# 13 - Policy Gradient Methods:

Jan 26, 2022

All methods thus far used action-value methods.
Now we use value functions to learn a parameterized policy, which can select actions without using a value function.

$\vec{\theta}$: policy's parameter vector, $\vec{\theta} \in \mathbb{R}^{d'}$

$\pi(a|s, \vec{\theta}) = Pr\{A_t = a | S_t = s, \vec{\theta}_t = \vec{\theta}\}$

↳ Probability that action $a$ is taken at time $t$ in state $s$ w/ parameter $\vec{\theta}$.

## Policy Gradients use Gradient Ascent:

$$\vec{\theta}_{t+1} = \vec{\theta}_t + \alpha \widehat{\nabla J(\vec{\theta}_t)}$$

- All methods that follow this schema are policy gradient methods.
- Methods that learn both policy and value functions are actor-critic methods.

## Constraints on Policy Parameterization:

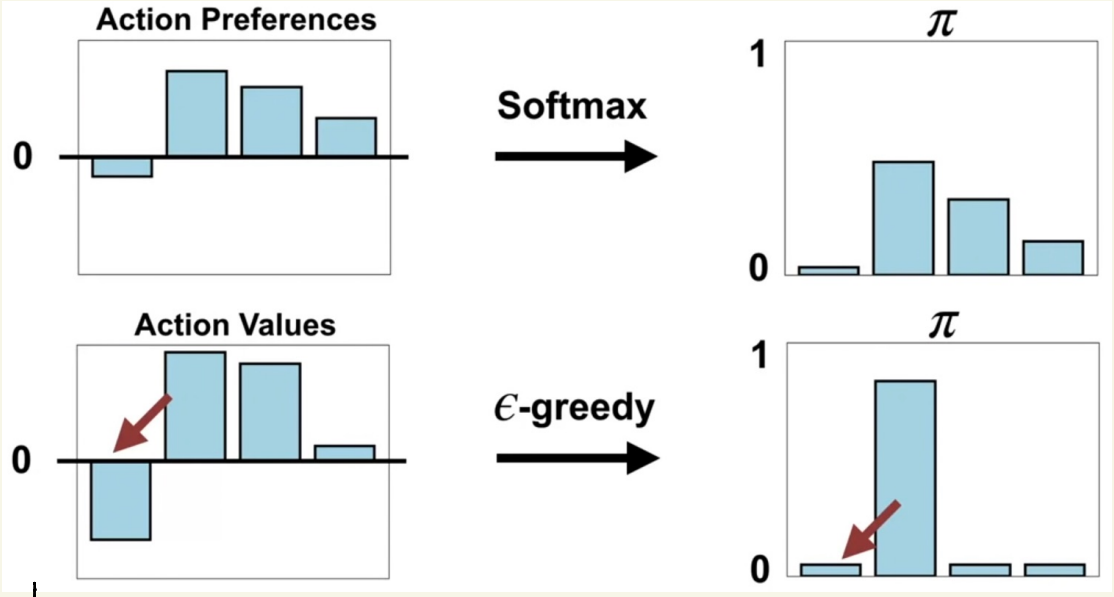
1. $\pi(a|s, \vec{\theta}) \geq 0$ for all $a \in A$ and $s \in S$
2. $\sum_{a \in A} \pi(a|s, \vec{\theta}) = 1$ for all $s \in S$

## The Softmax Policy Parameterization:

$$\pi(a|s, \vec{\theta}) = \frac{e^{h(s,a,\vec{\theta})}}{\sum_{b \in A} e^{h(s,b,\vec{\theta})}}$$

→ $e$ action preference

→ normalizes output




↳ For $\epsilon$-greedy, even if the agent learns that an action has terrible consequences, it still selects it more often relative to softmax.

↳ Cuz all actions except the best have equal prob.

## Formalizing the Goal as an Objective:

Episodic: $G_t = \sum_{t=0}^{T} R_t$

Continuing: $G_t = \sum_{t=0}^{\infty} \gamma^t R_t$ , $G_t = \sum_{t=0}^{\infty} R_t - r(\pi)$

## The Average Reward Objective:

$$r(\pi) = \sum_s \mu_\pi(s) \sum_a \pi(a|s,\vec{\theta}) \sum_{s',r} p(s',r|s,a) r$$

## Optimizing the Average Reward Objective:

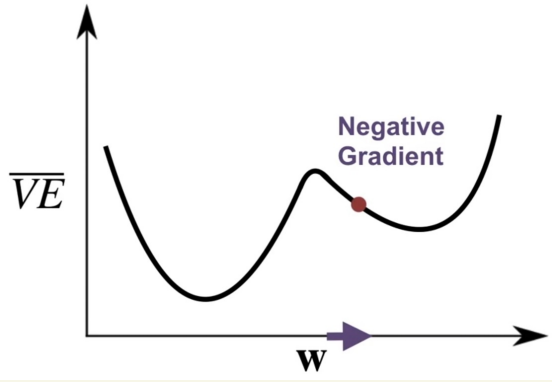
$$\nabla r(\pi) = \nabla \sum_s \mu_\pi(s) \sum_a \pi(a|s,\vec{\theta}) \sum_{s',r} p(s',r|s,a) r$$

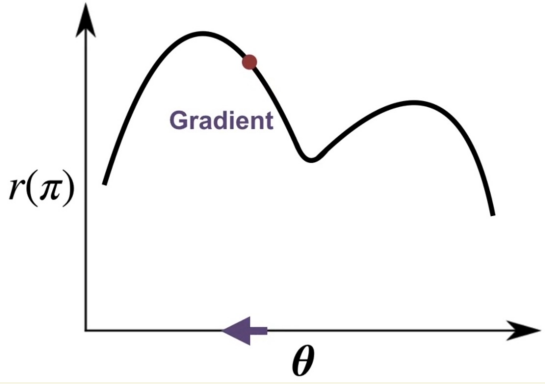
↳ Policy gradient method

※ But here, $\nabla \sum_s \mu_\pi(s)$ depends on $\vec{\theta}$ that we are optimizing.

↳ This is solved by the Policy Gradient Theorem

## Gradient Descent vs Ascent:



# The Gradient of the Objective!

$$\nabla r(\pi) = \nabla \sum_s \mu(s) \sum_a \pi(a \mid s, \boldsymbol{\theta}) \sum_{s',r} p(s', r \mid s, a) r$$

$$= \sum_s \boxed{\nabla \mu(s)} \sum_a \pi(a \mid s, \boldsymbol{\theta}) \sum_{s',r} p(s', r \mid s, a) r$$

$$+ \sum_s \mu(s) \nabla \sum_a \pi(a \mid s, \boldsymbol{\theta}) \sum_{s',r} p(s', r \mid s, a) r$$

→ how do we get $\nabla \mu(s)$?

# Result of Policy Gradient Theorem!

$$\nabla r(\pi) = \sum_s \mu(s) \sum_a \nabla \pi(a|s, \vec{\theta}) q_\pi(s, a)$$

# Stochastic Gradient Ascent for Policy Parameters!

$$\vec{\theta}_{t+1} \equiv \vec{\theta}_t + \alpha \nabla \ln \pi(A_t|S_t, \vec{\theta}_t) q_\pi(S_t, A_t)$$

↳ $\nabla \ln(f(x)) = \frac{\nabla f(x)}{f(x)}$

✱ Used for easier computation.

# Proof of Policy Gradient Theorem!

**Proof of the Policy Gradient Theorem (episodic case)**

With just elementary calculus and re-arranging of terms, we can prove the policy gradient theorem from first principles. To keep the notation simple, we leave it implicit in all cases that $\pi$ is a function of $\boldsymbol{\theta}$, and all gradients are also implicitly with respect to $\boldsymbol{\theta}$. First note that the gradient of the state-value function can be written in terms of the action-value function as

$$
\begin{aligned}
\nabla v_\pi(s) &= \nabla \left[ \sum_a \pi(a|s) q_\pi(s,a) \right], \quad \text{for all } s \in \mathcal{S} && \text{(Exercise 3.18)} \\
&= \sum_a \Big[ \nabla \pi(a|s) q_\pi(s,a) + \pi(a|s) \nabla q_\pi(s,a) \Big] && \text{(product rule of calculus)} \\
&= \sum_a \Big[ \nabla \pi(a|s) q_\pi(s,a) + \pi(a|s) \nabla \sum_{s',r} p(s',r|s,a) \big(r + v_\pi(s')\big) \Big] \\
& && \text{(Exercise 3.19 and Equation 3.2)} \\
&= \sum_a \Big[ \nabla \pi(a|s) q_\pi(s,a) + \pi(a|s) \sum_{s'} p(s'|s,a) \nabla v_\pi(s') \Big] && \text{(Eq. 3.4)} \\
&= \sum_a \Big[ \nabla \pi(a|s) q_\pi(s,a) + \pi(a|s) \sum_{s'} p(s'|s,a) && \text{(unrolling)} \\
&\qquad \sum_{a'} \big[ \nabla \pi(a'|s') q_\pi(s',a') + \pi(a'|s') \sum_{s''} p(s''|s',a') \nabla v_\pi(s'') \big] \Big] \\
&= \sum_{x \in \mathcal{S}} \sum_{k=0}^{\infty} \Pr(s \to x, k, \pi) \sum_a \nabla \pi(a|x) q_\pi(x,a),
\end{aligned}
$$

after repeated unrolling, where $\Pr(s \to x, k, \pi)$ is the probability of transitioning from state $s$ to state $x$ in $k$ steps under policy $\pi$. It is then immediate that

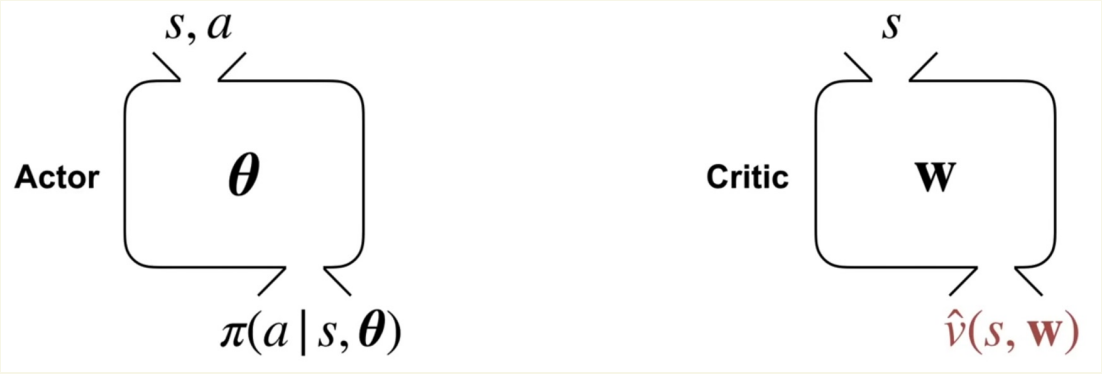
$$
\begin{aligned}
\nabla J(\boldsymbol{\theta}) &= \nabla v_\pi(s_0) \\
&= \sum_s \left( \sum_{k=0}^{\infty} \Pr(s_0 \to s, k, \pi) \right) \sum_a \nabla \pi(a|s) q_\pi(s,a) \\
&= \sum_s \eta(s) \sum_a \nabla \pi(a|s) q_\pi(s,a) && \text{(box page 199)} \\
&= \sum_{s'} \eta(s') \sum_s \frac{\eta(s)}{\sum_{s'} \eta(s')} \sum_a \nabla \pi(a|s) q_\pi(s,a) \\
&= \sum_{s'} \eta(s') \sum_s \mu(s) \sum_a \nabla \pi(a|s) q_\pi(s,a) && \text{(Eq. 9.3)} \\
&\propto \sum_s \mu(s) \sum_a \nabla \pi(a|s) q_\pi(s,a) && \text{(Q.E.D.)}
\end{aligned}
$$

# 13.5 - Actor-Critic Methods:

Parameterized policy plays the role of an actor.
Value function plays the role of a critic.

⊗ The idea is to learn the parameters and the value function together.

$$\vec{\theta}_{t+1} \equiv \vec{\theta}_t + \alpha \nabla \ln \pi(A_t | S_t, \vec{\theta}_t)\left[R_{t+1} - \bar{R} + \hat{v}(S_{t+1}, \vec{w})\right]$$



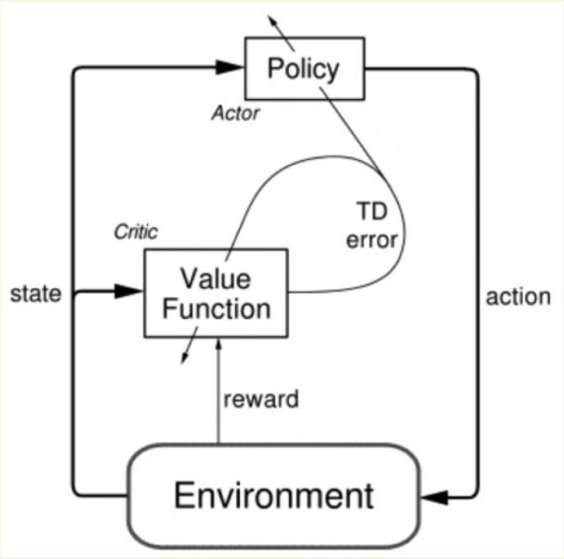

↳ Critic trained w/ average reward semi-gradient TD(0)

## Subtract a Baseline!

$$\vec{\theta}_{t+1} \equiv \vec{\theta}_t + \alpha \nabla \ln \pi(A_t | S_t, \vec{\theta}_t)\left[R_{t+1} - \bar{R} + \hat{v}(S_{t+1}, \vec{w}) - \hat{v}(S_t, \vec{w})\right]$$

TD Error δ

# Actor-Critic Interaction:




Actor changes policy to exceed critic's expectation.

Critic is updating value function to evaluate actor's changing policy.

Kinda like GAN, but actor is not completely dependent on critic.

**Actor-Critic (continuing), for estimating $\pi_\theta \approx \pi_*$**

Input: a differentiable policy parameterization $\pi(a \mid s, \boldsymbol{\theta})$

Input: a differentiable state-value function parameterization $\hat{v}(s, \mathbf{w})$

Initialize $\bar{R} \in \mathbb{R}$ to $0$

Initialize state-value weights $\mathbf{w} \in \mathbb{R}^d$ and policy parameter $\boldsymbol{\theta} \in \mathbb{R}^{d'}$ (e.g. to $0$)

Algorithm parameters: $\alpha^{\mathbf{w}} > 0$, $\alpha^{\boldsymbol{\theta}} > 0$, $\alpha^{\bar{R}} > 0$

Initialize $S \in \mathcal{S}$

Loop forever (for each time step):

$A \sim \pi(\cdot \mid S, \boldsymbol{\theta})$

Take action $A$, observe $S'$, $R$

$\delta \leftarrow R - \bar{R} + \hat{v}(S', \mathbf{w}) - \hat{v}(S, \mathbf{w})$

$\bar{R} \leftarrow \bar{R} + \alpha^{\bar{R}} \delta$

$\mathbf{w} \leftarrow \mathbf{w} + \alpha^{\mathbf{w}} \delta \nabla \hat{v}(S, \mathbf{w})$

$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha^{\boldsymbol{\theta}} \delta \nabla \ln \pi(A \mid S, \boldsymbol{\theta})$

$S \leftarrow S'$

## Actor-Critic

$$\vec{w} \leftarrow \vec{w} + \alpha^{\vec{w}} \delta \nabla \hat{v}(S, \vec{w})$$

$$\vec{\theta} \leftarrow \vec{\theta} + \alpha^{\vec{\theta}} \delta \nabla \ln \pi(A|S, \vec{\theta})$$

## Policy Update with a Softmax Policy:

1. $$\pi(a|s, \vec{\theta}) \equiv \frac{e^{h(s,a,\vec{\theta})}}{\sum_{b \in A} e^{h(s,a,\vec{\theta})}}$$

2. $$\nabla \hat{v}(s, \vec{w}) = \vec{x}(s)$$

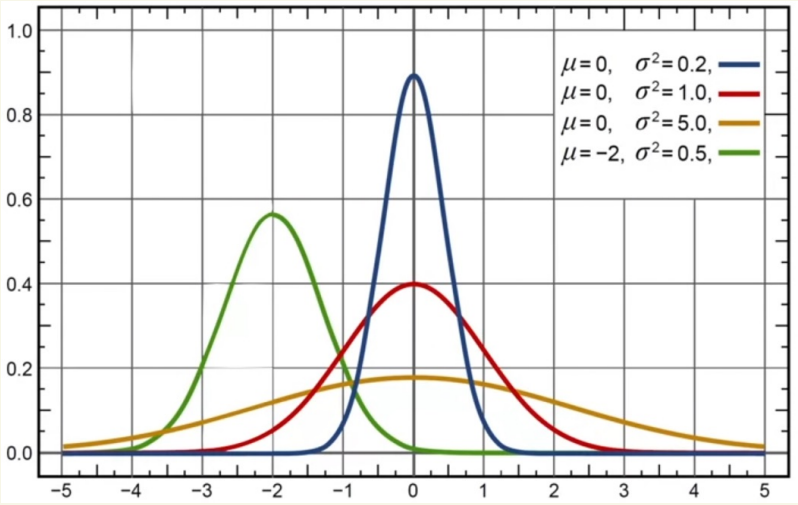
   ↳ $$\vec{w} \leftarrow \vec{w} + \alpha^{\vec{w}} \delta \vec{x}(S)$$

3. $$\nabla \ln \pi(a|s, \vec{\theta}) = \vec{x}_h(s,a) - \sum_b \pi(b|s, \vec{\theta}) \vec{x}_h(s,b)$$

   ↳ $$\vec{\theta} \leftarrow \vec{\theta} + \alpha^{\vec{\theta}} \delta \left[ \vec{x}_h(s,a) - \sum_b \pi(b|s, \vec{\theta}) \vec{x}_h(s,b) \right]$$

## Gaussian Distribution:

$$p(x) \equiv \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$




## Gaussian Policy:

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

$$\mu(s,\vec{\theta}) \equiv \vec{\theta}_\mu^T \vec{x}(s)$$

$$\sigma(s,\vec{\theta}) \equiv \exp(\vec{\theta}_\sigma^T \vec{x}(s))$$

$$\vec{\theta} \equiv \begin{bmatrix} \vec{\theta}_\mu \\ \vec{\theta}_\sigma \end{bmatrix}$$

- Parameterizing the policy as a continuous distribution helps it select from a range of non-discrete possibilities.
  - ↳ It might not always be good to select from a discrete set of actions.

Gradients:

$$\nabla \ln \pi(a|s, \vec{\theta}_\mu) = \frac{1}{\sigma(s, \vec{\theta})^2}\left(a - \mu(s, \vec{\theta})\right)\vec{x}(s)$$

$$\nabla \ln \pi(a|s, \vec{\theta}_\sigma) = \left(\frac{\left(a - \mu(s, \vec{\theta})\right)^2}{\sigma(s, \vec{\theta})^2} - 1\right)\vec{x}(s)$$