

where $P^\pi = (p_{ss'}^\pi)$ is transition matrix induced by policy π (see Lecture 1).

Proof. This lemma can be proved by expanding $(I - \gamma (P^\pi)^T)^{-1}$.

- Using the matrix form of visitation measure, it is evident that the alternative expression of state value in terms of visitation measure is indeed

$$V^\pi = (I - \gamma P^\pi)^{-1} r^\pi.$$

Performance Difference Lemma

Given a policy π, the advantage function is defined as

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

which measures how well a single action is compared with average state value.

Lemma 2 (Performance Difference Lemma)

For any two policies π_1, π_2, one has

$$V^{\pi_1}(\mu) - V^{\pi_2}(\mu) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{\mu}^{\pi_1}}\left[\mathbb{E}_{a\sim\pi_1(\cdot|s)}\left[A^{\pi_2}(s,a)\right]\right].$$

Proof of Lemma 2

Recall from Lecture 1 that

$$V^{\pi_1} - V^{\pi_2} = (I - \gamma P^{\pi_1})^{-1}(\mathcal{T}^{\pi_1} V^{\pi_2} - V^{\pi_2}).$$

It follows that

$$\begin{aligned}
V^{\pi_1}(\mu) - V^{\pi_2}(\mu) &= \mu^T (V^{\pi_1} - V^{\pi_2}) \\
&= (\mathcal{T}^{\pi_1} V^{\pi_2} - V^{\pi_2})^T (I - \gamma (P^{\pi_1})^T)^{-1} \mu \\
&= \frac{1}{1-\gamma} (\mathcal{T}^{\pi_1} V^{\pi_2} - V^{\pi_2})^T d_\mu^{\pi_1} \\
&= \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_\mu^{\pi_1}} \left[\mathbb{E}_{a \sim \pi_1(\cdot|s)} \left[A^{\pi_2}(s,a) \right] \right],
\end{aligned}$$

where the third line follows from Lemma 1 and the last line follows from the fact

$$\mathcal{T}^{\pi_1} V^{\pi_2} - V^{\pi_2} = \mathbb{E}_{a \sim \pi_1(\cdot|s)} \left[A^{\pi_2}(s,a) \right].$$

Optimization Methods

- Gradient free methods
  - Random search
  - Simulated annealing
  - Various evolutionary algorithms
- Gradient ascent methods
  - Compute gradient by finite difference
  - Compute gradient analytically

For gradient free methods, see for example Chapter 10 of “Algorithms for decision making” by Kochenderfer et al., 2022.

Policy Gradient Theorem

Theorem 2 (Policy Gradient Theorem)

Recalling the definition of visitation measure, we have

$$\begin{aligned}
\nabla_\theta V^{\pi_\theta}(\mu) &= \mathbb{E}_{\tau\sim P_\mu^{\pi_\theta}}\left[\sum_{t=0}^{\infty}\gamma^t Q^{\pi_\theta}(s_t,a_t)\nabla_\theta \log \pi_\theta(a_t|s_t)\right] \\
&= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_\mu^{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\left[Q^{\pi_\theta}(s,a)\nabla_\theta \log \pi_\theta(a|s)\right].
\end{aligned}$$

- Policy gradient theorem expresses policy gradient as a weighted average of $\nabla_\theta \log \pi_\theta(a|s)$ over all state-action pairs. Note that $\nabla_\theta \log \pi_\theta(a|s)$ is direction that $\pi_\theta(a|s)$ increases (i.e., probability of selecting a at s increases).

Proof of Theorem 2

A direct calculation yields that

$$
\begin{aligned}
\nabla_\theta V^{\pi_\theta}(s_0) &= \nabla_\theta \left(\mathbb{E}_{a_0\sim\pi_\theta(\cdot|s_0)}\left[Q^{\pi_\theta}(s_0,a_0)\right]\right) \\
&= \mathbb{E}_{a_0\sim\pi_\theta(\cdot|s_0)}\left[Q^{\pi_\theta}(s_0,a_0)\nabla\log\pi_\theta(a_0|s_0) + \nabla_\theta Q^{\pi_\theta}(s_0,a_0)\right] \\
&= \mathbb{E}_{a_0\sim\pi_\theta(\cdot|s_0)}\left[Q^{\pi_\theta}(s_0,a_0)\nabla\log\pi_\theta(a_0|s_0) + \gamma\mathbb{E}_{s_1}\left[\nabla_\theta V^{\pi_\theta}(s_1)\right]\right] \\
&= \cdots \\
&= \mathbb{E}_{\tau\sim P_{s_0}^{\pi_\theta}}\left[\sum_{t=0}^{\infty}\gamma^t Q^{\pi_\theta}(s_t,a_t)\nabla_\theta\log\pi_\theta(a_t|s_t)\right].
\end{aligned}
$$

Average over $s_0 \sim \mu$ completes the proof.

Policy Gradient Ascent

$$\theta \leftarrow \theta + \alpha \cdot \mathbb{E}_{s,a}\left[Q^{\pi_\theta}(s,a)\nabla_\theta \log \pi_\theta(a|s)\right]$$
$$= \theta + \alpha \cdot \mathbb{E}_{s,a}\left[\frac{Q^{\pi_\theta}(s,a)}{\pi_\theta(a|s)}\nabla_\theta \pi_\theta(a|s)\right]$$

- Large $Q^{\pi_\theta}(s,a)$ means that weight in front of the direction $\nabla_\theta \pi_\theta(a|s)$ is large. Thus, the method attempts to exploit actions with large action values.
- Small $\pi_\theta(a|s)$ means that weight in front of the direction $\nabla_\theta \pi_\theta(a|s)$ is large. This reflects that the method attempts to explore actions with low probability.
- Policy gradient method also fits into the framework of policy evaluation and policy improvement, where policy evaluation affects direction to improve the policy and policy improvement is achieved by updating policy parameter. Thus, analysis of policy gradient methods often boils down to analysis of improvement ability in policy domain.

Policy Gradient in Terms of Advantage Function

Theorem 3 (Policy Gradient in Terms of Advantage Function)

We have

$$\nabla_\theta V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_\mu^{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\left[A^{\pi_\theta}(s,a)\nabla_\theta \log \pi_\theta(a|s)\right],$$

provided $\sum_a \pi_\theta(a|s) = 1$ *for any* θ.

Proof. The result follows from the fact

$$\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\left[\nabla_\theta \log \pi_\theta(a|s)\right] = \nabla_\theta\Big(\sum_a \pi_\theta(a|s)\Big) = 0.$$

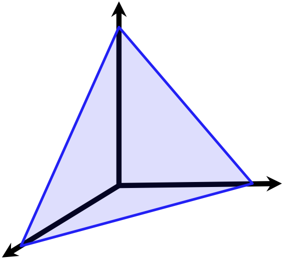
Remark

- Condition $\sum_a \pi_\theta(a|s) = 1$ is necessary for advantage expression of policy gradient. Note that simplex parameterization does not meet this condition.
- For parameterization that $\sum_a \pi_\theta(a|s) = 1$ is not satisfied, policy gradient expression in terms of action value is the gradient of the extended function based on visitation measure expression rather than trajectory expression. For example, the trajectory expression may diverge out of probability simplex.

Table of Contents

Projected Policy Gradient (PPG)

Policy Gradient



Parameter at each s:

$$\Delta = \left\{\pi_s \in \mathbb{R}^{|\mathcal{A}|} : \sum_{a=1}^{|\mathcal{A}|} \pi_{s,a} = 1\right\}$$

Simplex parameterization ($\pi_\theta(a|s) = \pi_{s,a}$):

$$\Pi = \left\{\pi = (\pi_s)_{s\in\mathcal{S}} \mid \pi_s \in \Delta \text{ for all } s \in \mathcal{S}\right\}$$

Lemma 3
The policy gradient under simplex parameterization is given by

$$\nabla_{\pi_s} V^\pi(\mu) = \frac{d_\mu^\pi(s)}{1-\gamma} Q^\pi(s,\cdot)\,.$$

Proof. This result follows directly from Theorem 2 by noting that

$$\frac{\partial \log \pi_{s',a'}}{\partial \pi_{s,a}} = \frac{1}{\pi_{s,a}} 1_{[s'=s,a'=a]}.$$

Projected Policy Gradient (PPG)

Algorithm

PPG updates policy as follows:

$$\pi^{k+1} = \arg\min_{\pi\in\Pi}\Big\{ -\eta_k\langle \nabla_\pi V^\pi(\mu)\,|_{\pi=\pi^k}, \pi - \pi^k\rangle + \frac{1}{2}\|\pi - \pi^k\|_2^2\Big\},$$
$$= \arg\min_{\pi\in\Pi}\Big\{ \sum_{s\in\mathcal{S}} \Big(-\eta_k\langle \nabla_{\pi_s} V^\pi(\mu)\,|_{\pi=\pi^k}, \pi_s - \pi_s^k\rangle + \frac{1}{2}\|\pi_s - \pi_s^k\|_2^2\Big)\Big\}.$$

Explicitly, one has

$$\pi_s^{k+1} = \mathrm{Proj}_\Delta\Big(\pi_s^k + \eta_k \nabla_{\pi_s} V^\pi(\mu)\,|_{\pi=\pi^k}\Big)$$
$$= \mathrm{Proj}_\Delta\Big(\pi_s^k + \frac{\eta_k d_\mu^k(s)}{1-\gamma} Q^k(s,\cdot)\Big), \quad \forall s\in\mathcal{S},$$

where d_μ^k, $Q^k(s,\cdot)$ are short for $d_\mu^{\pi^k}$, $Q^{\pi^k}(s,\cdot)$, respectively.

Projected Policy Gradient (PPG)

Convergence Results

- PPG converges at a sublinear rate $O(1/k)$ for any constant step size;
- With increasing adaptive step sizes, PPG converges linearly;
- PPG achieves exact convergence in a finite number of iterations.

"Projected Policy Gradient Converges in a Finite Number of Iterations" by Jiacai Liu, Wenye Li, and Ke Wei, 2023.

Softmax Policy Gradient (Softmax PG)

Policy Gradient

Softmax parameterization ($\theta = (\theta_s)$, where $\theta_s = (\theta_{s,a}) \in \mathbb{R}^{|\mathcal{A}|}$):

$$\pi_\theta(a|s) = \frac{\exp(\theta_{s,a})}{\sum_{\tilde{a}} \exp(\theta_{s,\tilde{a}})}.$$

It is evident that $\sum_a \pi_\theta(a|s) = 1$ holds for all θ.

Lemma 4

The policy gradient under softmax parameterization is given by

$$\nabla_{\theta_s} V^{\pi_\theta}(\mu) = \frac{d_\mu^{\pi_\theta}(s)}{1-\gamma} \pi_\theta(\cdot|s) A^{\pi_\theta}(s, \cdot).$$

Proof of Lemma 4

First it is not hard to see that

$$\frac{\partial \log \pi_\theta(a'|s')}{\partial \theta_{s,a}} = 1_{[s'=s]}(1_{[a'=a]} - \pi_\theta(a|s)).$$

Plugging this into Theorem 3 yields

$$\begin{aligned}
\frac{\partial V^{\pi_\theta}(\mu)}{\partial \theta_{s,a}} &= \frac{1}{1-\gamma}\mathbb{E}_{s'\sim d_\mu^{\pi_\theta}}\mathbb{E}_{a'\sim\pi_\theta(\cdot|s')}\left[A^{\pi_\theta}(s',a')\frac{\partial \log \pi_\theta(a'|s')}{\partial \theta_{s,a}}\right] \\
&= \frac{d_\mu^{\pi_\theta}(s)}{1-\gamma}\mathbb{E}_{a'\sim\pi_\theta(\cdot|s)}\left[A^{\pi_\theta}(s,a')\big(1_{[a'=a]} - \pi_\theta(a|s)\big)\right] \\
&= \frac{d_\mu^{\pi_\theta}(s)}{1-\gamma}\pi_\theta(a|s)A^{\pi_\theta}(s,a) - \frac{d_\mu^{\pi_\theta}(s)}{1-\gamma}\pi_\theta(a|s)\mathbb{E}_{a'\sim\pi_\theta(\cdot|s)}[A^{\pi_\theta}(s,a')] \\
&= \frac{d_\mu^{\pi_\theta}(s)}{1-\gamma}\pi_\theta(a|s)A^{\pi_\theta}(s,a),
\end{aligned}$$

where the last equality follows from the fact $\mathbb{E}_{a'\sim\pi_\theta(\cdot|s)}[A^{\pi_\theta}(s,a')] = 0$.

Softmax Policy Gradient (Softmax PG)

Algorithm and Convergence Result

Softmax PG updates policy parameter as follows:

$$\theta_s^{k+1} = \theta_s^k + \frac{\eta_k d_\mu^k(s)}{1-\gamma} \pi_k(\cdot|s) A^k(s,\cdot), \quad \forall s \in \mathcal{S},$$

where π_k, d_μ^k, $A^k(s,\cdot)$ are short for π_{θ^k}, $d_\mu^{\pi_{\theta^k}}$, $A^{\pi_{\theta^k}}(s,\cdot)$, respectively.

- Softmax converges at a sublinear rate $O(1/k)$ for small constant step size.

“On the Global Convergence Rates of Softmax Policy Gradient Methods” by Jincheng Mei et al., 2020.

Policy Mirror Ascent (PMA)

Algorithm

Recall that $\Pi = \left\{\pi = (\pi_s)_{s\in\mathcal{S}} \mid \pi_s \in \Delta \text{ for all } s \in \mathcal{S}\right\}$. In contrast to PPG, PMA uses visitation measure re-weighted Bregman divergence as regularizer:

$$\pi^{k+1} = \underset{\pi\in\Pi}{\arg\min}\left\{ \sum_{s\in\mathcal{S}} \left(-\eta_k \langle \nabla_{\pi_s} V^{\pi}(\mu)\,|_{\pi=\pi^k}, \pi_s - \pi_s^k\rangle + \frac{d_\mu^k(s)}{1-\gamma}\cdot D_h(\pi_s, \pi_s^k)\right)\right\},$$

where $D_h(p, p')$ is Bregman divergence defined through a function h as follows:

$$D_h(p', p) = h(p') - h(p) - \langle \nabla h(p), p' - p\rangle.$$

More explicitly, one has

$$\pi_s^{k+1} = \underset{\pi_s\in\Delta}{\operatorname{argmin}}\left\{ -\eta_k\langle Q^k(s,\cdot), \pi_s - \pi_s^k\rangle + D_h(\pi_s, \pi_s^k)\right\}, \quad \forall s \in \mathcal{S}.$$

Policy Mirror Ascent (PMA)

Particular Example: Projected Q-Ascent (PQA)

When $h(p) = \frac{1}{2}\|p\|_2^2$, one has

$$D_h(p', p) = \frac{1}{2}\|p'\|_2^2 - \frac{1}{2}\|p\|_2^2 - \langle p, p' - p\rangle = \frac{1}{2}\|p' - p\|_2^2.$$

In this case, PMA reduces to PQA:

$$\begin{aligned}\pi_s^{k+1} &= \operatorname*{argmin}_{\pi_s \in \Delta} \Big\{ -\eta_k \langle Q^k(s,\cdot), \pi_s - \pi_s^k \rangle + \frac{1}{2}\|\pi_s - \pi_s^k\|_2^2 \Big\} \\ &= \mathrm{Proj}_\Delta\Big(\pi_s^k + \eta_k Q^k(s,\cdot)\Big), \quad \forall s \in \mathcal{S},\end{aligned}$$

which can be viewed as a preconditioned version of PPG.

Policy Mirror Ascent (PMA)

Particular Example: Exponentiated Q-Ascent (EQA)

When $h(p) = \sum_a p_a \log p_a$, one has

$$
\begin{aligned}
D_h(p', p) &= \sum_a p'_a \log p'_a - \sum_a p_a \log p_a - \sum_a (\log p_a + 1)(p'_a - p_a) \\
&= \sum_a p'_a \log \frac{p'_a}{p_a},
\end{aligned}
$$

which is KL divergence between two probability vectors. In this case, PMA reduces to EQA (can be derived using supplement lemma given in next slide):

$$
\pi^{k+1}_{s,a} = \pi^k_{s,a} \cdot \frac{\exp(\eta_k Q^k(s,a))}{Z^k_s},
$$

where Z^k_s is a normalized constant such that $\sum_a \pi^{k+1}_{s,a} = 1$. It is worth noting EQA coincides with natural policy gradient method (NPG, discussed in next lecture) under softmax parameterization in policy space.

EQA: Supplement Lemma

Lemma 5

Given any vector x, the solution to the optimization problem

$$\min_{p\in\Delta} \sum_a p_a(\log p_a - x_a)$$

is given by $p_a^* = \frac{\exp(x_a)}{\sum_{a'}\exp(x_{a'})}$. *Moreover, one has* $\log p_a^* - x_a = \log p_{a'}^* - x_{a'}$ *for* $a \neq a'$.

Proof. The first claim can be proved via KKT condition or by writing the objective function into a KL divergence, and the second claim can be verified directly.

Policy Mirror Ascent (PMA)

Convergence Results

- PPG converges at a sublinear rate $O(1/k)$ for any constant step size;
- With increasing adaptive step sizes, PPG converges linearly;
- Similar to PPG, PQA indeed converges in a finite number of iterations.

"On the Convergence Rates of Policy Gradient Methods" by Lin Xiao, 2022;
"Projected Policy Gradient Converges in a Finite Number of Iterations" by Jiacai Liu, Wenye Li, and Ke Wei, 2023.

Remark

The updates of PPG, PQA and EQA can be unified into the following form:

$$\begin{aligned}
\pi_s^+ &= \operatorname*{arg\,min}_{\tilde{\pi}_s \in \Delta} \left\{ -\eta_s \left\langle Q^\pi(s, \cdot), \tilde{\pi}_s - \pi_s \right\rangle + D_h(\tilde{\pi}_s, \pi_s) \right\} \\
&= \operatorname*{arg\,min}_{\tilde{\pi}_s \in \Delta} \left\{ -\eta_s \left\langle Q^\pi(s, \cdot), \tilde{\pi}_s \right\rangle + D_h(\tilde{\pi}_s, \pi_s) \right\}.
\end{aligned}$$

As $\eta_s \to \infty$, one has $\pi_s^+ \approx \operatorname*{arg\,min}_{\tilde{\pi}_s \in \Delta} \left\{ -\eta_s \left\langle Q^\pi(s, \cdot), \tilde{\pi}_s \right\rangle \right\}$, which is indeed PI update.

- There exists a finite threshold on η_s for PPG and PQA to be equivalent to PI. The equivalence of EQA to PI when $\eta_s \to \infty$ can also be observed from its explicit update rule.

Table of Contents

MC Evaluation of Policy Gradient

The expectation in policy gradient expression requires MC evaluation.

- Sample N episodes:

$$\tau^{(i)} = (s_0^{(i)}, a_0^{(i)}, r_0^{(i)}, \cdots, s_{T-1}^{(i)}, a_{T-1}^{(i)}, r_{T-1}^{(i)}, s_T^{(i)}) \sim \pi_\theta;$$

- Use return $G_t = \sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'}$ as an unbiased estimate of $Q^{\pi_\theta}(s_t, a_t)$:

$$\nabla J(\theta) \approx \frac{1}{N} \sum_{i=1}^{N} \sum_{t=0}^{T-1} \gamma^t G_t^{(i)} \nabla_\theta \log \pi_\theta(a_t^{(i)} | s_t^{(i)}).$$

REINFORCE

Algorithm 1: REINFORCE

Initialization: $\pi_\theta(a|s)$ and θ_0.

for $k = 0, 1, 2, \ldots$ **do**

Sample episodes $\mathcal{D}_k = \{\tau^{(i)}\}$:

$$\tau^{(i)} = (s_0^{(i)}, a_0^{(i)}, r_0^{(i)}, \cdots, s_{T-1}^{(i)}, a_{T-1}^{(i)}, r_{T-1}^{(i)}, s_T^{(i)}) \sim \pi_{\theta_k}$$

Policy gradient calculation:

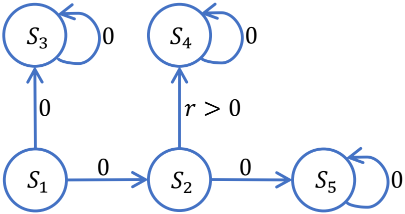
$$g_k = \frac{1}{|\mathcal{D}_k|} \sum_{i=1}^{|\mathcal{D}_k|} \sum_{t=0}^{T-1} \gamma^t G_t^{(i)} \nabla_\theta \log \pi_{\theta_k}(a_t^{(i)}|s_t^{(i)})$$

Policy parameter update:

$$\theta_{k+1} = \theta_k + \alpha_k g_k$$

end

Illustrative Example



- Suffice to consider states s_1 and s_2 since s_3, s_4 and s_5 are terminal states.
- Denote the up ($\uparrow$) action by a_1 and the right ($\rightarrow$) action by a_2.
- Consider the softmax parameterization,

$$\pi_\theta(a|s) = \frac{\exp(\theta_{s,a})}{\sum_{a' \in \mathcal{A}} \exp(\theta_{s,a'})},$$

with parameters $\theta = (\theta_{s_1,a_1}, \theta_{s_1,a_2}, \theta_{s_2,a_1}, \theta_{s_2,a_2})^T$.

Illustrative Example (Cont'd)

Assume $\gamma = 1$. Let $\theta_0 = (0, 0, 0, 0)^\mathsf{T}$. Sample episode $\tau = (s_1, a_2, 0, s_2, a_1, r, s_4)$.
First recall that

$$\frac{\partial \log \pi_\theta(a'|s')}{\partial \theta_{s,a}} = 1_{[s'=s]}(1_{[a'=a]} - \pi_\theta(a|s)).$$

At timestep $t = 0$:

- Calculate the total rewards: $G_0 = 0 + r = r$;
- Calculate $\nabla_\theta \log \pi_\theta(a_2|s_1) = (-\frac{1}{2}, \frac{1}{2}, 0, 0)^\mathsf{T}$.

At timestep $t = 1$:

- Calculate the total rewards: $G_1 = r$;
- Calculate $\nabla_\theta \log \pi_\theta(a_1|s_2) = (0, 0, \frac{1}{2}, -\frac{1}{2})^\mathsf{T}$.

Parameter update:

$$\theta \leftarrow \theta + \nabla_\theta \log \pi_\theta(a_2|s_1)G_0 + \nabla_\theta \log \pi_\theta(a_1|s_2)G_1 = (-\frac{r}{2}, \frac{r}{2}, \frac{r}{2}, -\frac{r}{2})^T.$$

Variance Reduction with Baseline

Recall the action value expression

$$\nabla V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_\mu^{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\left[Q^{\pi_\theta}(s,a)\nabla_\theta \log \pi_\theta(a|s)\right].$$

Conditioned on s, we would like to find a baseline $b(s)$ such that variance of $(Q^{\pi_\theta}(s,a) - b(s))\nabla_\theta \log \pi_\theta(a|s)$ is reduced. Assume $\sum_a \pi_\theta(a|s) = 1$ for any θ.

- Expectation is not changed by adding $b(s)$ since

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

- The optimal $b(s)$ (with respect to a) is given by

$$b(s) = \frac{\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\left[Q^{\pi_\theta}(s,a)\|\nabla_\theta \log \pi_\theta(a|s)\|_2^2\right]}{\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\left[\|\nabla_\theta \log \pi_\theta(a|s)\|_2^2\right]}.$$

Variance Reduction with Baseline (Cont'd)

- With the baseline, the action value expression for policy gradient becomes

$$\nabla V^{\pi_\theta}(\mu) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_\mu^{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}\left[(Q^{\pi_\theta}(s,a) - b(s))\nabla_\theta \log \pi_\theta(a|s)\right]$$
$$= \mathbb{E}_{\tau\sim P_\mu^{\pi_\theta}}\left[\sum_{t=0}^{\infty}\gamma^t\big(Q^{\pi_\theta}(s_t,a_t) - b(s_t)\big)\nabla_\theta \log \pi_\theta(a_t|s_t)\right].$$

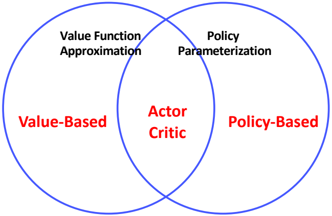
- Note that the optimal $b(s)$ can be viewed as the expected value of Q-values, but weighted by gradient magnitudes. Thus, it is reasonable to take

$$b(s) = V^{\pi_\theta}(s),$$

which leads to the advantage function expression of policy gradient.

Table of Contents

Overall Idea



- Value-based: Learn value function
- Policy-based: Learn policy function
- Actor-critic: Learn value and policy functions

Actor-Critic Methods

Motivation. MC policy gradient evaluation is sample inefficient and has high variance. Similar to VFA in value-based RL, we can approximate values that appears in policy gradient and update VFA parameters in learning process.

- Actor: Learn parameterized policy π_θ via policy gradient;
- Critic: Learn value function $V(:;\omega)$ or $Q(:;\omega)$ in $\nabla V^{\pi_\theta}(\mu)$ via policy evaluation.

Recall TD evaluation for state value and action value parameter as follows:

$$
\begin{aligned}
\text{(State value)} \quad & \delta_t = r_t + \gamma \cdot V(s_{t+1};\omega) - V(s_t;\omega) \\
& \omega \leftarrow \omega + \alpha_t \, \delta_t \, \nabla_\omega V(s_t;\omega) \\
\text{(Action value)} \quad & \delta_t = r_t + \gamma \cdot Q(s_{t+1}, a_{t+1};\omega) - Q(s_t, a_t;\omega) \\
& \omega \leftarrow \omega + \alpha_t \, \delta_t \, \nabla_\omega Q(s_t, a_t;\omega)
\end{aligned}
$$

Action-Value Actor-Critic

Algorithm 2: Action-Value Actor-Critic

Initialization: policy parameters θ_0, action value function parameter ω_0.

for $t = 0, 1, \cdots$ **do**

  Sample a tuple $(s_t, a_t, r_t, s_{t+1}, a_{t+1}) \sim \pi_\theta$

  Calculate $\delta_t \leftarrow r_t + \gamma \cdot Q(s_{t+1}, a_{t+1}; \omega) - Q(s_t, a_t; \omega)$

  Critic update: $\omega \leftarrow \omega + \alpha_t \, \delta_t \, \nabla_\omega Q(s_t, a_t; \omega)$

  Actor update: $\theta \leftarrow \theta + \beta_t \, Q(s_t, a_t; \omega) \nabla_\theta \log \pi_\theta \, (a_t | s_t)$

end

There are other versions of actor-critic, for example, the parameters are only updated at the end of an episode by using all the episode data simultaneously.

Advantage Actor-Critic Method (A2C)

In A2C, advantage function expression for policy gradient is used and value function approximation is applied to state values:

$$Q(s_t, a_t) \approx r_t + \gamma V(s_{t+1}; \omega), \quad A(s_t, a_t) \approx \underbrace{r_t + \gamma V(s_{t+1}; \omega) - V(s_t; \omega)}_{\delta_t}$$

Algorithm 3: Advantage Actor-Critic (A2C)

Initialization: policy parameters θ_0, state value function parameter ω_0.

for $t = 0, 1, \cdots$ **do**

  Sample a tuple $(s_t, a_t, r_t, s_{t+1}) \sim \pi_\theta$

  Calculate $\delta_t \leftarrow r_t + \gamma V(s_{t+1}; \omega) - V(s_t; \omega)$

  Critic update: $\omega \leftarrow \omega + \alpha_t \, \delta_t \, \nabla_\omega V(s_t; \omega)$

  Actor update: $\theta \leftarrow \theta + \beta_t \, \delta_t \, \nabla_\theta \log \pi_\theta \, (a_t | s_t)$

end

Questions?