Algorithmic and Theoretical Foundations of RL

Policy Optimization

Table of Contents

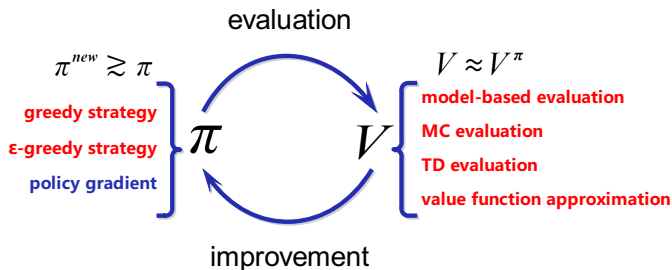# Value-Based vs Policy-Based RL

- Value-based RL: Learn (optimal) state/action values and policy is implicitly inferred (e.g., greedy or ϵ-greedy);
- Policy-based RL: Directly parametrize the policy and search in the space of policies via optimization.

Consider a policy parameterization 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.

However, since $J(\theta)$ can be highly non-concave, optimization methods tend to converge to a local optimum rather than a global optimum.

For discrete action space, a common choice is softmax policy (Boltzmann distribution):

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

A special example is when

$$f_\theta(s, a) = \phi(s, a)^\top \theta,$$

where $\phi(s, a)$ is a feature vector.

For continuous action space, Gaussian distribution is natural:

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

A special example is

$$\pi_\theta(\cdot|s) \text{ is the pdf of } \mathcal{N}(\phi(s)^T\theta, \sigma^2),$$

where $\phi(s)$ is a feature vector.

Policy Optimization

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

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

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 the RL problem as

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \, J(\theta).$$

▶ The initial state distribution can be for example Dirac delta distribution, uniform distribution, or stationary distribution under policy π_θ. Here we assume μ is a fixed distribution .

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

Theorem 1. For any policy π, there holds

$$\mathbb{E}_{s_0 \sim \mu} \left[v_\pi(s_0) \right] = \frac{1}{1-\gamma} \mathbb{E}_{s \sim d_\mu^\pi} \mathbb{E}_{a \sim \pi(\cdot|s)} \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[r(s, a, s') \right],$$

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.

If $\pi = \pi_\theta$, then $\mathbb{E}_{s_0 \sim \mu} \left[v_\pi(s_0) \right] = J(\theta)$.

$$v_\pi(s_0) = \mathbb{E}_{\tau \sim P_{s_0}^\pi} \left[\sum_{t=0}^\infty \gamma^t r(s_t, a_t, s_{t+1}) \right]$$

$$= \sum_{t=0}^\infty \mathbb{E}_{\tau \sim P_{s_0}^\pi} \left[\gamma^t r(s_t, a_t, s_{t+1}) \right]$$

$$= \sum_{t=0}^\infty \sum_s \sum_a \sum_{s'} \gamma^t \cdot P(s_t = s | s_0, \pi) \pi(a|s) P(s'|s, a) r(s, a, s')$$

$$= \sum_s \sum_a \sum_{s'} \left(\sum_{t=0}^\infty \gamma^t \cdot P(s_t = s | s_0, \pi) \right) \pi(a|s) P(s'|s, a) r(s, a, s')$$

$$= \frac{1}{1-\gamma} \sum_s \sum_a \sum_{s'} d_{s_0}^\pi(s) \pi(a|s) P(s'|s, a) r(s, a, s').$$

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

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 $(I - \gamma(P^\pi)^T)^{-1}$.

- ▶ 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.

Theorem 2. Suppose π_θ is differentiable for all states and actions. We have the following expressions for $\nabla J(\theta)$:

▶ Return expression:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}} \left[r(\tau) \sum_{t=0}^{\infty} \nabla_\theta \log \pi_\theta(a_t|s_t) \right];$$

▶ Action value expression:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}} \left[\sum_{t=0}^{\infty} \gamma^t q_{\pi_\theta}(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t|s_t) \right].$$

The following log derivative trick will be very useful in the proof:

$$\nabla f(x) = f(x) \nabla \log f(x).$$

By the definition of $J(\theta)$, we have

$$
\begin{aligned}
\nabla J(\theta) &= \sum_{\tau} r(\tau) \nabla_\theta P_\mu^{\pi_\theta} \\
&= \sum_{\tau} r(\tau) P_\mu^{\pi_\theta} \nabla_\theta \log P_\mu^{\pi_\theta} \\
&= \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}} \left[r(\tau) \log P_\mu^{\pi_\theta} \right] \\
&= \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}} \left[r(\tau) \sum_{t=0}^{\infty} \nabla_\theta \log \pi_\theta(a_t | s_t) \right].
\end{aligned}
$$

Proof of Theorem 2 (Cont'd)

For the action value expression, we have

$$\mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}} \left[r(\tau) \sum_{t=0}^\infty \nabla_\theta \log \pi_\theta(a_t|s_t) \right]$$

$$= \sum_{t=0}^\infty \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}} \left[r(\tau) \nabla_\theta \log \pi_\theta(a_t|s_t) \right]$$

$$= \sum_{t=0}^\infty \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}} \left[\sum_{k=0}^\infty \gamma^k r(s_k, a_k, s_{k+1}) \nabla_\theta \log \pi_\theta(a_t|s_t) \right]$$

$$= \sum_{t=0}^\infty \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}} \left[\sum_{k=0}^{t-1} \gamma^k r(s_k, a_k, s_{k+1}) \nabla_\theta \log \pi_\theta(a_t|s_t) \right]$$

$$+ \sum_{t=0}^\infty \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}} \left[\sum_{k=t}^\infty \gamma^k r(s_k, a_k, s_{k+1}) \nabla_\theta \log \pi_\theta(a_t|s_t) \right]$$

The part can also be derived via the recursive relation $v_{\pi_\theta}(s_0) = \sum_{a_0} \pi_\theta(a_0|s_0) q_{\pi_\theta}(s_0, a_0)$ and the chain rule.

$$= \sum_{t=0}^{\infty} \mathbb{E}_{\tau \sim P_{\mu}^{\pi_\theta}} \left[\sum_{k=t}^{\infty} \gamma^k r(s_k, a_k, s_{k+1}) \nabla_\theta \log \pi_\theta(a_t|s_t) \right]$$

$$= \sum_{t=0}^{\infty} \mathbb{E}_{\tau \sim P_{\mu}^{\pi_\theta}} \left[\gamma^t q_{\pi_\theta}(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t|s_t) \right],$$

where in the fourth equality we have used the fact

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

The policy gradient theorem allows us to decompose the policy gradient into the expression of an expectation over state-action pairs.

Theorem 3. Recalling the definition of visitation measure, we have

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

Proof. The proof is overall similar to that of Theorem 1.

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

▶ 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.

Table of Contents

From the proof of Theorem 2, we can see that the random variable in action value expression potentially has smaller variance than that in return version since some randomness is eliminated in the proof. Thus we consider MC evaluation of action value expression.

▶ 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 $q_{\pi_\theta}(s_t, a_t)$:

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

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:

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

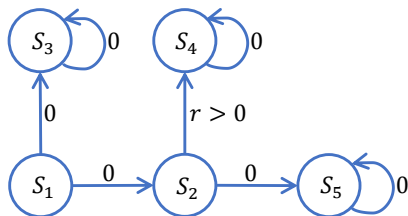Policy parameter update:

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

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 $\theta_0 = (0, 0, 0, 0)^\top$ and sample episode $\tau = (s_1, a_2, 0, s_2, a_1, r, s_4)$.
First note that

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

At timestep $t = 0$:

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

At timestep $t = 1$:

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

Parameter update:

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

Recall the action value expression

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

Conditioned on s, we would like to find a baseline (control variate) $b(s)$ such that the variance of $(q_{\pi_\theta}(s,a) - b(s)) \nabla_\theta \log \pi_\theta(a|s)$ can be reduced.

▶ The 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 $b(s)$ is given by

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

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

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

▶ 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 is a common baseline in practice. With this choice for $b(s)$, we have the advantage function expression for policy gradient

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim P_\mu^{\pi_\theta}} \left[\sum_{t=0}^\infty \gamma^t A_{\pi_\theta}(s_t, a_t) \nabla_\theta \log \pi_\theta(a_t|s_t) \right],$$

where $A_{\pi_\theta}(s_t, a_t) = q_{\pi_\theta(s_t, a_t)} - v_{\pi_\theta}(s_t)$ is the advantage function.
Since $v_{\pi_\theta}(s)$ is not directly accessible, we may estimate it via value function approximation $v_{\pi_\theta}(s) \approx v(s; \omega)$.

REINFORCE with Baseline

Algorithm 2: REINFORCE with Baseline

Initialization: initial policy parameters θ_0, initial value function parameters ω_0 .

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

    Sample a trajectories $\mathcal{D}_k = \{\tau_i\}$ following π_{θ_k}.

    for $t = 0, 1, 2 \ldots T - 1$ *of each trajectory* τ_i **do**

        Compute the return: $G_t^{(i)} = \sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'}^{(i)}$.

        Compute advantage estimates: $\widehat{A}_t^{(i)} = G_t^{(i)} - V(s_t; \omega_k)$.

    end

    Estimate policy gradient:

$$g_k = \frac{1}{|\mathcal{D}_k|} \sum_{i=1}^{|\mathcal{D}_k|} \sum_{t=0}^{T-1} \gamma^t \cdot \widehat{A}_t^{(i)} \nabla_\theta \log \pi_{\theta_k} \left(a_t^{(i)} | s_t^{(i)} \right).$$

    Policy parameter update:

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

    Fit value function by regression on mean-squared error (via e.g., GD/SGD):

$$\omega_{k+1} = \operatorname*{argmin}_\omega \frac{1}{|\mathcal{D}_k| T} \sum_{i=1}^{|\mathcal{D}_k|} \sum_{t=0}^{T-1} \left(V(s_t; \omega) - G_t^{(i)} \right)^2$$

end

Table of Contents

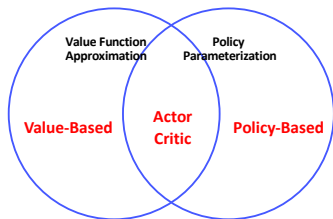▶ Value-based: Learn value function

▶ (Pure) Policy-based: Learn policy function

▶ Actor-critic: learn value and policy functions

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 the policy gradient and update VFA parameters in learning process.

▶ Actor: learn parameterized policy $\pi_\theta(\cdot|\cdot)$ via policy gradient;

▶ Critic: learn value function $v(:;\omega)$ or $q(:;\omega)$ in $\nabla J(\theta)$ via policy evaluation.

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

(State value) $\quad \delta_t = r_t + \gamma \cdot v(s_{t+1};\omega) - v(s_t;\omega), \ \omega \leftarrow \omega + \alpha_t \delta_t \nabla_\omega v(s_t;\omega)$

(Action value) $\quad \delta_t = r_t + \gamma \cdot q(s_{t+1}, a_{t+1};\omega) - q(s_t, a_t;\omega)$

$\qquad\qquad\quad \omega \leftarrow \omega + \alpha_t \delta_t \nabla_\omega q(s_t, a_t;\omega) - q(s_t, a_t;\omega)$

Actor-critic (or policy gradient) methods fit into the framework of policy evaluation and policy improvement. Policy improvement is achieved by gradient ascent, hence is actor. The evaluation of policy (state/action/advantage) instructs the direction to improve the policy, hence is critic.

Algorithm 3: 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: $\omega \leftarrow \omega + \beta_t \cdot \delta_t \nabla_\omega q(s_t, a_t; \omega)$

    Actor update: $\theta \leftarrow \theta + \alpha_t \cdot q(s_t, a_t; \omega) \nabla_\theta \log \pi_\theta (a_t | s_t)$

end

There are other versions of actor-critic, e.g., 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 used 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 4: 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: $\omega \leftarrow \omega + \beta_t \cdot \delta_t \nabla_\omega v(s_t; \omega)$

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

end

Questions?