Advanced Policy Optimization

Sham Kakade and Kianté Brantley

CS 2824: Foundations of Reinforcement Learning

Given two policies π and π', what is the performance difference: $V^{\pi}(s_0) - V^{\pi'}(s_0) = ??$

Given two policies π and π', what is the performance difference: $V^{\pi}(s_0) - V^{\pi'}(s_0) = ??$

(Diff in performances $\Leftarrow \Rightarrow$ Diff in policies?)

Setting and Notation

Discounted infinite horizon MDP:

$$\mathscr{M} = \{S, A, \gamma, r, P\}$$

State visitation: $d^{\pi}_{s_0}(s) = (1-\gamma)\sum_{h=0}^{\infty} \gamma^h \mathbb{P}^{\pi}_h(s; s_0)$

Setting and Notation

Discounted infinite horizon MDP:

$$\mathscr{M} = \{S, A, \gamma, r, P\}$$

State visitation: $d^{\pi}_{s_0}(s) = (1-\gamma)\sum_{h=0}^{\infty} \gamma^h \mathbb{P}^{\pi}_h(s; s_0)$

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

(The “advantage” of deviating from π for one and only one step)

Setting and Notation

Discounted infinite horizon MDP:

$$\mathscr{M} = \{S, A, \gamma, r, P\}$$

State visitation: $d^{\pi}_{s_0}(s) = (1-\gamma)\sum_{h=0}^{\infty} \gamma^h \mathbb{P}^{\pi}_h(s; s_0)$

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

(The “advantage” of deviating from π for one and only one step)

(Quick sanity check: $A^{\pi}(s, \pi(s)) = 0$)

Setting and Notation

Discounted infinite horizon MDP:

$$\mathscr{M} = \{S, A, \gamma, r, P\}$$

State visitation: $d^{\pi}_{s_0}(s) = (1-\gamma)\sum_{h=0}^{\infty} \gamma^h \mathbb{P}^{\pi}_h(s; s_0)$

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

(The "advantage" of deviating from π for one and only one step)

(Quick sanity check: $A^{\pi}(s, \pi(s)) = 0$)

Recall PI:

$$\arg\max_a Q^{\pi}(s,a) = \arg\max_a A^{\pi}(s,a),$$

i.e., Policy-improve step seeks the action that has the **largest adv**

PDL:

Given two policies $\pi : S \mapsto \Delta(A),\ \pi' : S \mapsto \Delta(A)$, recall $V^{\pi}(s_0) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \,|\, \pi\right]$

PDL:

Given two policies $\pi : S \mapsto \Delta(A),\ \pi' : S \mapsto \Delta(A)$, recall $V^{\pi}(s_0) = \mathbb{E}\left[\sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \,|\, \pi\right]$

Performance Difference Lemma (PDL):

$$V^{\pi}(s_0) - V^{\pi'}(s_0) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi}_{s_0}}\left[\mathbb{E}_{a\sim\pi(\cdot|s)}Q^{\pi'}(s,a) - V^{\pi'}(s)\right]$$

$$:= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi}_{s_0}}\left[\mathbb{E}_{a\sim\pi(\cdot|s)}A^{\pi'}(s,a)\right]$$

PDL Explanation

$$V^{\pi}(s_0) - V^{\pi'}(s_0) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi}_{s_0}}\left[\mathbb{E}_{a\sim\pi(\cdot|s)}A^{\pi'}(s,a)\right]$$

PDL Proof

$$V^{\pi}(s_0) - V^{\pi'}(s_0) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi}_{s_0}}\left[\mathbb{E}_{a\sim\pi(\cdot|s)}A^{\pi'}(s,a)\right]$$

Proof of Sketch (see reading material for detailed steps)

PDL Proof

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

Proof of Sketch (see reading material for detailed steps)

$$V^{\pi}(s_0) - V^{\pi'}(s_0)$$

$$= V^{\pi}(s_0) - \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[r(s_0,a_0) + \gamma\mathbb{E}_{s'\sim P(s_0,a_0)}V^{\pi'}(s')\right] + \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[r(s_0,a_0) + \gamma\mathbb{E}_{s'\sim P(s_0,a_0)}V^{\pi'}(s')\right] - V^{\pi'}(s_0)$$

PDL Proof

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

Proof of Sketch (see reading material for detailed steps)

$$V^{\pi}(s_0) - V^{\pi'}(s_0)$$

$$= V^{\pi}(s_0) - \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[r(s_0,a_0) + \gamma\mathbb{E}_{s'\sim P(s_0,a_0)}V^{\pi'}(s')\right] + \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[r(s_0,a_0) + \gamma\mathbb{E}_{s'\sim P(s_0,a_0)}V^{\pi'}(s')\right] - V^{\pi'}(s_0)$$

$$= \gamma\mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\mathbb{E}_{s_1\sim P(s_0,a_0)}\left[V^{\pi}(s_1) - V^{\pi'}(s_1)\right] + \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[r(s_0,a_0) + \gamma\mathbb{E}_{s'\sim P(s_0,a_0)}V^{\pi'}(s')\right] - V^{\pi'}(s_0)$$

PDL Proof

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

Proof of Sketch (see reading material for detailed steps)

$$V^{\pi}(s_0) - V^{\pi'}(s_0)$$

$$= V^{\pi}(s_0) - \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[r(s_0,a_0) + \gamma\mathbb{E}_{s'\sim P(s_0,a_0)}V^{\pi'}(s')\right] + \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[r(s_0,a_0) + \gamma\mathbb{E}_{s'\sim P(s_0,a_0)}V^{\pi'}(s')\right] - V^{\pi'}(s_0)$$

$$= \gamma\mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\mathbb{E}_{s_1\sim P(s_0,a_0)}\left[V^{\pi}(s_1) - V^{\pi'}(s_1)\right] + \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[r(s_0,a_0) + \gamma\mathbb{E}_{s'\sim P(s_0,a_0)}V^{\pi'}(s')\right] - V^{\pi'}(s_0)$$

$$= \gamma\mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\mathbb{E}_{s_1\sim P(s_0,a_0)}\left[V^{\pi}(s_1) - V^{\pi'}(s_1)\right] + \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[Q^{\pi'}(s_0,a_0) - V^{\pi'}(s_0)\right]$$

PDL Proof

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

Proof of Sketch (see reading material for detailed steps)

$$V^{\pi}(s_0) - V^{\pi'}(s_0)$$

$$= V^{\pi}(s_0) - \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[r(s_0,a_0) + \gamma\mathbb{E}_{s'\sim P(s_0,a_0)}V^{\pi'}(s')\right] + \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[r(s_0,a_0) + \gamma\mathbb{E}_{s'\sim P(s_0,a_0)}V^{\pi'}(s')\right] - V^{\pi'}(s_0)$$

$$= \gamma\mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\mathbb{E}_{s_1\sim P(s_0,a_0)}\left[V^{\pi}(s_1) - V^{\pi'}(s_1)\right] + \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[r(s_0,a_0) + \gamma\mathbb{E}_{s'\sim P(s_0,a_0)}V^{\pi'}(s')\right] - V^{\pi'}(s_0)$$

$$= \gamma\mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\mathbb{E}_{s_1\sim P(s_0,a_0)}\left[V^{\pi}(s_1) - V^{\pi'}(s_1)\right] + \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[Q^{\pi'}(s_0,a_0) - V^{\pi'}(s_0)\right]$$

$$= \gamma\mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\mathbb{E}_{s_1\sim P(s_0,a_0)}\left[V^{\pi}(s_1) - V^{\pi'}(s_1)\right] + \mathbb{E}_{a_0\sim\pi(\cdot|s_0)}\left[A^{\pi'}(s_0,a_0)\right]$$

Outline for Today:

1. Trust Region Policy Optimization

2. Proximal Policy Optimization (PPO)

3. Group Relative Policy Optimization

Trust Region Policy Optimization

Want to manipulate $V^{\pi_\theta}(s_0)$ **into an objective** that we can estimate from sampled data

$$V^{\pi_\theta}(s_0) - V^{\pi_{\theta_t}}(s_0) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi_\theta}}\left[\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}A^{\pi_{\theta_t}}(s,a)\right]$$

Trust Region Policy Optimization

Want to manipulate $V^{\pi_\theta}(s_0)$ **into an objective** that we can estimate from sampled data

$$V^{\pi_\theta}(s_0) - V^{\pi_{\theta_t}}(s_0) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi_\theta}}\left[\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}A^{\pi_{\theta_t}}(s,a)\right]$$

$$V^{\pi_\theta}(s_0) = V^{\pi_{\theta_t}}(s_0) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi_\theta}}\left[\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}A^{\pi_{\theta_t}}(s,a)\right]$$

Define $L_{\theta_t}(\theta)$ to be the "**surrogate objective**" that ignores change in state distribution:

Trust Region Policy Optimization

Want to manipulate $V^{\pi_\theta}(s_0)$ **into an objective** that we can estimate from sampled data

$$V^{\pi_\theta}(s_0) - V^{\pi_{\theta_t}}(s_0) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi_\theta}}\left[\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}A^{\pi_{\theta_t}}(s,a)\right]$$

$$V^{\pi_\theta}(s_0) = V^{\pi_{\theta_t}}(s_0) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi_\theta}}\left[\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}A^{\pi_{\theta_t}}(s,a)\right]$$

Define $L_{\theta_t}(\theta)$ to be the "**surrogate objective**" that ignores change in state distribution:

$$L_{\theta_t}(\theta) := V^{\pi_{\theta_t}}(s_0) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi_{\theta_t}}}\left[\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}A^{\pi_{\theta_t}}(s,a)\right]$$

Trust Region Policy Optimization

Want to manipulate $V^{\pi_\theta}(s_0)$ **into an objective** that we can estimate from sampled data

$$V^{\pi_\theta}(s_0) - V^{\pi_{\theta_t}}(s_0) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi_\theta}}\left[\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}A^{\pi_{\theta_t}}(s,a)\right]$$

$$V^{\pi_\theta}(s_0) = V^{\pi_{\theta_t}}(s_0) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi_\theta}}\left[\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}A^{\pi_{\theta_t}}(s,a)\right]$$

Define $L_{\theta_t}(\theta)$ to be the "**surrogate objective**" that ignores change in state distribution:

$$L_{\theta_t}(\theta) := V^{\pi_{\theta_t}}(s_0) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi_{\theta_t}}}\left[\mathbb{E}_{a\sim\pi_\theta(\cdot|s)}A^{\pi_{\theta_t}}(s,a)\right]$$

$$L_{\theta_t}(\theta) := V^{\pi_{\theta_t}}(s_0) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_{\theta_t}},\, a\sim\pi_{\theta_t}}\left[\frac{\pi_\theta(a\,|\,s)}{\pi_{\theta_t}(a\,|\,s)}A^{\pi_{\theta_t}}(s,a)\right].$$

Trust Region Policy Optimization

Define $L_{\theta_0}(\theta)$ to be the "**surrogate objective**" that ignores change in state distribution:

$$L_{\theta_t}(\theta) := V^{\pi_{\theta_t}}(s_0) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_{\theta_t}},\, a\sim\pi_{\theta_t}}\left[\frac{\pi_\theta(a\,|\,s)}{\pi_{\theta_t}(a\,|\,s)}A^{\pi_{\theta_t}}(s,a)\right].$$

Trust Region Policy Optimization

Define $L_{\theta_0}(\theta)$ to be the "**surrogate objective**" that ignores change in state distribution:

$$L_{\theta_t}(\theta) := V^{\pi_{\theta_t}}(s_0) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi_{\theta_t}},\, a\sim\pi_{\theta_t}}\left[\frac{\pi_\theta(a\,|\,s)}{\pi_{\theta_t}(a\,|\,s)}A^{\pi_{\theta_t}}(s,a)\right].$$

We know the first order Taylor expansion of $L_{\theta_t}(\theta)$

$$L_{\theta_t}(\theta) \approx L_{\theta_t}(\theta_t) + \nabla_\theta L_{\theta_t}(\theta_t)^\top(\theta-\theta_t)$$

Recap on Natural Policy Gradient:

At iteration t:

$$\max_{\pi_\theta} \mathbb{E}_{s\sim d_\mu^{\pi_{\theta_t}}} \left[\mathbb{E}_{a\sim\pi_\theta(s)} A^{\pi_{\theta_t}}(s,a) \right]$$

$$\text{s.t., } KL\left(\rho_{\pi_{\theta_t}} | \rho_{\pi_\theta}\right) \leq \delta$$

Intuition: maximize local adv subject to being incremental (in KL);

Recap on Natural Policy Gradient:

At iteration t:

$$\max_{\pi_\theta} \mathbb{E}_{s\sim d_\mu^{\pi_{\theta_t}}} \left[\mathbb{E}_{a\sim\pi_\theta(s)} A^{\pi_{\theta_t}}(s,a) \right] \longrightarrow$$ First-order Taylor expansion at θ_t

$$\text{s.t., } KL\left(\rho_{\pi_{\theta_t}} | \rho_{\pi_\theta}\right) \leq \delta \longrightarrow$$ second-order Taylor expansion at θ_t

Intuition: maximize local adv subject to being incremental (in KL);

Recap on Natural Policy Gradient:

At iteration t:

$$\max_{\pi_\theta} \mathbb{E}_{s\sim d_\mu^{\pi_{\theta_t}}} \left[\mathbb{E}_{a\sim\pi_\theta(s)} A^{\pi_{\theta_t}}(s,a) \right]$$

→ First-order Taylor expansion at θ_t

$$\text{s.t., } KL\left(\rho_{\pi_{\theta_t}} | \rho_{\pi_\theta}\right) \leq \delta$$

→ second-order Taylor expansion at θ_t

Intuition: maximize local adv subject to being incremental (in KL);

$$\max_\theta \nabla_\theta J(\pi_{\theta_t})^\top (\theta - \theta_t)$$
$$\text{s.t. } (\theta - \theta_t)^\top F_{\theta_t} (\theta - \theta_t) \leq \delta$$

Recap on Natural Policy Gradient:

At iteration t:

$$\max_{\pi_\theta} \mathbb{E}_{s\sim d_\mu^{\pi_{\theta_t}}}\left[\mathbb{E}_{a\sim\pi_\theta(s)} A^{\pi_{\theta_t}}(s,a)\right] \longrightarrow \text{First-order Taylor expansion at } \theta_t$$

$$\text{s.t., } KL\left(\rho_{\pi_{\theta_t}} | \rho_{\pi_\theta}\right) \leq \delta \longrightarrow \text{second-order Taylor expansion at } \theta_t$$

Intuition: maximize local adv subject to being incremental (in KL);

$$\max_\theta \nabla_\theta J(\pi_{\theta_t})^\top(\theta-\theta_t)$$
$$\text{s.t. } (\theta-\theta_t)^\top F_{\theta_t}(\theta-\theta_t) \leq \delta$$

$$\theta_{t+1} = \theta_t + \eta F_{\theta_t}^{-1} \nabla_\theta J(\pi_{\theta_t})$$

NPG

Recap on Natural Policy Gradient:

At iteration t:

$$\max_{\pi_\theta} \mathbb{E}_{s\sim d_\mu^{\pi_{\theta_t}}}\left[\mathbb{E}_{a\sim\pi_\theta(s)} A^{\pi_{\theta_t}}(s,a)\right] \longrightarrow \text{First-order Taylor expansion at } \theta_t$$

$$\text{s.t., } KL\left(\rho_{\pi_{\theta_t}} | \rho_{\pi_\theta}\right) \le \delta \longrightarrow \text{second-order Taylor expansion at } \theta_t$$

Intuition: maximize local adv subject to being incremental (in KL);

$$\max_\theta \nabla_\theta J(\pi_{\theta_t})^\top(\theta-\theta_t)$$
$$\text{s.t. } (\theta-\theta_t)^\top F_{\theta_t}(\theta-\theta_t) \le \delta$$

$$\theta_{t+1} = \theta_t + \eta F_{\theta_t}^{-1}\nabla_\theta J(\pi_{\theta_t})$$

NPG

$$F_{\theta_t} := \mathbb{E}_{s,a\sim d_\mu^{\pi_{\theta_t}}}\left[\nabla_\theta \ln \pi_{\theta_t}(a|s)\left(\nabla_\theta \ln \pi_{\theta_t}(a|s)\right)^\top\right] \in \mathbb{R}^{dim_\theta \times dim_\theta}$$

Policy Gradient (e.g., REINFORCE) can unstable and slow

The potential high-variance in PG can make learning very unstable

Policy Gradient (e.g., REINFORCE) can unstable and slow

The potential high-variance in PG can make learning very unstable

Natural Policy gradient is computational expensive

Even compute fisher information matrix is slow

Outline for Today:

1. Trust Region Policy Optimization

2. Proximal Policy Optimization (PPO)

An extension of NPG (even faster in practice)

Given a current policy π^t, we perform policy update to π^{t+1}

Proximal Policy Optimization (PPO)

$$\max_{\theta} \; \mathbb{E}_{s\sim d_\mu^{\pi_{\theta_t}}}\left[\mathbb{E}_{a\sim\pi_\theta(\cdot|s)} A^{\pi_{\theta_t}}(s,a)\right] - \lambda \underbrace{\mathbb{E}_{s\sim d_\mu^{\pi_t}}\left[\mathrm{KL}\Big(\pi_{\theta_t}(a \mid s) \,\|\, \pi_\theta(a \mid s)\Big)\right]}_{\text{regularization}}$$

Proximal Policy Optimization (PPO)

Construct a **batch Supervised Learning** style objective using $\mathscr{D} = \{s, a, A^{\pi_{\theta_t}}(s, a)\}$

$$\max_{\theta} \ell(\theta) = \max_{\theta} \mathbb{E}_{s \sim d^{\pi_{\theta_t}}} \mathbb{E}_{a \sim \pi_{\theta}(\cdot|s)} \cdot A^{\pi_{\theta_t}}(s, a)$$

Proximal Policy Optimization (PPO)

Construct a **batch Supervised Learning** style objective using $\mathscr{D} = \{s, a, A^{\pi_{\theta_t}}(s, a)\}$

$$\max_{\theta} \ell(\theta) = \max_{\theta} \mathbb{E}_{s\sim d^{\pi_{\theta_t}}} \mathbb{E}_{a\sim\pi_\theta(\cdot|s)} \cdot A^{\pi_{\theta_t}}(s, a)$$

IW trick $$\rightarrow \mathbb{E}_{s\sim d^{\pi_{\theta_t}}} \mathbb{E}_{a\sim\pi_{\theta_t}(\cdot|s)} \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s, a)$$

Proximal Policy Optimization (PPO)

Construct a **batch Supervised Learning** style objective using $\mathscr{D} = \{s, a, A^{\pi_{\theta_t}}(s, a)\}$

$$\max_{\theta} \ell(\theta) = \max_{\theta} \mathbb{E}_{s \sim d^{\pi_{\theta_t}}} \mathbb{E}_{a \sim \pi_\theta(\cdot|s)} \cdot A^{\pi_{\theta_t}}(s, a)$$

IW trick $$\rightarrow \mathbb{E}_{s \sim d^{\pi_{\theta_t}}} \mathbb{E}_{a \sim \pi_{\theta_t}(\cdot|s)} \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s, a)$$

$$\approx \frac{1}{N} \sum_{s,a} \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s, a)$$

Proximal Policy Optimization (PPO)

Construct a **batch Supervised Learning** style objective using $\mathscr{D} = \{s, a, A^{\pi_{\theta_t}}(s, a)\}$

$$\hat{\ell}(\theta) = \sum_{s,a} \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s, a)$$

Proximal Policy Optimization (PPO)

Construct a **batch Supervised Learning** style objective using $\mathscr{D} = \{s, a, A^{\pi_{\theta_t}}(s, a)\}$

$$\hat{\ell}(\theta) = \sum_{s,a} \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s, a)$$

Trick 1: clipping to make sure π_θ stay close to π_{θ_t} (ensuring stability in training)

Proximal Policy Optimization (PPO)

Construct a **batch Supervised Learning** style objective using $\mathscr{D} = \{s, a, A^{\pi_{\theta_t}}(s, a)\}$

$$\hat{\ell}(\theta) = \sum_{s,a} \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s, a)$$

Trick 1: clipping to make sure π_θ stay close to π_{θ_t} (ensuring stability in training)

$$\hat{\ell}_{clip}(\theta) = \sum_{s,a} \text{clip}\left(\frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)}, 1-\epsilon, 1+\epsilon\right) \cdot A^{\pi_{\theta_t}}(s, a)$$

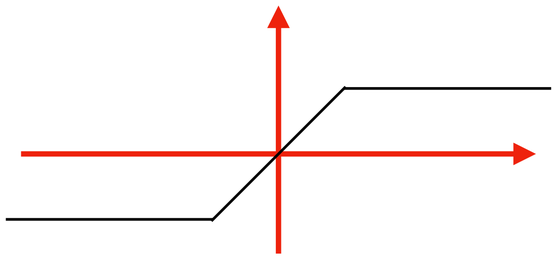
Proximal Policy Optimization (PPO)

Construct a **batch Supervised Learning** style objective using $\mathscr{D} = \{s, a, A^{\pi_{\theta_t}}(s,a)\}$

$$\hat{\ell}(\theta) = \sum_{s,a} \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s,a)$$

Trick 1: clipping to make sure π_θ stay close to π_{θ_t} (ensuring stability in training)

$$\hat{\ell}_{clip}(\theta) = \sum_{s,a} \text{clip}\left(\frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)}, 1-\epsilon, 1+\epsilon \right) \cdot A^{\pi_{\theta_t}}(s,a)$$



$\text{clip}(x, 1-\epsilon, 1+\epsilon)$

Proximal Policy Optimization (PPO)

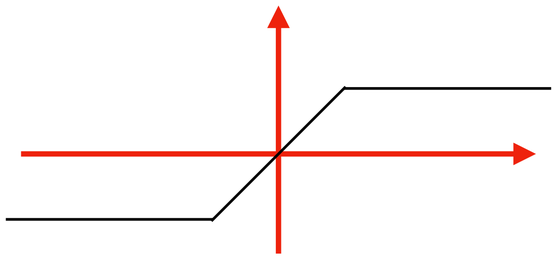
Construct a **batch Supervised Learning** style objective using $\mathscr{D} = \{s, a, A^{\pi_{\theta_t}}(s, a)\}$

$$\hat{\ell}(\theta) = \sum_{s,a} \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s, a)$$

Trick 1: clipping to make sure π_θ stay close to π_{θ_t} (ensuring stability in training)

$$\hat{\ell}_{clip}(\theta) = \sum_{s,a} \text{clip}\left(\frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)}, 1-\epsilon, 1+\epsilon\right) \cdot A^{\pi_{\theta_t}}(s, a)$$

Stop updating $\pi_\theta(a \mid s)$ if it is too different from $\pi_{\theta_t}(a \mid s)$



$\text{clip}(x, 1-\epsilon, 1+\epsilon)$

Proximal Policy Optimization (PPO)

Trick 2, take the min of the clipped and uncipped (original) obj

$$\hat{\ell}_{final}(\theta) = \sum_{s,a} \min \left\{ \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s,a), \quad \text{clip}\left(\frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)}, 1-\epsilon, 1+\epsilon \right) \cdot A^{\pi_{\theta_t}}(s,a) \right\}$$

Original obj

clipped obj which ensures no abrupt change in action probabilities

Proximal Policy Optimization (PPO)

Trick 2, take the min of the clipped and uncipped (original) obj

$$\hat{\ell}_{final}(\theta) = \sum_{s,a} \min \left\{ \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s,a), \quad \text{clip}\left(\frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)}, 1-\epsilon, 1+\epsilon \right) \cdot A^{\pi_{\theta_t}}(s,a) \right\}$$

Just consider one term inside the summation:

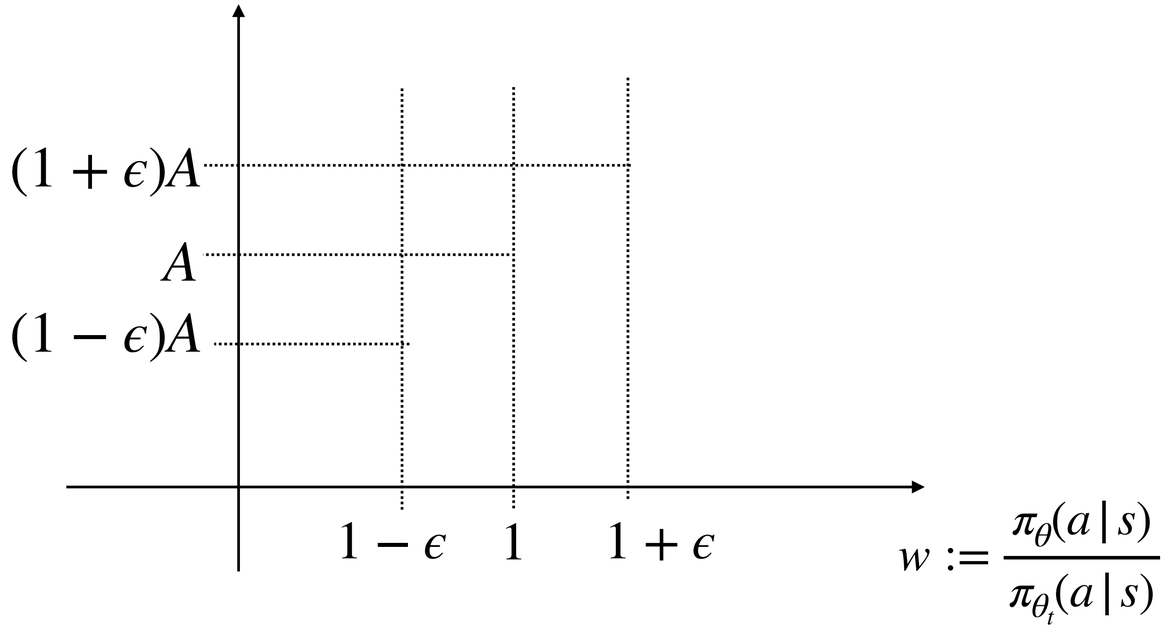
Proximal Policy Optimization (PPO)

Trick 2, take the min of the clipped and uncipped (original) obj

$$\hat{\ell}_{final}(\theta) = \sum_{s,a} \min \left\{ \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s,a), \quad \text{clip}\left(\frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)}, 1-\epsilon, 1+\epsilon \right) \cdot A^{\pi_{\theta_t}}(s,a) \right\}$$

Just consider one term inside the summation:

When $A^{\pi_{\theta_t}}(s,a) > 0$



Proximal Policy Optimization (PPO)

Trick 2, take the min of the clipped and uncipped (original) obj

$$\hat{\ell}_{final}(\theta) = \sum_{s,a} \min \left\{ \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s,a), \quad \text{clip}\left(\frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)}, 1-\epsilon, 1+\epsilon \right) \cdot A^{\pi_{\theta_t}}(s,a) \right\}$$

Just consider one term inside the summation:

Proximal Policy Optimization (PPO)

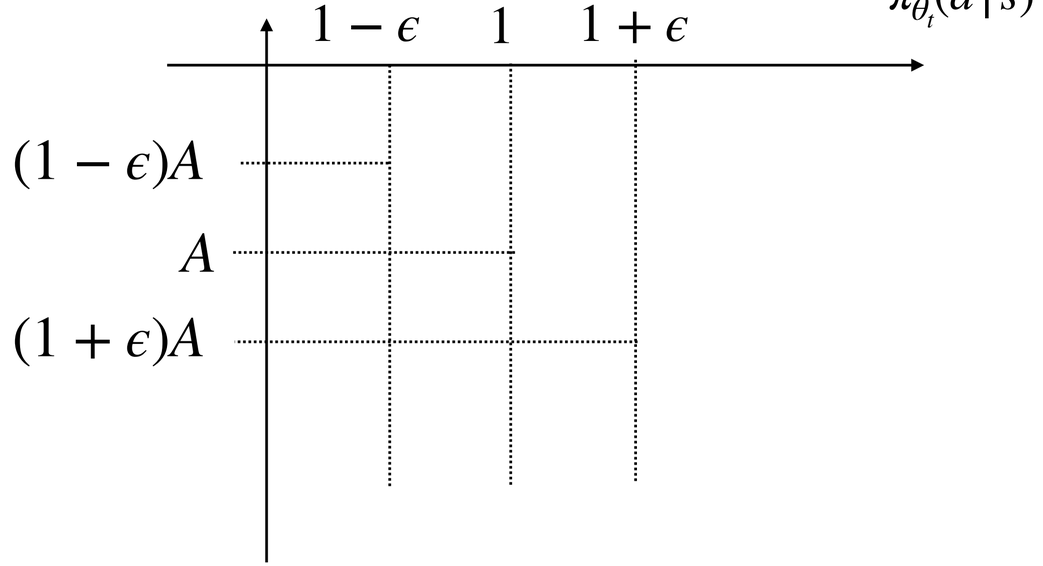
Trick 2, take the min of the clipped and uncipped (original) obj

$$\hat{\ell}_{final}(\theta) = \sum_{s,a} \min \left\{ \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s,a), \quad \text{clip}\left(\frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)}, 1-\epsilon, 1+\epsilon \right) \cdot A^{\pi_{\theta_t}}(s,a) \right\}$$

Just consider one term inside the summation:

When $A^{\pi_{\theta_t}}(s,a) < 0$

$$w := \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)}$$



Proximal Policy Optimization (PPO)

Trick 2, take the min of the clipped and uncipped (original) obj

$$\hat{\ell}_{final}(\theta) = \sum_{s,a} \min \left\{ \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s,a), \quad \text{clip}\left(\frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)}, 1-\epsilon, 1+\epsilon \right) \cdot A^{\pi_{\theta_t}}(s,a) \right\}$$

Original obj

clipped obj which ensures no abrupt change in action probabilities

Proximal Policy Optimization (PPO)

Trick 2, take the min of the clipped and uncipped (original) obj

$$\hat{\ell}_{final}(\theta) = \sum_{s,a} \min \left\{ \frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \cdot A^{\pi_{\theta_t}}(s,a), \quad \text{clip}\left(\frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)}, 1-\epsilon, 1+\epsilon \right) \cdot A^{\pi_{\theta_t}}(s,a) \right\}$$

Original obj

clipped obj which ensures no abrupt change in action probabilities

We compute $\theta_{t+1} \approx \arg\max_\theta \hat{\ell}_{final}(\theta)$, via performing a few epoches of minbatch SG ascent (or Adam/Adagrad) on $\hat{\ell}_{final}$

Proximal Policy Optimization (PPO)

Initialize θ_0 for the policy

For $t = 0 \rightarrow T$:

Run π_{θ_t} to collect multiple trajectories, and form the dataset $\{s, a, A^{\pi_{\theta_t}}(s, a)\}$

Proximal Policy Optimization (PPO)

Initialize θ_0 for the policy

For $t = 0 \rightarrow T$:

Run π_{θ_t} to collect multiple trajectories, and form the dataset $\{s, a, A^{\pi_{\theta_t}}(s, a)\}$

Construct the loss $\hat{\ell}_{final}(\theta)$ using the dataset

Proximal Policy Optimization (PPO)

Initialize θ_0 for the policy

For $t = 0 \rightarrow T$:

Run π_{θ_t} to collect multiple trajectories, and form the dataset $\{s, a, A^{\pi_{\theta_t}}(s, a)\}$

Construct the loss $\hat{\ell}_{final}(\theta)$ using the dataset

Perform a few steps of mini-batch gradient updates on $\hat{\ell}_{final}(\theta)$ to get θ_{t+1}

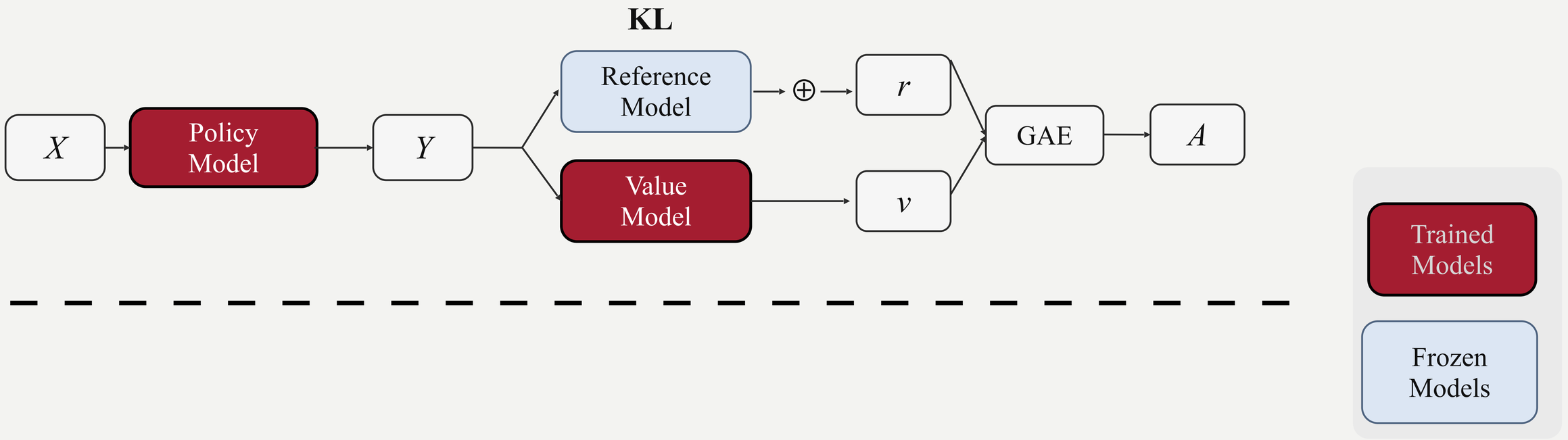
Outline for Today:

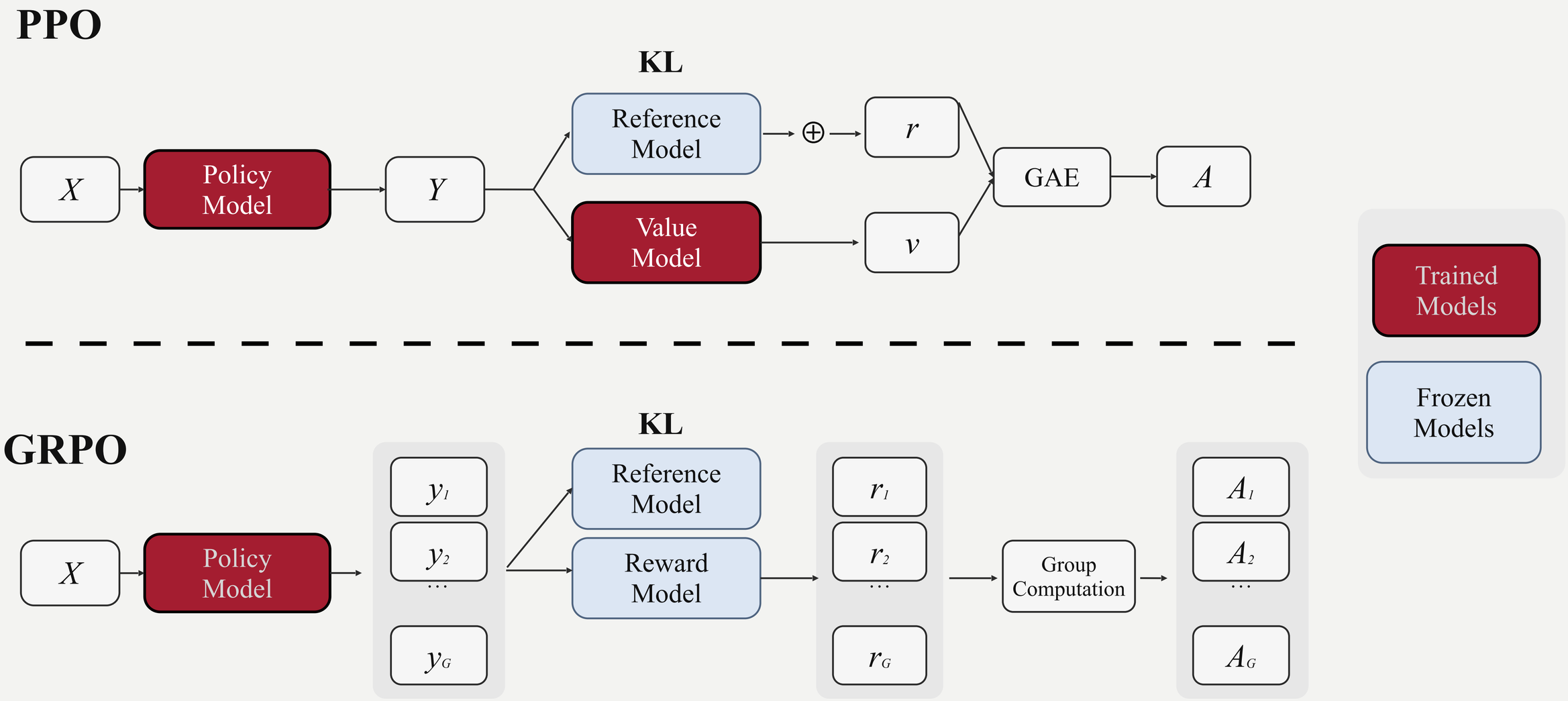
1. Trust Region Policy Optimization

2. Proximal Policy Optimization (PPO)

3. Group Relative Policy Optimization

PPO


KL
X
Policy Model
Y
Reference Model
Value Model
r
v
GAE
A
Trained Models
Frozen Models


PPO
KL
X
Policy Model
Y
Reference Model
Value Model
r
v
GAE
A
Trained Models
Frozen Models
GRPO
KL
X
Policy Model
y_1
y_2
y_G
Reference Model
Reward Model
r_1
r_2
r_G
Group Computation
A_1
A_2
A_G

Group Relative Policy Optimization

For each question q, **GRPO samples** a group of outputs $\{o_1, o_2, \cdots, o_G\}$ from the old policy $\pi_{\theta_{\text{old}}}$ and then optimizes the policy model by maximizing the following objective

$$\mathbb{E}_{q\sim P(Q),\,\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(O|q)}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left\{\min\left[\frac{\pi_\theta(o_{i,t}\mid q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}\mid q,o_{i,<t})}\hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\theta(o_{i,t}\mid q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}\mid q,o_{i,<t})},1-\epsilon,1+\epsilon\right)\hat{A}_{i,t}\right]\right\}\right]$$

Group Relative Policy Optimization

For each question q, **GRPO samples** a group of outputs $\{o_1, o_2, \cdots, o_G\}$ from the old policy $\pi_{\theta_{\text{old}}}$ and then optimizes the policy model by maximizing the following objective

$$\mathbb{E}_{q\sim P(Q),\,\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(O|q)}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left\{\min\left[\frac{\pi_\theta(o_{i,t}\mid q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}\mid q,o_{i,<t})}\hat{A}_{i,t},\ \text{clip}\left(\frac{\pi_\theta(o_{i,t}\mid q,o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}\mid q,o_{i,<t})},1-\epsilon,1+\epsilon\right)\hat{A}_{i,t}\right]\right\}\right]$$

where the advantage $A_{i,t}$ is computed using rewards $\mathbf{r} = \{r_1, r_2, \cdots, r_G\}$:

$$\hat{A}_{i,t} = \tilde{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})},$$

Summary

NPG controls the changes in the policy space (KL) directly

Summary

NPG controls the changes in the policy space (KL) directly

NPG allows one to have big jumps in parameter space, as long as the outcome (distribution) does not change too much

Summary

NPG controls the changes in the policy space (KL) directly

NPG allows one to have big jumps in parameter space, as long as the outcome (distribution) does not change too much

PPO is a more practical versions of NPG — making NPG really scalable while maintaing the high level idea of NPG

Summary

NPG controls the changes in the policy space (KL) directly

NPG allows one to have big jumps in parameter space, as long as the outcome (distribution) does not change too much

PPO is a more practical versions of NPG — making NPG really scalable while maintaing the high level idea of NPG

GRPO is a more practical versions of PPO — removing the value function