Actor-Critic Methods

Chen-Yu Wei

Review: Full-Information Policy Learning in MDPs

$$\theta_{k+1} = \operatorname*{argmax}_{\theta} \left(\underbrace{V^{\pi_\theta}(\rho) - V^{\pi_{\theta_k}}(\rho)} - \frac{1}{\eta} D(\theta, \theta_k) \right)$$

$$\approx \sum_{s,a} d_\rho^{\pi_{\theta_k}}(s) \left(\pi_\theta(a|s) - \pi_{\theta_k}(a|s)\right) Q^{\pi_{\theta_k}}(s,a) = \mathbb{E}_{\underset{\sim \pi_{\theta_k}}{(s_i,a_i)}} \left[\frac{\pi_\theta(a_i|s_i) - \pi_{\theta_k}(a_i|s_i)}{\pi_{\theta_k}(a_i|s_i)} \boxed{Q^{\pi_{\theta_k}}(s_i,a_i)} \right]$$

$$\approx (\theta - \theta_k)^\top \underbrace{\sum_{s,a} d_\rho^{\pi_{\theta_k}}(s) \left(\nabla_\theta \pi_\theta(a|s) \Big|_{\theta=\theta_k} \right) Q^{\pi_{\theta_k}}(s,a)}$$

$$= \mathbb{E}_{(s_i,a_i)} \left[\frac{\nabla_\theta \pi_\theta(a_i|s_i)|_{\theta=\theta_k}}{\pi_{\theta_k}(a_i|s_i)} \boxed{Q^{\pi_{\theta_k}}(s_i,a_i)} \right]$$

PG/NPG: Estimate them using the empirical sum of reward in the trajectory (i.e., Monte Carlo estimator)

We can also use other estimators to balance bias and variance

Actor-Critic Methods

Use value function approximation to estimate $Q^{\pi_{\theta_k}}(s_i, a_i)$ or $A^{\pi_{\theta_k}}(s_i, a_i)$

Use $V_\phi(s)$ to approximate $V^{\pi_{\theta_k}}(s)$

Use $Q_\phi(s, a)$ to approximate $Q^{\pi_{\theta_k}}(s, a)$

Possible estimators for $A^{\pi_{\theta_k}}(s, a)$:

Let $(s_1, a_1, r_1, s_2, a_2, r_2 \ldots)$ be a trajectory starting from $s_1 = s, a_1 = a$

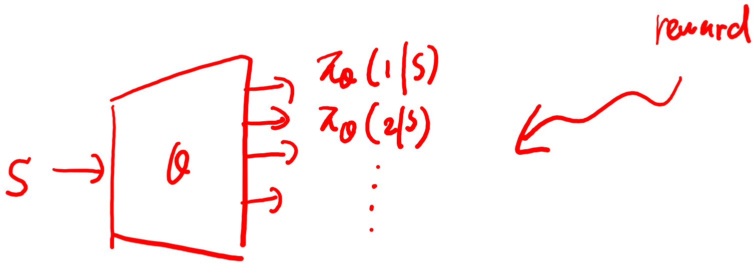
$$Q_\phi(s_1, a_1) - \mathbb{E}_{a' \sim \pi_{\theta_k}(\cdot|s)}[Q_\phi(s_1, a')]$$

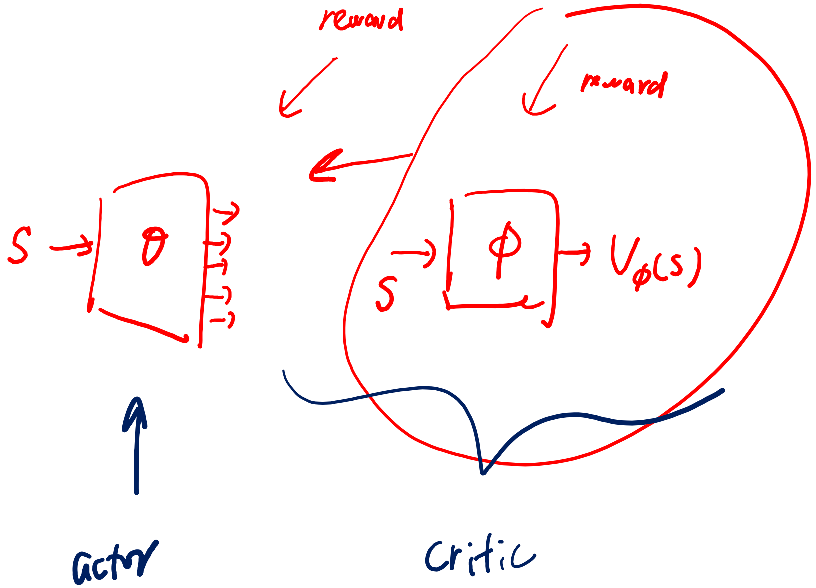
$$r_1 + \gamma V_\phi(s_2) - V_\phi(s_1) \qquad r_1 + \gamma Q_\phi(s_2, a_2) - \mathbb{E}_{a' \sim \pi_{\theta_k}(\cdot|s)}[Q_\phi(s_1, a')]$$

$$r_1 + \gamma r_2 + \gamma^2 V_\phi(s_3) - V_\phi(s_1) \qquad r_1 + \gamma r_2 + \gamma^2 Q_\phi(s_3, a_3) - \mathbb{E}_{a' \sim \pi_{\theta_k}(\cdot|s)}[Q_\phi(s_1, a')]$$

$$\vdots \qquad \vdots$$

Pure Policy-Based Methods vs. Actor-Critic Methods





Actor-Critic with Q_ϕ

(find π^*) (off-policy) Q-learning: $Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha\left[r + \max_{a'} Q(s',a')\right] \Rightarrow Q^*$

(given π) TD-learning: $Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha\left[r + \sum_{a'} \pi(a'|s')Q(s',a')\right] \Rightarrow Q^\pi$ (on-policy)

For $k = 1, 2, \ldots$

Use π_{θ_k} to collect n trajectories

$$\left(s_1^{(1)}, a_1^{(1)}, r_1^{(1)}, \cdots, s_{\tau_1}^{(1)}, a_{\tau_1}^{(1)}, r_{\tau_1}^{(1)}\right), \ldots\ldots, \left(s_1^{(n)}, a_1^{(n)}, r_1^{(n)}, \cdots, s_{\tau_n}^{(n)}, a_{\tau_n}^{(n)}, r_{\tau_n}^{(n)}\right)$$

Define

$$g = \frac{1}{n}\sum_{i=1}^{n}\sum_{h=1}^{\tau_n} \frac{\nabla_\theta\, \pi_\theta\left(a_h^{(i)} \middle| s_h^{(i)}\right)\Big|_{\theta=\theta_k}}{\pi_{\theta_k}\left(a_h^{(i)} \middle| s_h^{(i)}\right)} \underbrace{Q_{\phi_k}\left(s_h^{(i)}, a_h^{(i)}\right)}_{\approx Q^{\pi_{\theta_k}}} \quad \text{or} \quad \frac{1}{n}\sum_{i=1}^{n}\sum_{h=1}^{\tau_n}\sum_{a} \nabla_\theta\, \pi_\theta\left(a \middle| s_h^{(i)}\right)\Big|_{\theta=\theta_k} Q_{\phi_k}\left(s_h^{(i)}, a\right)$$

Perform updates

$$\theta_{k+1} \leftarrow \theta_k + \eta g \qquad \phi_{k+1} \leftarrow \phi_k - \lambda \nabla_\phi \sum_{i=1}^{n}\sum_{h=1}^{\tau_n}\left(Q_\phi\left(s_h^{(i)}, a_h^{(i)}\right) - r_h^{(i)} - \gamma Q_{\phi_k}\left(s_{h+1}^{(i)}, a_{h+1}^{(i)}\right)\right)^2\Bigg|_{\phi=\phi_k}$$

Advantage Actor-Critic (A2C) = PG + V_ϕ

For $k = 1,\ 2, \ldots$

Use π_{θ_k} to collect n trajectories

$$\left(s_1^{(1)}, a_1^{(1)}, r_1^{(1)}, \cdots, s_{\tau_1}^{(1)}, a_{\tau_1}^{(1)}, r_{\tau_1}^{(1)}\right), \ldots \ldots, \left(s_1^{(n)}, a_1^{(n)}, r_1^{(n)}, \cdots, s_{\tau_n}^{(n)}, a_{\tau_n}^{(n)}, r_{\tau_n}^{(n)}\right)$$

$$= \frac{\nabla_\theta \pi_\theta(a_h^{(i)}|s_h^{(i)})\big|_{\theta=\theta_k}}{\pi_{\theta_k}(a_h^{(i)}|s_h^{(i)})} \qquad \approx A^{\pi_k}(s_h^{(i)}, a_h^{(i)})$$

$$\mathbb{E}(\cdot) = \sum_{s,a} d_\rho^{\pi_{\theta_k}}(s)\, \nabla_\theta \pi_\theta(a|s)\, A^{\pi_k}(s,a)$$

Define

$$g = \frac{1}{n}\sum_{i=1}^{n}\sum_{h=1}^{\tau_n} \nabla_\theta \log \pi_\theta\left(a_h^{(i)} \middle| s_h^{(i)}\right)\Big|_{\theta=\theta_k} \underbrace{\left(r_h^{(i)} + \gamma V_{\phi_k}\left(s_{h+1}^{(i)}\right) - V_{\phi_k}\left(s_h^{(i)}\right)\right)}$$

or any other advantage estimator in the previous slide

Perform updates

$V_\phi \approx V^{\pi_{\theta_k}}$.

$$\theta_{k+1} \leftarrow \theta_k + \eta g \qquad \phi_{k+1} \leftarrow \phi_k - \lambda \nabla_\phi \frac{1}{n}\sum_{i=1}^{n}\sum_{h=1}^{\tau_n}\left(V_\phi\left(s_h^{(i)}\right) - r_h^{(i)} - \gamma V_{\phi_k}\left(s_{h+1}^{(i)}\right)\right)^2\Bigg|_{\phi=\phi_k}$$

Mnih et al., Asynchronous Methods for Deep Reinforcement Learning. 2016.

Proximal Policy Optimization (PPO) = NPG + V_ϕ

For $k = 1,\ 2, \ldots$

Use π_{θ_k} to collect n trajectories

$$\left(s_1^{(1)}, a_1^{(1)}, r_1^{(1)}, \cdots, s_{\tau_1}^{(1)}, a_{\tau_1}^{(1)}, r_{\tau_1}^{(1)}\right), \ldots\ldots, \left(s_1^{(n)}, a_1^{(n)}, r_1^{(n)}, \cdots, s_{\tau_n}^{(n)}, a_{\tau_n}^{(n)}, r_{\tau_n}^{(n)}\right)$$

Perform updates

or any other advantage estimator in the previous slide

$$\theta_{k+1} \leftarrow \underset{\theta}{\operatorname{argmax}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \sum_{h=1}^{\tau_n} \frac{\pi_\theta\left(a_h^{(i)} \middle| s_h^{(i)}\right)}{\pi_{\theta_k}\left(a_h^{(i)} \middle| s_h^{(i)}\right)} \left(r_h^{(i)} + \gamma V_{\phi_k}\left(s_{h+1}^{(i)}\right) - V_{\phi_k}\left(s_h^{(i)}\right)\right) - \frac{1}{\eta} \frac{1}{n} \sum_{i=1}^{n} \sum_{h=1}^{\tau_n} \mathrm{KL}\left(\pi_\theta\left(\cdot \middle| s_h^{(i)}\right), \pi_{\theta_k}\left(\cdot \middle| s_h^{(i)}\right)\right) \right\}$$

$$\phi_{k+1} \leftarrow \phi_k - \lambda \nabla_\phi \frac{1}{n} \sum_{i=1}^{n} \sum_{h=1}^{\tau_n} \left(V_\phi\left(s_h^{(i)}\right) - r_h^{(i)} - \gamma V_{\phi_k}\left(s_{h+1}^{(i)}\right)\right)^2 \Bigg|_{\phi=\phi_k}$$

Schulman et al., Proximal Policy Optimization Algorithms. 2017.

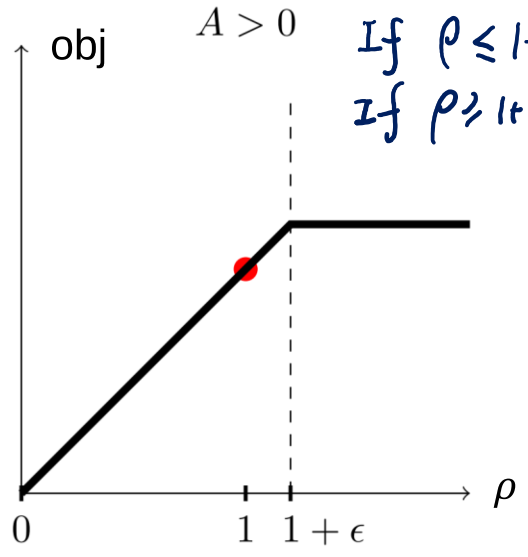
$\frac{r(a)\mathbb{1}(a_t=a)}{p_t(a)}$ $\frac{(r(a)-1)}{p_t(a)}\mathbb{1}(a_t=a)$

Additional Technique 1: Clipped Objective (for PPO)

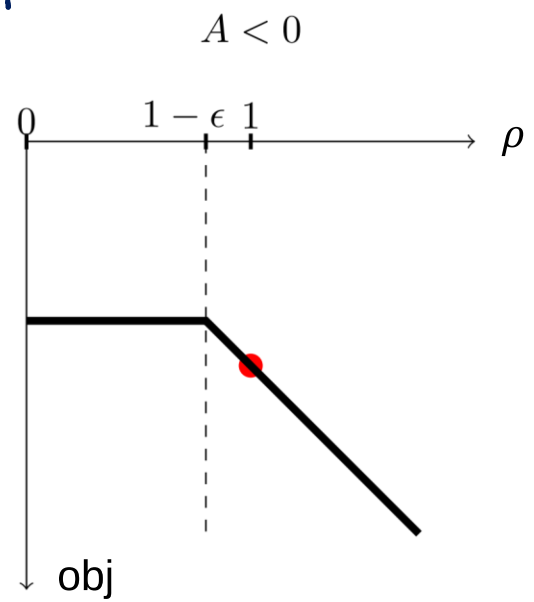
$$\rho := \frac{\pi_\theta\left(a_h^{(i)} \middle| s_h^{(i)}\right)}{\pi_{\theta_k}\left(a_h^{(i)} \middle| s_h^{(i)}\right)} \qquad A := \left(r_h^{(i)} + \gamma V_{\phi_k}\left(s_{h+1}^{(i)}\right) - V_{\phi_k}\left(s_h^{(i)}\right)\right)$$

$\mathrm{clip}_{[1-\varepsilon,1+\varepsilon]}(\rho) = \min\left(\max(\rho, 1-\varepsilon), 1+\varepsilon\right)$

Instead of using ρA as the objective, use $\min\{\rho A,\ \mathrm{clip}_{[1-\epsilon,1+\epsilon]}(\rho)A\}$



If $\rho \in [1-\varepsilon, 1+\varepsilon] \Rightarrow \rho A$
If $\rho \leq 1-\varepsilon \Rightarrow \rho A$
If $\rho \geq 1+\varepsilon \Rightarrow (1+\varepsilon)A$



If $\rho \in (1-\varepsilon, 1+\varepsilon] \Rightarrow \rho A$
If $\rho \leq 1-\varepsilon \Rightarrow (1-\varepsilon)A$ (strange case)
If $\rho \geq 1+\varepsilon \Rightarrow \rho A$

algorithm	avg. normalized score
No clipping or penalty	-0.39
Clipping, $\epsilon = 0.1$	0.76
Clipping, $\epsilon = 0.2$	**0.82**
Clipping, $\epsilon = 0.3$	0.70
Adaptive KL $d_{\text{targ}} = 0.003$	0.68
Adaptive KL $d_{\text{targ}} = 0.01$	0.74
Adaptive KL $d_{\text{targ}} = 0.03$	0.71
Fixed KL, $\beta = 0.3$	0.62
Fixed KL, $\beta = 1.$	0.71
Fixed KL, $\beta = 3.$	0.72
Fixed KL, $\beta = 10.$	0.69

Schulman et al., Proximal Policy Optimization Algorithms. 2017.

Additional Technique 2: Entropy Bonus

In the objective of policy update, add a bonus term

$$H(\pi_\theta(\cdot\,|s)) = \sum_a \pi_\theta(a|s)\ln\frac{1}{\pi_\theta(a|s)}$$

For PPO:

$$\underset{\theta}{\operatorname{argmax}} \left\{ \frac{1}{n}\sum_{i=1}^{n}\sum_{h=1}^{\tau_n} \frac{\pi_\theta\left(a_h^{(i)}\middle|s_h^{(i)}\right)}{\pi_{\theta_k}\left(a_h^{(i)}\middle|s_h^{(i)}\right)} A_h^{(i)} - \frac{1}{\eta}\frac{1}{n}\sum_{i=1}^{n}\sum_{h=1}^{\tau_n} \mathrm{KL}\left(\pi_\theta\left(\cdot\middle|s_h^{(i)}\right), \pi_{\theta_k}\left(\cdot\middle|s_h^{(i)}\right)\right) + c\,\frac{1}{n}\sum_{i=1}^{n}\sum_{h=1}^{\tau_n} \underbrace{H\left(\pi_\theta\left(\cdot\middle|s_h^{(i)}\right)\right)}_{-\mathrm{KL}\left(\pi_\theta\left(\cdot\middle|s_h^{(i)}\right), \pi_{\mathrm{unif}}\left(\cdot\middle|s_h^{(i)}\right)\right)} \right\}$$

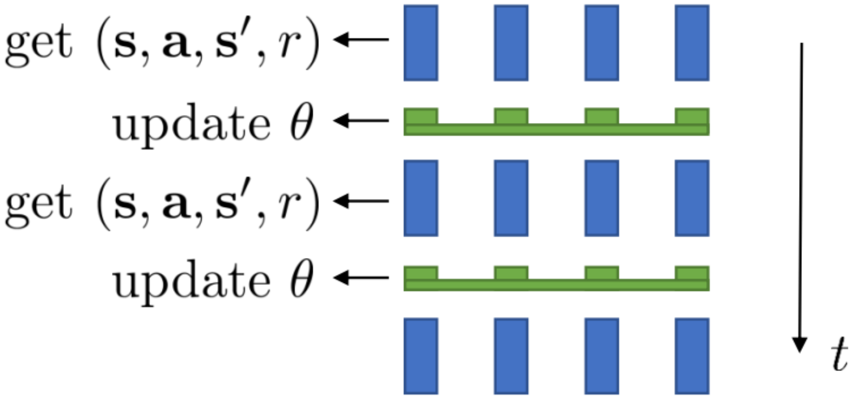
For A2C:

$$g = \frac{1}{n}\sum_{i=1}^{n}\sum_{h=1}^{\tau_n} \nabla_\theta \log \pi_\theta\left(a_h^{(i)}\middle|s_h^{(i)}\right)\Big|_{\theta=\theta_k} A_h^{(i)} + c\nabla_\theta \frac{1}{n}\sum_{i=1}^{n}\sum_{h=1}^{\tau_n} H\left(\pi_\theta\left(\cdot\middle|s_h^{(i)}\right)\right)$$

Additional Technique 3: Parallel Sample Collection

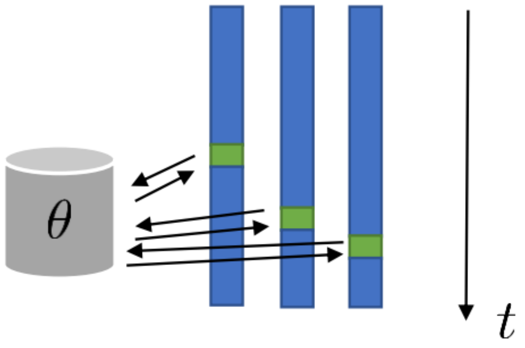
A2C

synchronized parallel actor-critic



A3C .

asynchronous parallel actor-critic

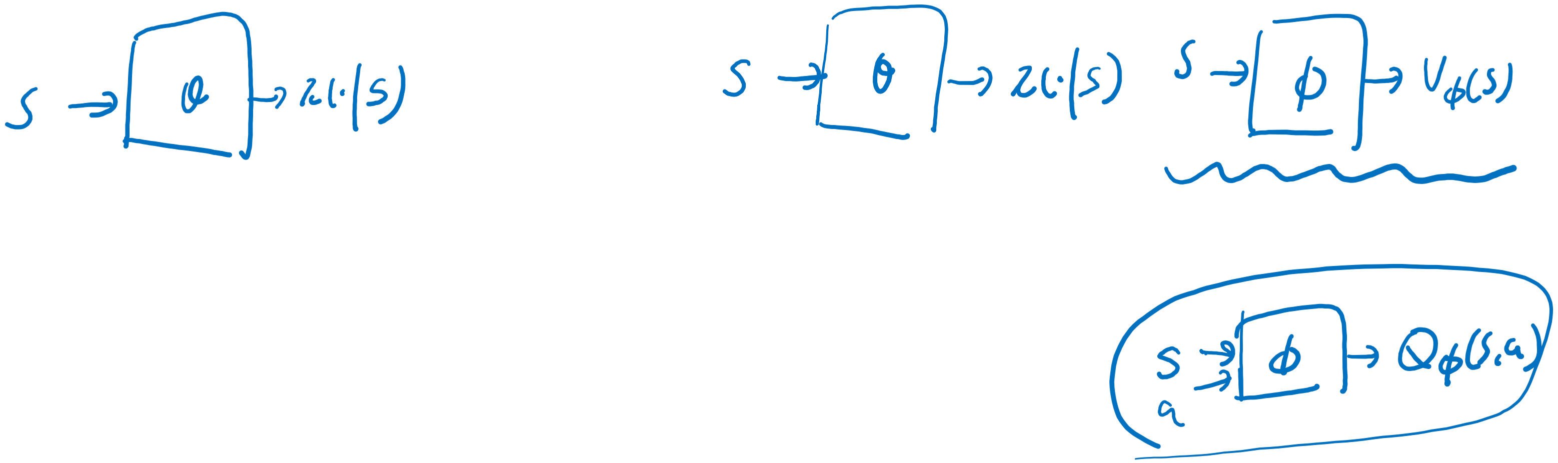


[Levine CS285 Lecture 6]

Actor-Critic Summary

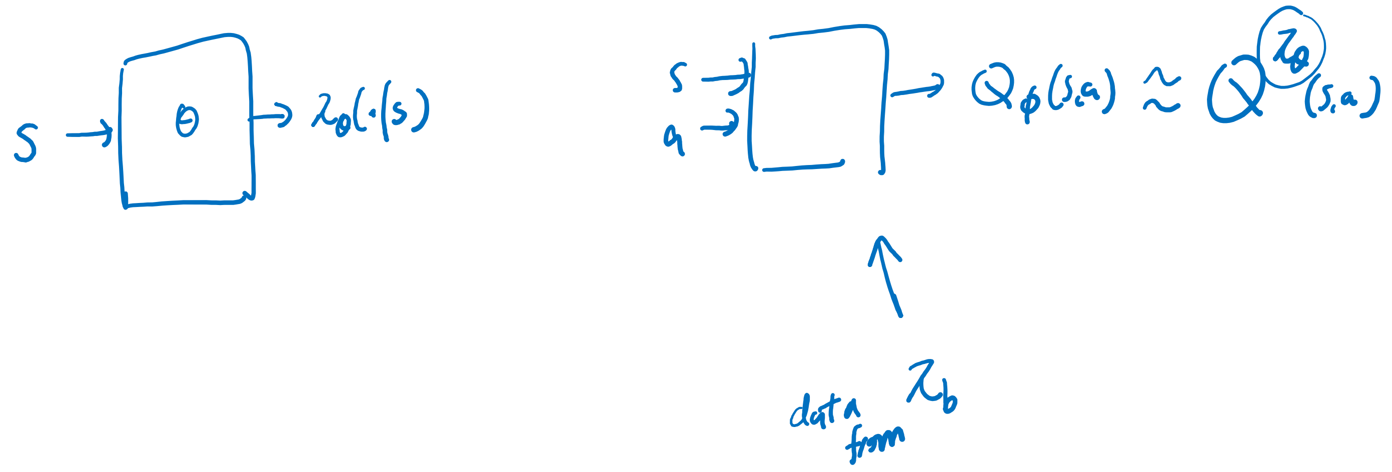
PG $\longrightarrow$ A2C

NPG $\longrightarrow$ PPO



Off-Policy Actor-Critic

- Leveraging **off-policy evaluation** → allow reusing data



Review: Full-Information Policy Learning in MDPs

$$\theta_{k+1} = \underset{\theta}{\operatorname{argmax}} \left(\underbrace{V^{\pi_\theta}(\rho) - V^{\pi_{\theta_k}}(\rho)} - \frac{1}{\eta} D(\theta, \theta_k) \right)$$

$$\approx \sum_{s,a} d_\rho^{\pi_{\theta_k}}(s) \left(\pi_\theta(a|s) - \pi_{\theta_k}(a|s) \right) Q^{\pi_{\theta_k}}(s,a)$$

$$\approx (\theta - \theta_k)^\top \sum_{s,a} d_\rho^{\pi_{\theta_k}}(s) \left(\nabla_\theta \pi_\theta(a|s) \Big|_{\theta=\theta_k} \right) Q^{\pi_{\theta_k}}(s,a)$$

Use any off-policy policy evaluation methods to find ϕ_k such that $Q_{\phi_k}(s,a) \approx Q^{\pi_{\theta_k}}(s,a)$

Suppose that our (s_i, a_i) samples are obtained from $\hat{\pi}$

Off-Policy Actor-Critic

$$\theta_{k+1} = \underset{\theta}{\operatorname{argmax}} \left(\underbrace{V^{\pi_\theta}(\rho) - V^{\pi_{\theta_k}}(\rho)} - \frac{1}{\eta} D(\theta, \theta_k) \right)$$

$$\approx \sum_{s,a} d_\rho^{\pi_{\theta_k}}(s) \left(\pi_\theta(a|s) - \pi_{\theta_k}(a|s) \right) Q_{\phi_k}(s,a) \quad = \mathbb{E}_{s \sim d_\rho^{\hat{\pi}}} \left[\cancel{\frac{d_\rho^{\pi_{\theta_k}}(s)}{d_\rho^{\hat{\pi}}(s)}} \sum_a \left(\pi_\theta(a|s) - \pi_{\theta_k}(a|s) \right) Q_{\phi_k}(s,a) \right]$$

$$\approx (\theta - \theta_k)^\top \sum_{s,a} d_\rho^{\pi_{\theta_k}}(s) \left(\nabla_\theta \pi_\theta(a|s) \Big|_{\theta=\theta_k} \right) Q_{\phi_k}(s,a) \quad = (\theta - \theta_k)^\top \mathbb{E}_{s \sim d_\rho^{\hat{\pi}}} \left[\cancel{\frac{d_\rho^{\pi_{\theta_k}}(s)}{d_\rho^{\hat{\pi}}(s)}} \sum_a \nabla_\theta \pi_\theta(a|s) \Big|_{\theta=\theta_k} Q_{\phi_k}(s,a) \right]$$

Use any off-policy policy evaluation methods to find ϕ_k such that $Q_{\phi_k}(s,a) \approx Q^{\pi_{\theta_k}}(s,a)$

Suppose that our (s_i, a_i) samples are obtained from $\hat{\pi}$

Actor-Critic + Replay Buffer

For $k = 1,\ 2, \ldots$

Collect samples using π_{θ_k}, and place them in the replay buffer

Sample a batch $\{(s_i, a_i, r_i, s_i')\}_{i=1}^{n}$ from replay buffer

Define

$$g = \frac{1}{n}\sum_{i=1}^{n}\sum_{a}\nabla_\theta \pi_\theta(a|s_i)\Big|_{\theta=\theta_k} Q_{\phi_k}(s_i, a)$$

Note: not using a_i here

Perform updates

Off-policy TD → unstable (more on this later)

$$\theta_{k+1} \leftarrow \theta_k + \eta g$$

$$\phi_{k+1} \leftarrow \phi_k - \lambda \nabla_\phi \frac{1}{n}\sum_{i=1}^{n}\left(Q_\phi(s_i, a_i) - r_i - \gamma\, \mathbb{E}_{a' \sim \pi_{\theta_k}(\cdot|s_i')}\left[Q_{\phi_k}(s_i', a')\right]\right)^2\Bigg|_{\phi=\phi_k}$$

Dealing with Continuous Action Sets

Review: Linear Bandits and One-Point Gradient Estimator

Feasible set $\mathcal{A} \subseteq \mathbb{R}^d$

For $t=1,\cdots,T$:

Learner choose $a_t \in \mathcal{A}$

Environment reveals $f_t(a_t)$, where $f_t : \mathcal{A} \to \mathbb{R}$

Ideal update

$$a_{t+1} \leftarrow a_t + \eta \nabla f_t(a_t)$$

$a_t - v$ a_t $a_t + v$

(1d)

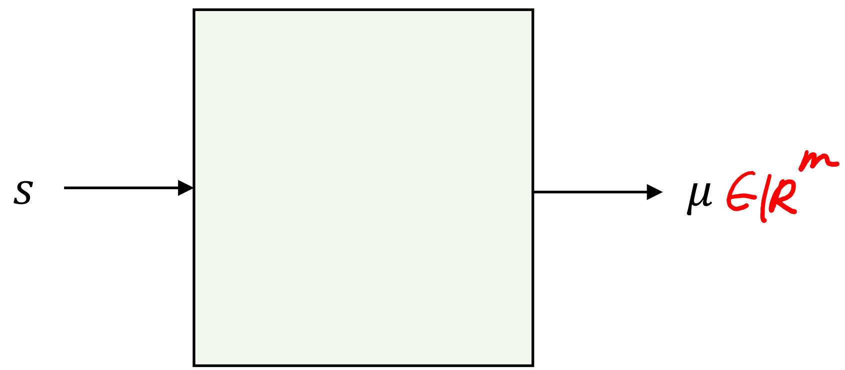
$$\nabla f_t(a_t) \approx \frac{f_t(a_t+v) - f_t(a_t-v)}{2v}$$

$$= \mathbb{E} \frac{f_t(\tilde{a}_t)\, s}{v} = \frac{f_t(\tilde{a}_t)(\tilde{a}_t - a_t)}{v^2}$$

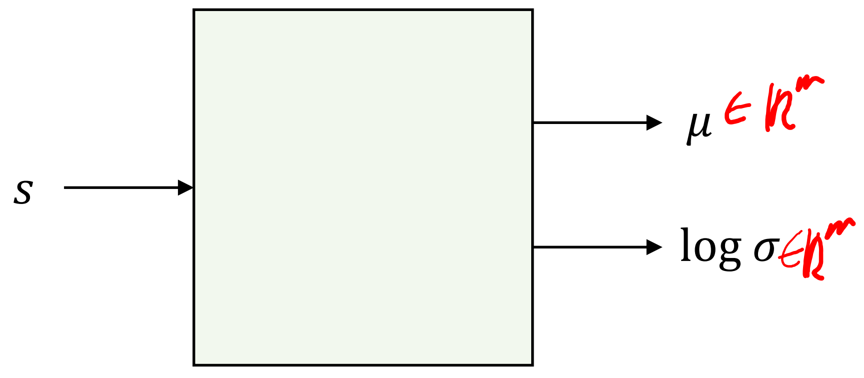
where $\tilde{a}_t = \begin{cases} a_t+v, & \text{w.p. } \frac{1}{2} \\ a_t-v, & \text{w.p. } \frac{1}{2} \end{cases}$

$$s = \begin{cases} 1, & \tilde{a}_t = a_t+v \\ -1, & \tilde{a}_t = a_t-v \end{cases}$$

Policy Network for Continuous Action Sets



Policy Network for Continuous Action Sets



$a_i = \mu_i + \varepsilon \sigma_i$, where $\varepsilon \sim \mathcal{N}(0,1)$

A2C / PPO with Continuous Action Sets

$\int_a \pi_\theta(a|s)\, da = 1$

μ_θ

$\pi_\theta(a) = \text{const} \cdot \exp\left(\frac{-(a-\mu_\theta)^2}{2\sigma^2}\right)$

$\mu_\theta \in \mathbb{R}^m$

$\theta \in \mathbb{R}^n$

$$g = \frac{1}{n}\sum_{i=1}^{n} \nabla_\theta \log \pi_\theta(a_i|s_i)\Big|_{\theta=\theta_k} A_i$$

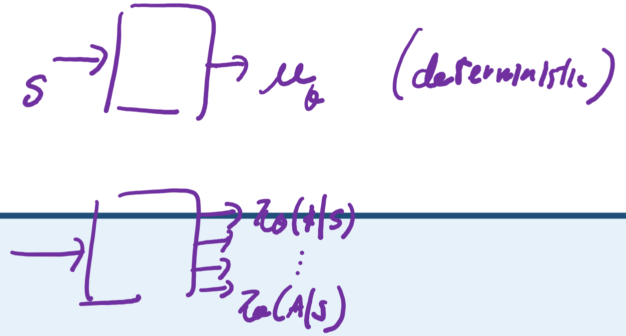
$\log \pi_\theta(a) = \text{const} - \frac{(a-\mu_\theta)^2}{2\sigma^2}$

$\nabla_\theta \log \pi_\theta(a) = \underbrace{\nabla_\theta \mu_\theta}_{\mathbb{R}^{n\times m}} \underbrace{\frac{(a-\mu_\theta)}{\sigma^2}}_{m} \in \mathbb{R}^n$

$$\theta_{k+1} \leftarrow \underset{\theta}{\operatorname{argmax}} \left\{ \frac{1}{n}\sum_{i=1}^{n} \frac{\pi_\theta(a_i|s_i)}{\pi_{\theta_k}(a_i|s_i)} A_i - \frac{1}{\eta}\frac{1}{n}\sum_{i=1}^{n} \mathrm{KL}\left(\pi_\theta(\cdot\,|s_i), \pi_{\theta_k}(\cdot\,|s_i)\right) \right\}$$

a

Recall: Actor-Critic with Q_ϕ Critic



For $k = 1,\ 2, \ldots$

Use π_{θ_k} to collect samples $\{(s_i, a_i, r_i, s_i')\}_{i=1}^{n}$

Define $\quad g = \frac{1}{n}\sum_{i=1}^{n}\sum_{a} \nabla_\theta\, \pi_\theta(a|s_i)\Big|_{\theta=\theta_k} Q_{\phi_k}(s_i, a)$

Perform updates

$$\theta_{k+1} \leftarrow \theta_k + \eta g \qquad \phi_{k+1} \leftarrow \phi_k - \lambda \nabla_\phi \sum_{i=1}^{n}\Big(Q_\phi(s_i, a_i) - r_i - \gamma\,\mathbb{E}_{a' \sim \pi_{\theta_k}(\cdot|s_i')}\big[Q_{\phi_k}(s_i', a')\big]\Big)^2\Bigg|_{\phi=\phi_k}$$

Deterministic Policy Gradient Theorem

Policy: $\mu_\theta(s) \in \mathbb{R}^m$

$$V^{\pi_{\theta+\Delta\theta}}(\rho) - V^{\pi_\theta}(\rho) = \sum_s d_\rho^{\pi_{\theta+\Delta\theta}}(s) \sum_a \left(\underline{\underline{\pi_{\theta+\Delta\theta}(a|s)}} - \underline{\underline{\pi_\theta(a|s)}} \right) Q^{\pi_\theta}(s, \underset{m}{a})$$

$$= \sum_s d_\rho^{\pi_{\theta+\Delta\theta}}(s) \left(Q^{\pi_\theta}(s, \mu_{\theta+\Delta\theta}(s)) - Q^{\pi_\theta}(s, \mu_\theta(s)) \right)$$

$$\Rightarrow \nabla_\theta V^{\pi_\theta}(\rho) \approx \sum_s d_\rho^{\pi_\theta}(s) \, \nabla_\theta \left[Q^{\pi_\theta}(s, \mu_\theta(s)) \right]$$

$$\left\{ \sum_s d_\rho^{\pi_\theta}(s) \left[\sum_a \nabla_\theta \pi_\theta(a|s) \, Q^{\pi_\theta}(s,a) \right] \right\}$$

A.C.

$Q_{\phi_k} \approx Q^{\mu_{\theta_k}}$

Deterministic Policy Gradient

Q-learning

$Q \leftarrow (1-\alpha) Q + \alpha \left(r + \gamma \max_{a'} Q(s', a') \right)$

For $k = 1,\ 2, \ldots$

Use μ_{θ_k} to collect samples $\{(s_i, a_i, r_i, s_i')\}_{i=1}^n$

Define $\quad g = \frac{1}{n} \sum_{i=1}^{n} \nabla_\theta Q_{\phi_k}\left(s_i, \mu_{\theta_k}(s_i)\right)\Big|_{\theta=\theta_k}$

Perform updates

$$\theta_{k+1} \leftarrow \theta_k + \eta g \qquad \phi_{k+1} \leftarrow \phi_k - \lambda \nabla_\phi \sum_{i=1}^{n} \left(Q_\phi(s_i, a_i) - r_i - \gamma Q_{\phi_k}(s_i', \mu_{\theta_k}(s_i')) \right)^2 \Big|_{\phi=\phi_k}$$

$\mu_{\theta_k}(s) \approx \operatorname{argmax}_a Q_{\phi_k}(s, a)$

Two Viewpoints for the Deterministic PG Algorithm

Deep Deterministic Policy Gradient (DDPG)

For $k = 1,\ 2, \ldots$

Use $\mu_{\theta_k}(s) + \mathcal{N}(0, \sigma^2)$ to collect samples and place them in replay buffer

Sample a batch $\{(s_i, a_i, r_i, s_i')\}_{i=1}^{n}$ from the replay buffer

$$\theta \leftarrow \theta + \eta \sum_{i=1}^{n} \nabla_\theta Q_\phi\big(s_i, \mu_\theta(s_i)\big)$$

$$\phi \leftarrow \phi - \lambda \nabla_\phi \sum_{i=1}^{n} \Big(Q_\phi(s_i, a_i) - r_i - \gamma Q_{\phi_{\text{tar}}}\big(s_i', \mu_{\theta_{\text{tar}}}(s_i{}')\big)\Big)^2$$

$$\theta_{\text{tar}} \leftarrow \overset{0.001}{\tau}\theta + (1-\tau)\theta_{\text{tar}}$$

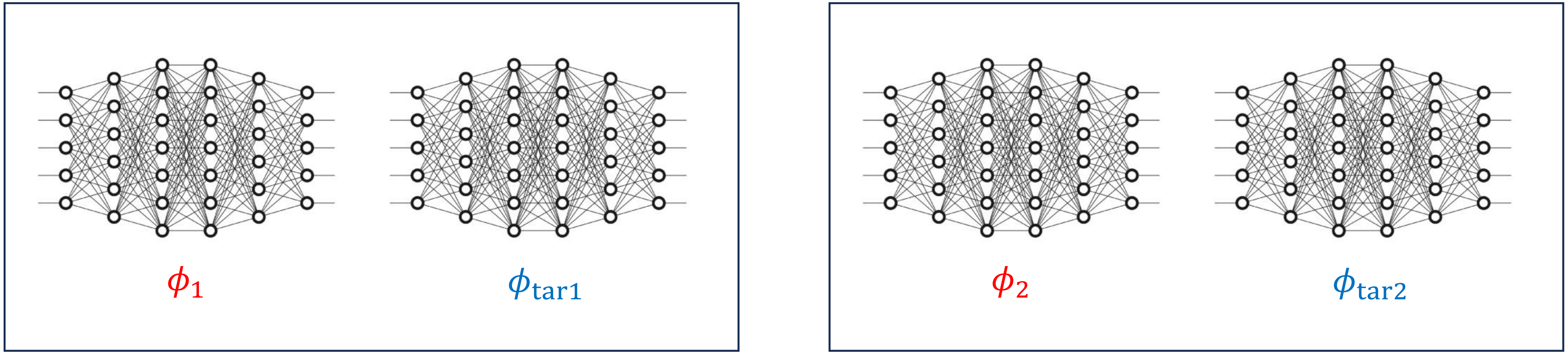
$$\phi_{\text{tar}} \leftarrow \tau\phi + (1-\tau)\phi_{\text{tar}}$$

Elements: replay buffer, target network, action noise

Lillicrap et al., Continuous control with deep reinforcement learning. 2015.

Further Stabilizing DDPG (1/3)

- Double Q-learning



Double Q-learning: When training ϕ_1, instead of using $Q_{\phi_{\text{tar1}}}$ to evaluate the regression target, use $Q_{\cancel{\phi_{\text{tar2}}}}$

TD3: $\min\left\{Q_{\phi_{\text{tar1}}}, Q_{\phi_{\text{tar2}}}\right\}$

Double Q-learning: Use independent samples to train ϕ_1 and ϕ_2

TD3: Use the same set of samples
(the independence between ϕ_1 and ϕ_2 only comes from random initialization)

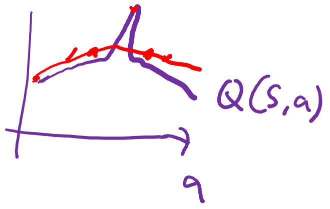
Further Stabilizing DDPG (2/3)

- Target policy smoothing

DDPG: use $Q_{\phi_{\text{tar}}}(s', \mu_{\theta_{\text{tar}}}(s'))$ as the regression target

TD3: sample $a' = \mu_{\theta_{\text{tar}}}(s') + \mathcal{N}(0, \sigma^2)$

use $Q_{\phi_{\text{tar}}}(s', a')$ as the regression target



Further Stabilizing DDPG (3/3)

- Delayed policy updates: running multiple steps of value updates before running one step of policy update

Twin Delayed DDPG (TD3)

For $k = 1,\ 2, \ldots$

Use $\mu_\theta(s) + \mathcal{N}(0, \sigma^2)$ to collect samples and place them in replay buffer

Sample a batch $\{(s_i, a_i, r_i, s_i')\}_{i=1}^n$ from the replay buffer

For each sample i, draw $a_i' \sim \mu_{\theta_{\text{tar}}}(s_i{}') + \mathcal{N}(0, \sigma^2 I)$

$$\phi_j \leftarrow \phi_j - \lambda \nabla_{\phi_j} \sum_{i=1}^{n} \left(Q_{\phi_j}(s_i, a_i) - r_i - \gamma \min_{\ell=1,2} Q_{\phi_{\text{tar}\ell}}(s_i', a_i') \right)^2 \qquad \forall j = 1,2$$

If $k \bmod M = 0$:

$$\theta \leftarrow \theta + \eta \sum_{i=1}^{n} \nabla_\theta Q_\phi\big(s_i, \mu_\theta(s_i)\big)$$

$$\theta_{\text{tar}} \leftarrow \tau\theta + (1-\tau)\theta_{\text{tar}}$$

$$\phi_{\text{tar}j} \leftarrow \tau\phi_j + (1-\tau)\phi_{\text{tar}j} \qquad \forall j = 1,2$$

Fujimoto et al., Addressing Function Approximation Error in Actor-Critic Methods. 2018.

Soft Actor-Critic (SAC)

- TD3 / DDPG: modeling a deterministic policy + additional noise for exploration
- SAC: modeling a randomized policy (by adding entropy as an exploration bonus)
- TD3 / DDPG vs. SAC is similar to ϵ-greedy vs. Boltzmann exploration

Entropy Bonus

Bandit

$H(\pi) = \boxed{-\sum_a \pi(a) \log \pi(a)}$

$$\pi = \arg\max_\pi \sum_a \pi(a) R(a) + \alpha\, H(\pi) = \arg\max_\pi \mathbb{E}_{a\sim\pi}[R(a) - \alpha \log \pi(a)]$$

$\log \frac{1}{\pi(a)}$

$\pi(a) = \exp\left(\frac{1}{\alpha} R(a)\right)$

MDP

$\sum_{h=0}^{\infty} \gamma^h \sum_a \pi(a|s_h) R(s_h, a) + \sum_{h=1}^{\infty} \gamma^h \alpha H(\pi(\cdot|s_h))$

$$\pi = \arg\max_\pi \mathbb{E}^\pi\left[\sum_{h=0}^{\infty} \gamma^h \left(\sum_a \pi(a|s_h) R(s_h, a) + \alpha\, H(\pi(\cdot\,|s_h))\right)\right]$$

$$= \arg\max_\pi \mathbb{E}^\pi\left[\sum_{h=0}^{\infty} \gamma^h \left(R(s_h, a_h) - \alpha \log \pi(a_h|s_h)\right)\right]$$

Bellman Equation with Entropy Bonus

$$Q^{\pi}(s,a) = \left[R(s,a) - \alpha \log \pi(a|s)\right] + \gamma \mathop{\mathbb{E}}_{s' \sim P(\cdot|s,a)} \left[V^{\pi}(s') \right]$$

(in SAC)

$$Q^{\pi}(s,a) = R(s,a) + \gamma \mathop{\mathbb{E}}_{s' \sim P(\cdot|s,a)} \left[V^{\pi}(s') + \alpha H(\pi(\cdot|s')) \right] \quad \checkmark$$

TD3 vs. SAC

- Value update

TD3: Sample $a' \sim \mu_\theta(s') + \mathcal{N}(0, \sigma^2)$

Use $Q_{\phi_{\text{tar}}}(s', a')$ as the regression target

SAC: Sample $a' \sim \pi_\theta(\cdot\,|s') = \mu_\theta(s') + \mathcal{N}(0, \sigma_\theta^2(s'))$

Use $Q_{\phi_{\text{tar}}}(s', a') - \alpha \log \pi_\theta\,(a'|s')$ as the regression target

Soft Actor-Critic (SAC)

For $k = 1,\ 2, \ldots$

Use $\mu_\theta(s) + \mathcal{N}(0, \sigma_\theta^2(s))$ to collect samples and place them in replay buffer

Sample a batch $\{(s_i, a_i, r_i, s_i')\}_{i=1}^n$ from the replay buffer

For each sample i, draw $a_i' \sim \mu_\theta(s_i{}') + \mathcal{N}(0, \sigma_\theta^2(s_i{}'))$

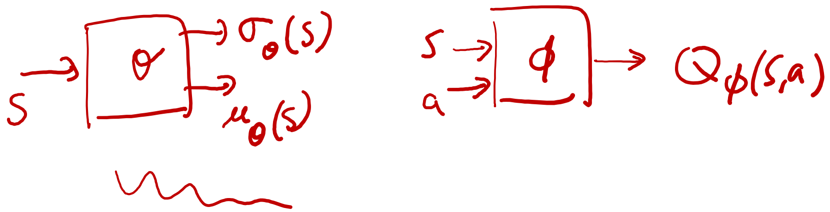
$$\phi_j \leftarrow \phi_j - \lambda \nabla_{\phi_j} \sum_{i=1}^{n} \left(Q_{\phi_j}(s_i, a_i) - r_i - \gamma \left(\min_{\ell=1,2} Q_{\phi_{\mathrm{tar}\ell}}(s_i', a_i') + \alpha \log \pi_\theta(a_i' | s_i{}') \right) \right)^2 \quad \forall j = 1,2$$

Perform Policy (θ) Update (to be specified later)

$$\phi_{\mathrm{tar}j} \leftarrow \tau \phi_j + (1 - \tau) \phi_{\mathrm{tar}j} \quad \forall j = 1,2$$

Haarnoja et al., Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. 2018.

TD3 vs. SAC



- Policy update

TD3: Do not view $-\alpha \log \pi_\theta \, (a|s)$ as part of the reward

Simply perform $\theta \leftarrow \theta + \eta \nabla_\theta Q_\phi(s, \mu_\theta(s))$

SAC: View $-\alpha \log \pi_\theta \, (a|s)$ as part of the reward

Perform the following:

Let $a_\theta(s) = \mu_\theta(s) + \epsilon \sigma_\theta(s)$ where $\epsilon \sim \mathcal{N}(0,1)$

Perform $\theta \leftarrow \theta + \eta \nabla_\theta \big(Q_\phi(s, a_\theta(s)) - \alpha \log \pi_\theta(a_\theta(s)|s)\big)$

$$\nabla_\theta \left(\int \pi_\theta(a|s) \, Q_\phi(s,a) \, da - \alpha \int \pi_\theta(a|s) \log \pi_\theta(a|s) \, da \right)$$

Policy Gradient with Entropy Bonus

$$\nabla_\theta \int_a \pi_\theta(a|s) \Big(Q_\phi(s,a) - \alpha \log \pi_\theta(a|s) \Big)\, da$$

$$\boxed{a \sim \pi_\theta(\cdot|s)} \Updownarrow \boxed{\begin{array}{l} \varepsilon \sim \mathcal{N}(0,1) \\ a = \mu_\theta(s) + \varepsilon\,\sigma_\theta(s) \end{array}}$$

$$= \nabla_\theta \, \mathbb{E}_{a\sim\pi(\cdot|s)} \Big[Q_\phi(s,a) - \alpha \log \pi_\theta(a|s) \Big]$$

$$= \nabla_\theta \, \mathbb{E}_{\varepsilon\sim \mathcal{N}(0,1)} \Big[Q_\phi\big(s, \mu_\theta(s) + \varepsilon\sigma_\theta(s)\big) - \alpha \log \pi_\theta\big(\mu_\theta(s) + \varepsilon\sigma_\theta(s) \,\big|\, s\big) \Big]$$

estimator can be constructed as:

① draw $\varepsilon \sim \mathcal{N}(0,1)$

② use $\nabla_\theta \Big[\qquad \Big]$

The Reparameterization Trick

1. inverse propensity weighting

$$\nabla_\theta \mathop{\mathbb{E}}_{a \sim \pi_\theta(\cdot|s)} \left[Q_\phi(s,a) - \alpha \log \pi_\theta(a|s) \right]$$

- First draw $\tilde{a} \sim \pi_\theta(\cdot|s)$
- construct $\dfrac{\nabla_\theta \pi_\theta(\tilde{a}|s)}{\pi_\theta(\tilde{a}|s)} \left[Q_\phi(s,\tilde{a}) - \alpha \log \pi_\theta(\tilde{a}|s) \right]$

2. Reparametrization

Soft Actor-Critic (SAC)

Further using $\pi_\theta(a|s) = \frac{1}{(2\pi\sigma_\theta(s)^2)^{d/2}} \exp\left(-\frac{\|a-\mu_\theta(s)\|^2}{\sigma_\theta(s)^2}\right)$

For $k = 1,\ 2, \ldots$

Use $\mu_\theta(s) + \mathcal{N}(0, \sigma_\theta^2(s))$ to collect samples and place them in replay buffer

Sample a batch $\{(s_i, a_i, r_i, s_i')\}_{i=1}^n$ from the replay buffer

For each sample i, draw $a_i' \sim \mu_\theta({s_i}') + \mathcal{N}(0, \sigma_\theta^2({s_i}'))$

$$\phi_j \leftarrow \phi_j - \lambda \nabla_{\phi_j} \sum_{i=1}^{n} \left(Q_{\phi_j}(s_i, a_i) - r_i - \gamma \left(\min_{\ell=1,2} Q_{\phi_{\text{tar}\ell}}(s_i', a_i') + \alpha \log \pi_\theta(a_i' | {s_i}') \right) \right)^2 \quad \forall j = 1,2$$

Let $a_\theta(s_i) = \mu_\theta(s_i) + \epsilon \sigma_\theta(s_i)$ where $\epsilon \sim \mathcal{N}(0, I)$

$$\theta \leftarrow \theta + \eta \sum_{i=1}^{n} \nabla_\theta \left(Q_\phi(s, a_\theta(s_i)) - \alpha \log \pi_\theta(a_\theta(s_i) | s_i) \right)$$

$$\phi_{\text{tar}j} \leftarrow \tau \phi_j + (1-\tau)\phi_{\text{tar}j} \quad \forall j = 1,2$$

Haarnoja et al., Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. 2018.