Policy Gradient

CMPT 729 G100
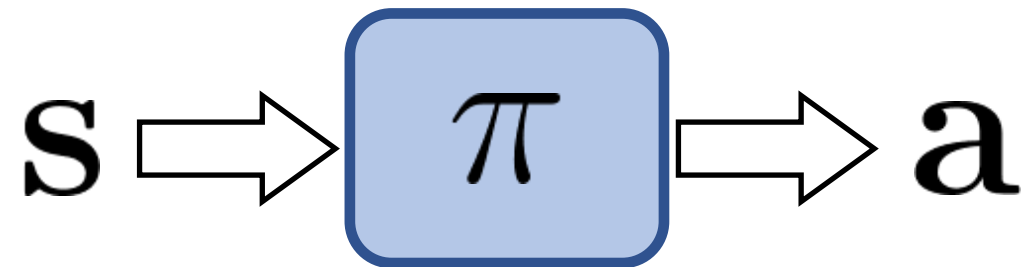
Jason Peng

Overview

- Taxonomy of RL Algorithms

- Policy Gradient

- Derivation

- Variance Reduction

- Applications
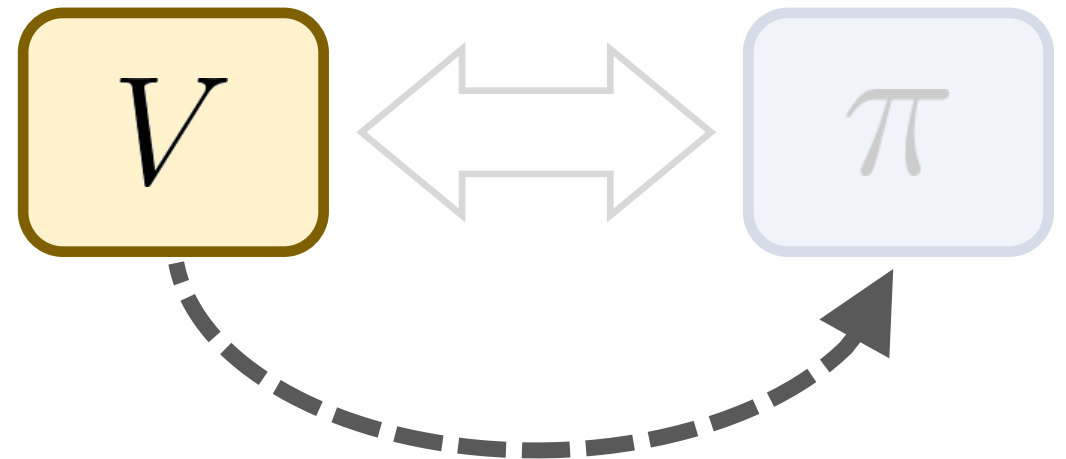
- General View of PG

Taxonomy of RL Algorithms

- **Policy-Based Methods**

- Value-Based Methods

- Actor-Critic Methods

- Model-Based Methods

$$\mathbf{s} \Rightarrow \boxed{\pi} \Rightarrow \mathbf{a}$$
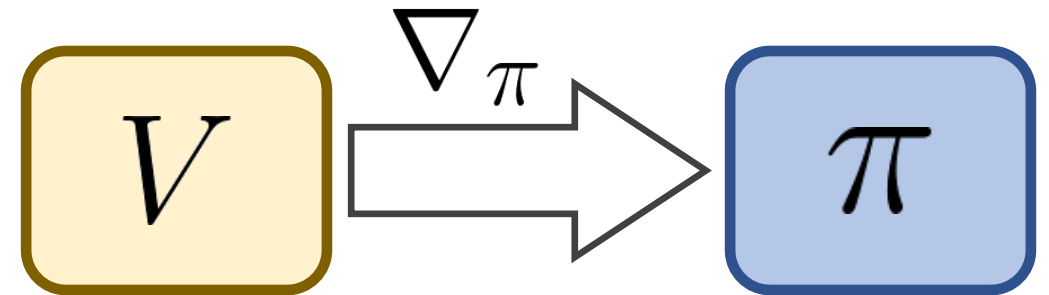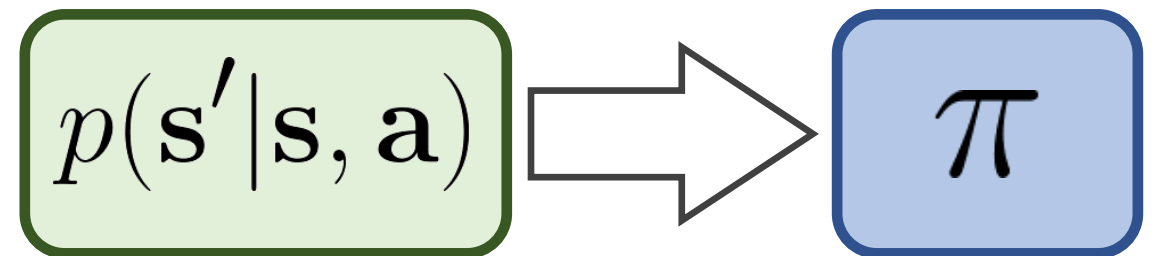
Taxonomy of RL Algorithms

- Policy-Based Methods

- **Value-Based Methods**

- Actor-Critic Methods

- Model-Based Methods

Taxonomy of RL Algorithms

- Policy-Based Methods

- Value-Based Methods

- **Actor-Critic Methods**

- Model-Based Methods

Taxonomy of RL Algorithms

- Policy-Based Methods

- Value-Based Methods

- Actor-Critic Methods

- **Model-Based Methods**
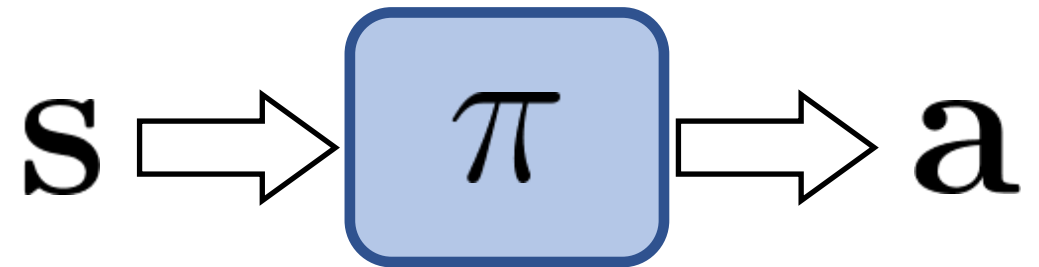
$$p(\mathbf{s}'|\mathbf{s}, \mathbf{a}) \Longrightarrow \pi$$

Taxonomy of RL Algorithms

• **Policy-Based Methods**

• Value-Based Methods

• Actor-Critic Methods

• Model-Based Methods

$$s \Rightarrow \boxed{\pi} \Rightarrow a$$

Nondifferentiable Objective

$$\theta^* = \arg \max_{\theta} J(\pi_\theta)$$

Just use gradient ascent!
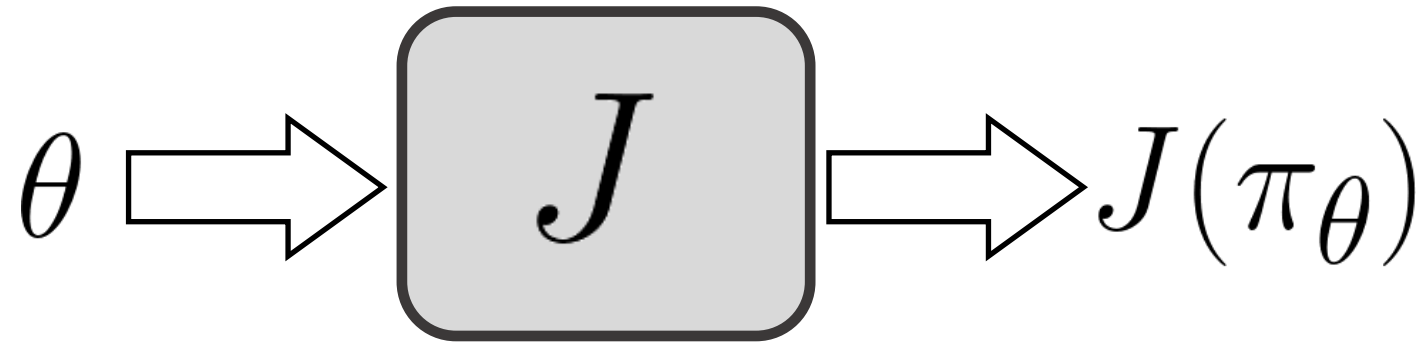
Objective is often
NOT differentiable

$$\nabla_\theta J(\pi_\theta)$$

Black Box Optimization

$$\theta^* = \arg\max_{\theta} \; J(\pi_\theta)$$

black box

$$\theta \Longrightarrow \boxed{J} \Longrightarrow J(\pi_\theta)$$

Black Box Optimization

- Adapt search samples base on objective

search distribution

sample   evaluate

θ^j

J
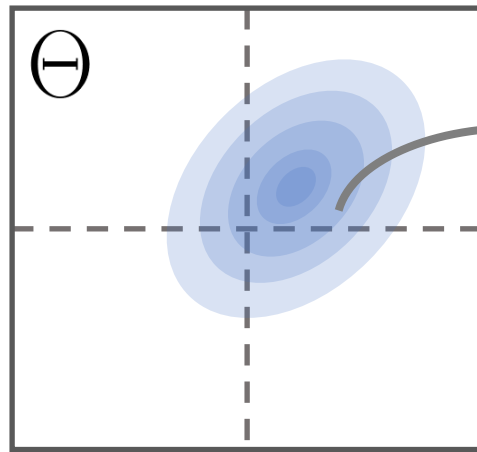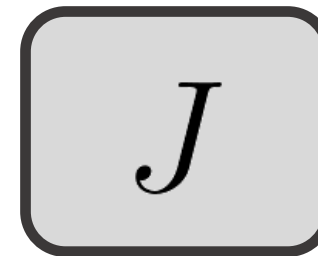
$J(\pi_{\theta^j})$

update

Black Box Optimization

$$\theta^* = \arg \max_{\theta} \; J(\pi_\theta)$$

$$\theta \Longrightarrow \boxed{J} \Longrightarrow J(\pi_\theta)$$
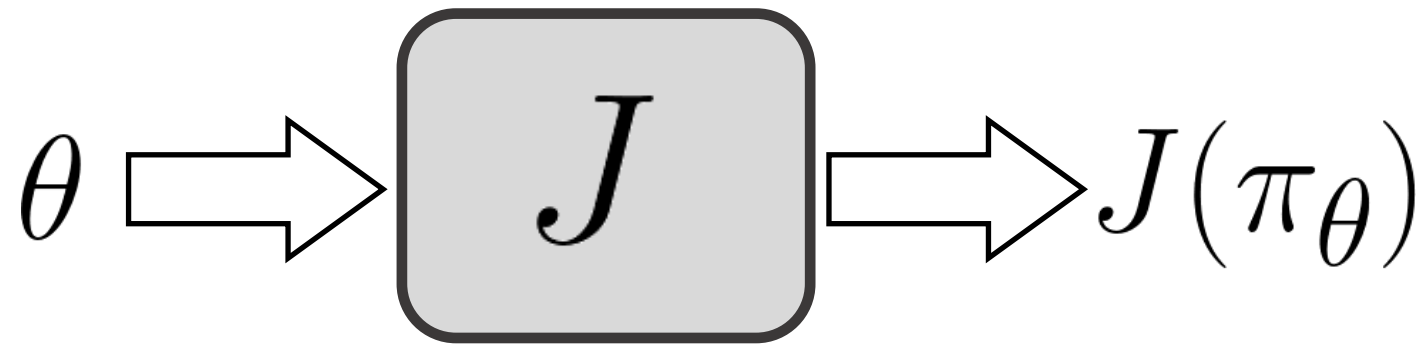
MDP
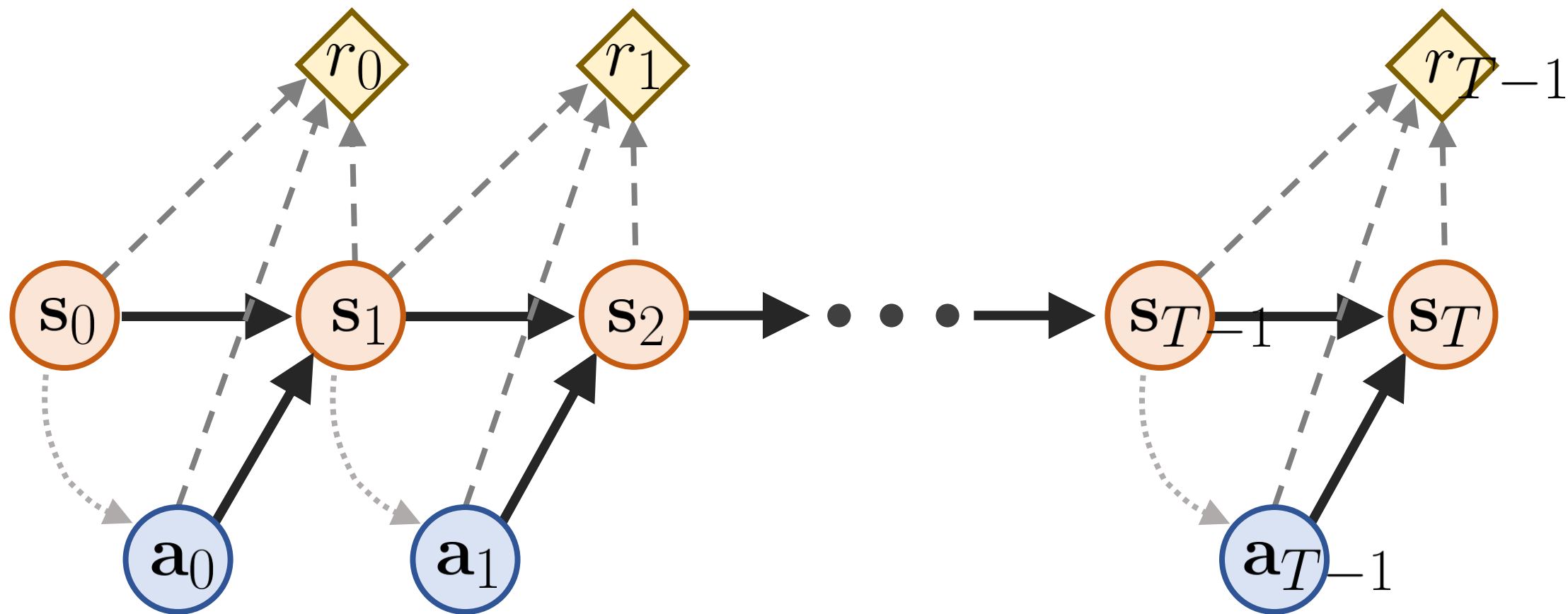
Behavioral Timescales

- Lifetime

Behavioral Timescales

- Lifetime

Evolutionary Methods
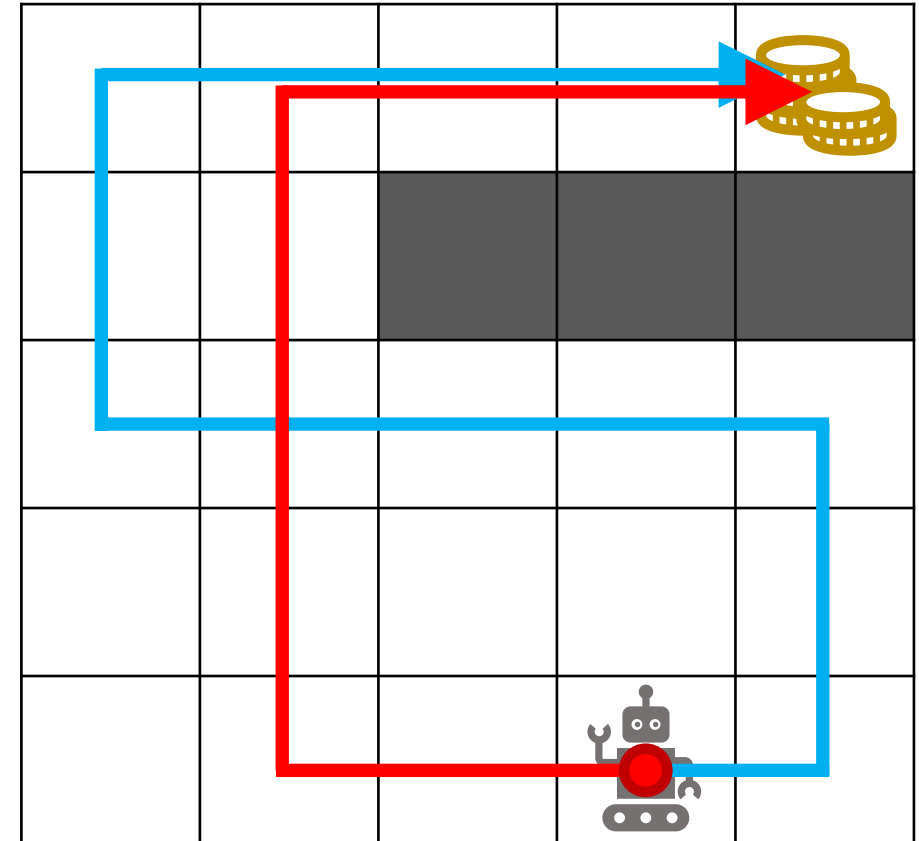
Behavioral Timescales
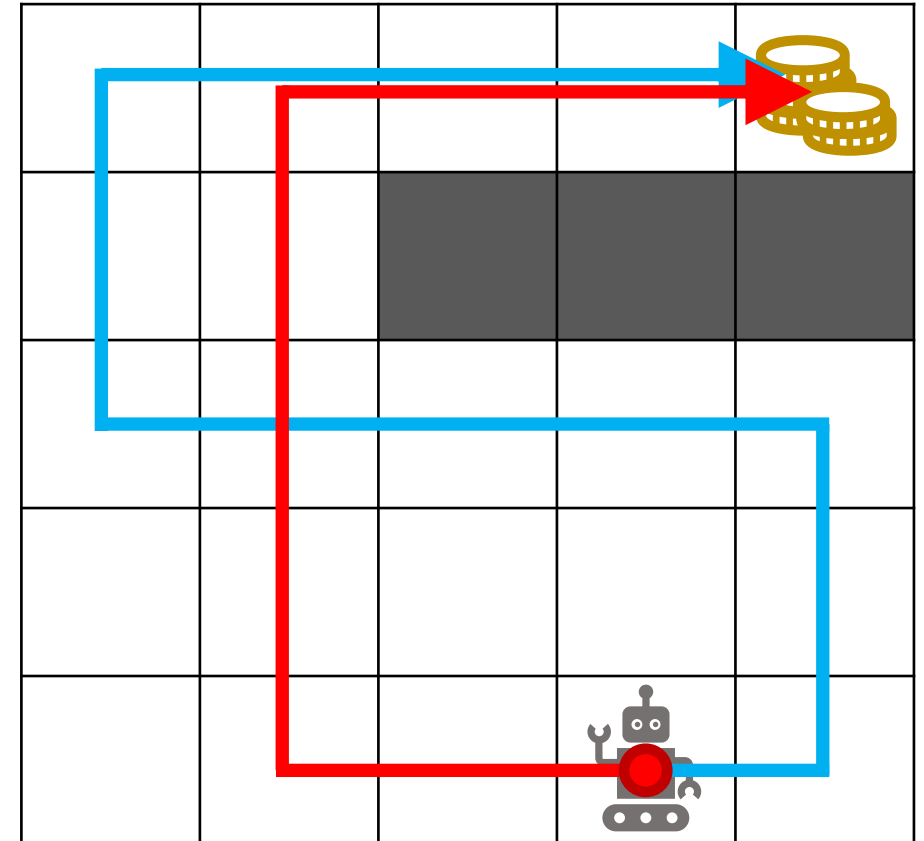
- Lifetime
- Trajectories

Behavioral Timescales

- Lifetime
- Trajectories

Behavioral Timescales
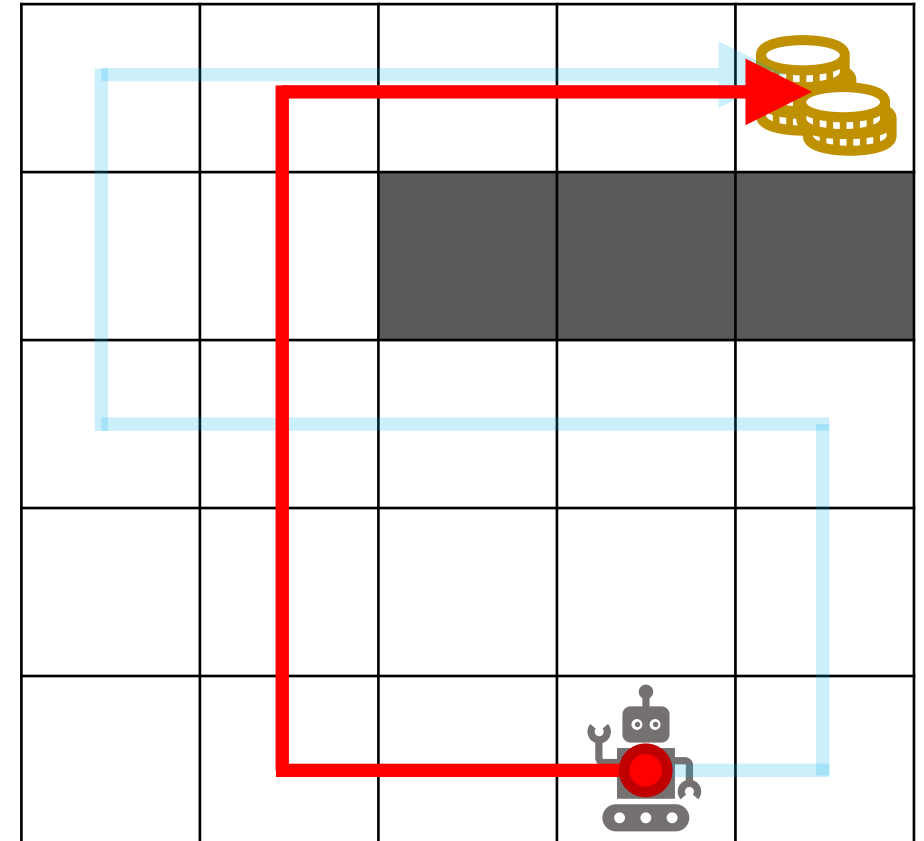
- Lifetime
- Trajectories
- Actions

Behavioral Timescales
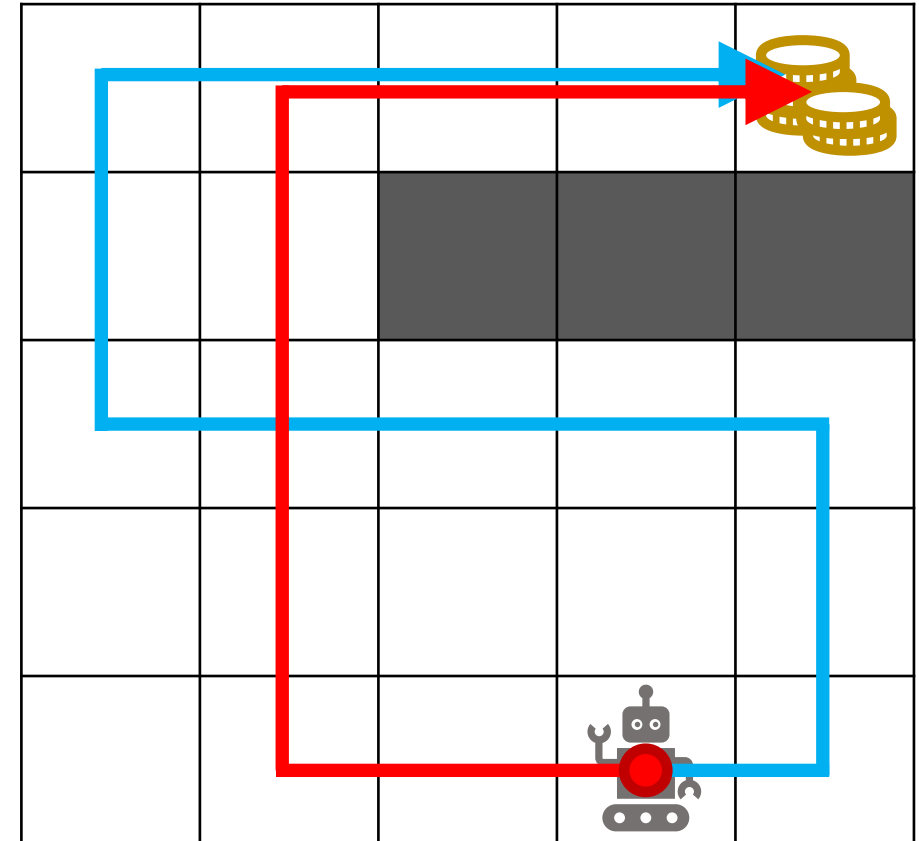
- Lifetime
- Trajectories
- Actions
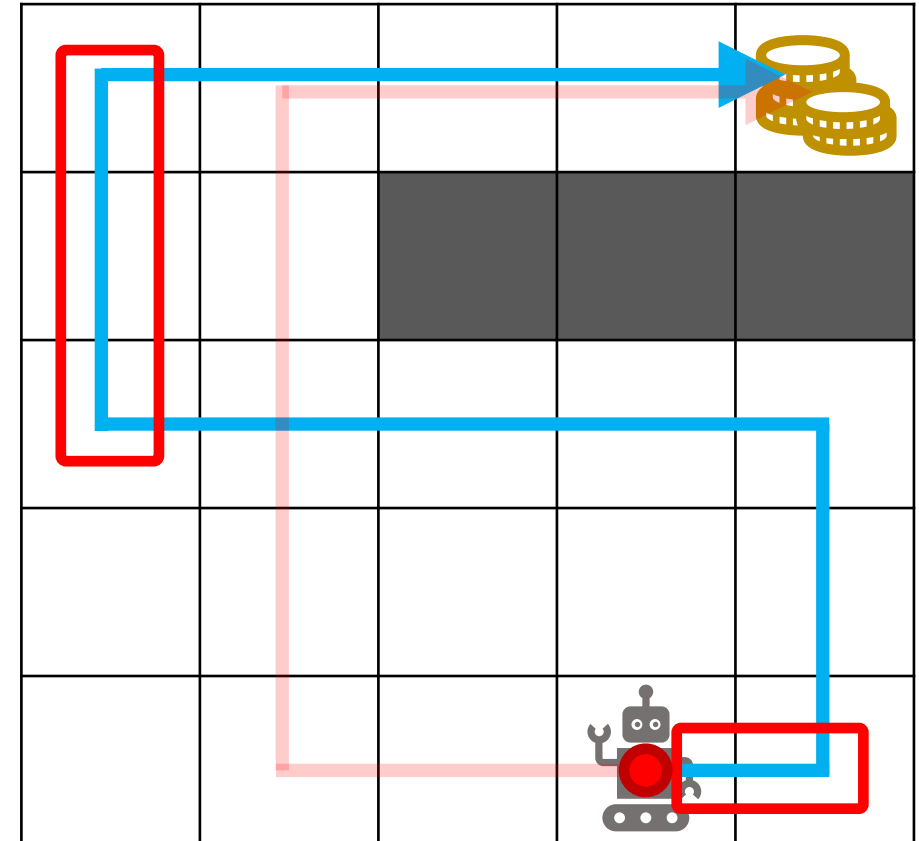
Nondifferentiable Objective

$$\theta^* = \arg \max_\theta \boxed{J(\pi_\theta)}$$

nondifferentiable
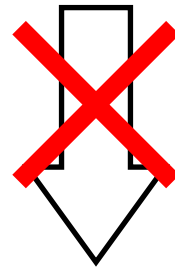
$$\nabla_\theta J(\pi_\theta)$$

Can we approximate?

Finite-Differences

- Approximate gradient using finite-differences

$$\frac{\partial J}{\partial \theta_j} \approx \frac{J(\pi_{\theta+\epsilon_j}) - J(\pi_{\theta-\epsilon_j})}{2\epsilon}$$
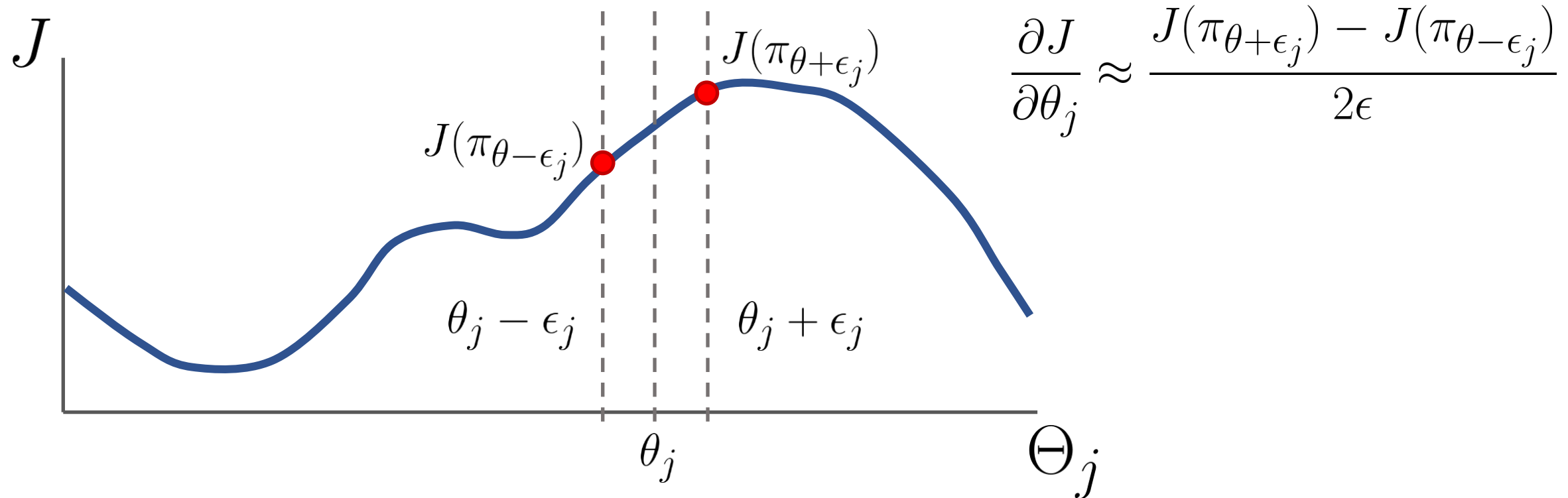
MDP
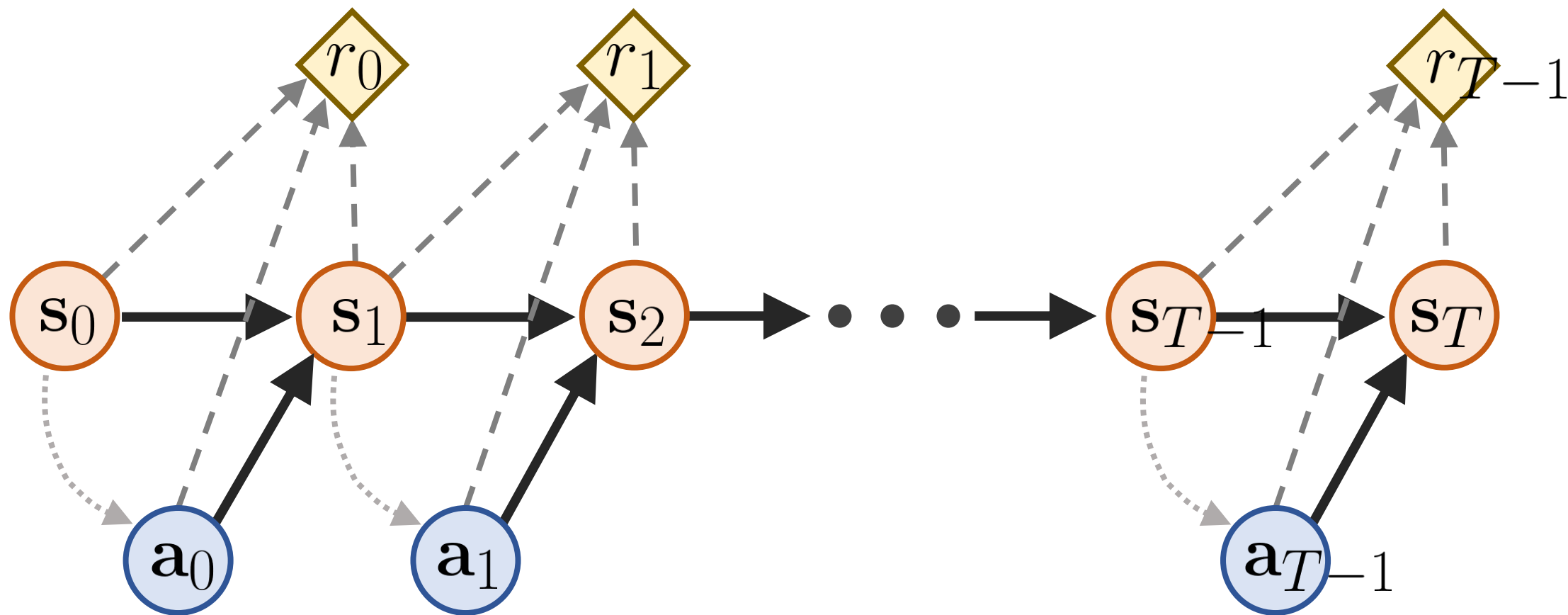
Notation

$$\nabla_\theta J(\pi_\theta)$$

$$\Downarrow$$
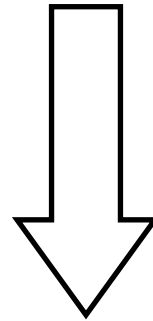
$$\underline{\nabla_\pi} J(\pi)$$

$$\theta$$
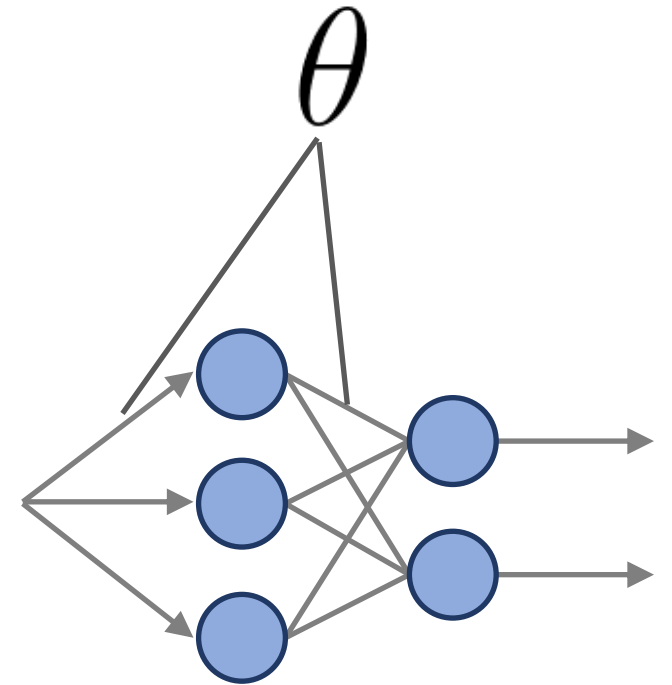
Policy Gradients

$$J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\sum_{t=0}^{T-1} \gamma^t r_t \right] = \mathbb{E}_{\tau \sim p(\tau|\pi)} \underline{[R(\tau)]}$$

return of a trajectory

Policy Gradients

$$J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\sum_{t=0}^{T-1} \gamma^t r_t \right] = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \right]$$

$$= \sum_\tau p(\tau|\pi) R(\tau)$$

Policy Gradients

$$J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\sum_{t=0}^{T-1} \gamma^t r_t \right] = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \right]$$

$$= \sum_{\tau} p(\tau|\pi) \underline{R(\tau)}$$

Policy Gradients

$$J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\sum_{t=0}^{T-1} \gamma^t r_t \right] = \mathbb{E}_{\tau \sim p(\tau|\pi)} [R(\tau)]$$

$$= \sum_{\tau} p(\tau|\pi) R(\tau)$$

$$\nabla_\pi J(\pi) = \sum_{\tau} \underline{\nabla_\pi p(\tau|\pi) R(\tau)}$$

completely intractable

Policy Gradients

$$\nabla_{\pi} J(\pi) = \sum_{\tau} \nabla_{\pi} p(\tau | \pi) R(\tau)$$

Score Function

$$\nabla_p p(x) = p(x) \frac{\nabla_p p(x)}{p(x)} = p(x) \nabla_p \log p(x)$$

Policy Gradients

$$\nabla_\pi J(\pi) = \sum_\tau \nabla_\pi p(\tau|\pi) R(\tau)$$

$$= \sum_\tau p(\tau|\pi) \nabla_\pi \log p(\tau|\pi) R(\tau)$$

Score Function

$$\nabla_p p(x) = p(x) \frac{\nabla_p p(x)}{p(x)} = p(x) \nabla_p \log p(x)$$

Policy Gradients

$$\nabla_\pi J(\pi) = \sum_\tau \nabla_\pi p(\tau|\pi) R(\tau)$$

$$= \sum_\tau p(\tau|\pi) \nabla_\pi \mathrm{log} p(\tau|\pi) R(\tau)$$

Score Function

$$\nabla_p p(x) = p(x) \frac{\nabla_p p(x)}{p(x)} = p(x) \nabla_p \mathrm{log} p(x)$$

Policy Gradients

$$\nabla_\pi J(\pi) = \sum_\tau \nabla_\pi p(\tau|\pi) R(\tau)$$

$$= \sum_\tau p(\tau|\pi) \nabla_\pi \log p(\tau|\pi) R(\tau)$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\nabla_\pi \log p(\tau|\pi) R(\tau) \right]$$

> **Score Function**
>
> $$\nabla_p p(x) = p(x) \frac{\nabla_p p(x)}{p(x)} = p(x) \nabla_p \log p(x)$$

Policy Gradients

$$\nabla_\pi J(\pi) = \sum_\tau \nabla_\pi p(\tau|\pi) R(\tau)$$

$$= \sum_\tau p(\tau|\pi) \nabla_\pi \log p(\tau|\pi) R(\tau)$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\underline{\nabla_\pi \log p(\tau|\pi)} R(\tau) \right]$$

Score Function

$$\nabla_p p(x) = p(x) \frac{\nabla_p p(x)}{p(x)} = p(x) \nabla_p \log p(x)$$

Policy Gradients

$$\nabla_\pi \log p(\tau|\pi) = \nabla_\pi \log \left(p(\mathbf{s}_0) \prod_{t=0}^{T-1} \pi(\mathbf{a}_t|\mathbf{s}_t) p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) \right)$$

$$= p(\tau|\pi)$$

Policy Gradients

$$\nabla_\pi \log p(\tau|\pi) = \nabla_\pi \log \left(p(\mathbf{s}_0) \prod_{t=0}^{T-1} \pi(\mathbf{a}_t|\mathbf{s}_t) p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) \right)$$

Policy Gradients

$$\nabla_\pi \log p(\tau|\pi) = \nabla_\pi \log \left(p(\mathbf{s}_0) \prod_{t=0}^{T-1} \pi(\mathbf{a}_t|\mathbf{s}_t) p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) \right)$$

Policy Gradients

$$\nabla_\pi \log p(\tau|\pi) = \nabla_\pi \log \left(p(\mathbf{s}_0) \prod_{t=0}^{T-1} \underline{\pi(\mathbf{a}_t|\mathbf{s}_t)} p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) \right)$$

Policy Gradients

$$\nabla_\pi \log p(\tau|\pi) = \nabla_\pi \log \left(p(\mathbf{s}_0) \prod_{t=0}^{T-1} \pi(\mathbf{a}_t|\mathbf{s}_t) p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) \right)$$

Policy Gradients

$$\nabla_\pi \log p(\tau|\pi) = \nabla_\pi \log \left(p(\mathbf{s}_0) \prod_{t=0}^{T-1} \pi(\mathbf{a}_t|\mathbf{s}_t) p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) \right)$$

$$= \nabla_\pi \left(\log p(\mathbf{s}_0) + \sum_{t=0}^{T-1} \log \pi(\mathbf{a}_t|\mathbf{s}_t) + \log p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) \right)$$

Policy Gradients

$$\nabla_\pi \log p(\tau|\pi) = \nabla_\pi \log \left(p(\mathbf{s}_0) \prod_{t=0}^{T-1} \pi(\mathbf{a}_t|\mathbf{s}_t) p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) \right)$$

$$= \nabla_\pi \left(\underline{\log p(\mathbf{s}_0)} + \sum_{t=0}^{T-1} \log \pi(\mathbf{a}_t|\mathbf{s}_t) + \underline{\log p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)} \right)$$

Independent of π

Policy Gradients

$$\nabla_\pi \log p(\tau|\pi) = \nabla_\pi \log \left(p(\mathbf{s}_0) \prod_{t=0}^{T-1} \pi(\mathbf{a}_t|\mathbf{s}_t) p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) \right)$$

$$= \nabla_\pi \left(\log p(\mathbf{s}_0) + \sum_{t=0}^{T-1} \log \pi(\mathbf{a}_t|\mathbf{s}_t) + \log p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t) \right)$$

$$= \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t)$$

Policy Gradients

$$\nabla_\pi J(\pi) = \sum_\tau \nabla_\pi p(\tau|\pi) R(\tau)$$

$$= \sum_\tau p(\tau|\pi) \nabla_\pi \log p(\tau|\pi) R(\tau)$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\nabla_\pi \log p(\tau|\pi) R(\tau) \right]$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$

Score Function

$$\nabla_\pi \pi(\tau) = \pi(\tau) \frac{\nabla_\pi \pi(\tau)}{\pi(\tau)} = \pi(\tau) \nabla_\pi \log \pi(\tau)$$

policy gradient
AKA. REINFORCE [Williams 1992]

Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning
[Williams 1992]

REINFORCE

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \underline{\nabla_\pi \log \pi(\mathbf{a}_t | \mathbf{s}_t)} \right]$$

REINFORCE

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log\pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$

REINFORCE

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log\pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$

$\mathbf{s}_0$

$R(\tau_0)\nabla_\pi \log\pi(\tau_0)$

$R(\tau_1)\nabla_\pi \log\pi(\tau_1)$

$R(\tau_k)\nabla_\pi \log\pi(\tau_k)$

REINFORCE

ALGORITHM: REINFORCE

1: $\theta \leftarrow$ initialize policy parameters

2: **while** not done **do**
3:     Sample trajectories $\{\tau^i\}$ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
4:     Estimate policy gradient
       $\nabla_\theta J(\pi_\theta) \approx \frac{1}{N} \sum_i R(\tau^i) \sum_t \nabla_\theta \log \pi_\theta(\mathbf{a}_t^i|\mathbf{s}_t^i)$
5:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
6: **end while**

7: return policy π_θ

Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning
[Williams 1992]

REINFORCE

ALGORITHM: REINFORCE

1: $\theta \leftarrow$ initialize policy parameters

2: **while** not done **do**
3:     Sample trajectories $\{\tau^i\}$ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
4:     Estimate policy gradient
        $\nabla_\theta J(\pi_\theta) \approx \frac{1}{N} \sum_i R(\tau^i) \sum_t \nabla_\theta \log \pi_\theta(\mathbf{a}_t^i|\mathbf{s}_t^i)$
5:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
6: **end while**

7: return policy π_θ

Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning
[Williams 1992]

REINFORCE

ALGORITHM: REINFORCE

1: $\theta \leftarrow$ initialize policy parameters

2: **while** not done **do**
3:   Sample trajectories $\{\tau^i\}$ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
4:   Estimate policy gradient
   $$\nabla_\theta J(\pi_\theta) \approx \frac{1}{N} \sum_i R(\tau^i) \sum_t \nabla_\theta \log \pi_\theta(\mathbf{a}_t^i|\mathbf{s}_t^i)$$
5:   Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
6: **end while**

7: return policy π_θ

Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning
[Williams 1992]

REINFORCE

ALGORITHM: REINFORCE

1: $\theta \leftarrow$ initialize policy parameters

2: **while** not done **do**

3:      Sample trajectories $\{\tau^i\}$ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$

4:      Estimate policy gradient

$$\nabla_\theta J(\pi_\theta) \approx \frac{1}{N} \sum_i R(\tau^i) \sum_t \nabla_\theta \log \pi_\theta(\mathbf{a}_t^i | \mathbf{s}_t^i)$$

5:      Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$

6: **end while**

7: return policy π_θ

Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning
[Williams 1992]

REINFORCE

ALGORITHM: REINFORCE

1: $\theta \leftarrow$ initialize policy parameters

2: **while** not done **do**
3:     Sample trajectories $\{\tau^i\}$ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
4:     Estimate policy gradient
    $\nabla_\theta J(\pi_\theta) \approx \frac{1}{N} \sum_i R(\tau^i) \sum_t \nabla_\theta \log \pi_\theta(\mathbf{a}_t^i|\mathbf{s}_t^i)$
5:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
6: **end while**

7: return policy π_θ

Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning [Williams 1992]

REINFORCE

ALGORITHM: REINFORCE

1: $\theta \leftarrow$ initialize policy parameters

2: **while** not done **do**
3:     Sample trajectories $\{\tau^i\}$ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
4:     Estimate policy gradient
        $\nabla_\theta J(\pi_\theta) \approx \frac{1}{N}\sum_i R(\tau^i)\sum_t \nabla_\theta \log\pi_\theta(\mathbf{a}_t^i|\mathbf{s}_t^i)$
5:     Update policy $\theta \leftarrow \theta + \alpha\nabla_\theta J(\pi_\theta)$
6: **end while**

7: return policy π_θ

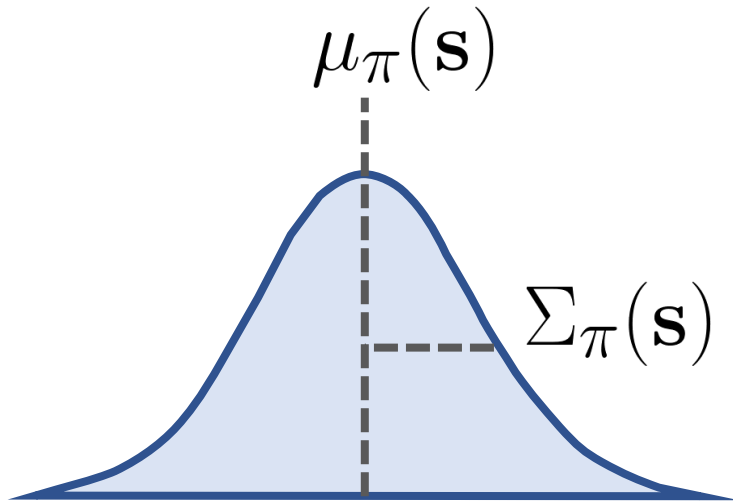Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning
[Williams 1992]

Action Distribution

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t | \mathbf{s}_t) \right]$$

must be differentiable

$\mu_\pi(\mathbf{s})$

$\Sigma_\pi(\mathbf{s})$

Gaussian Distribution
(Continuous Actions)

Categorical Distribution
(Discrete Actions)

Etc...

Problems

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t | \mathbf{s}_t) \right]$$

Problems

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$

Problems

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$

Problems
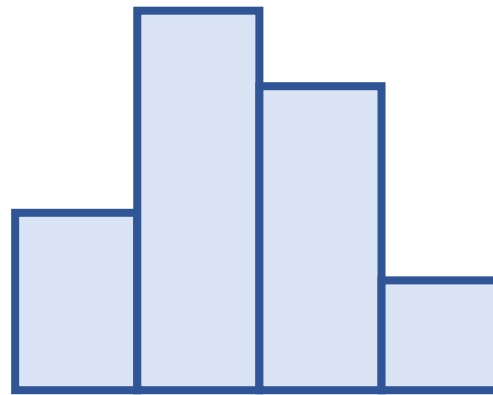
$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t | \mathbf{s}_t) \right]$$

$\pi(\mathbf{a}|\mathbf{s})$

$R(\tau^1) + \delta$

$R(\tau^0) + \delta$

$\mathbf{a}_0$ $\mathbf{a}_1$ $\mathbf{a}$

Problems

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$

$\pi(\mathbf{a}|\mathbf{s})$

$R(\tau^1) + \delta$

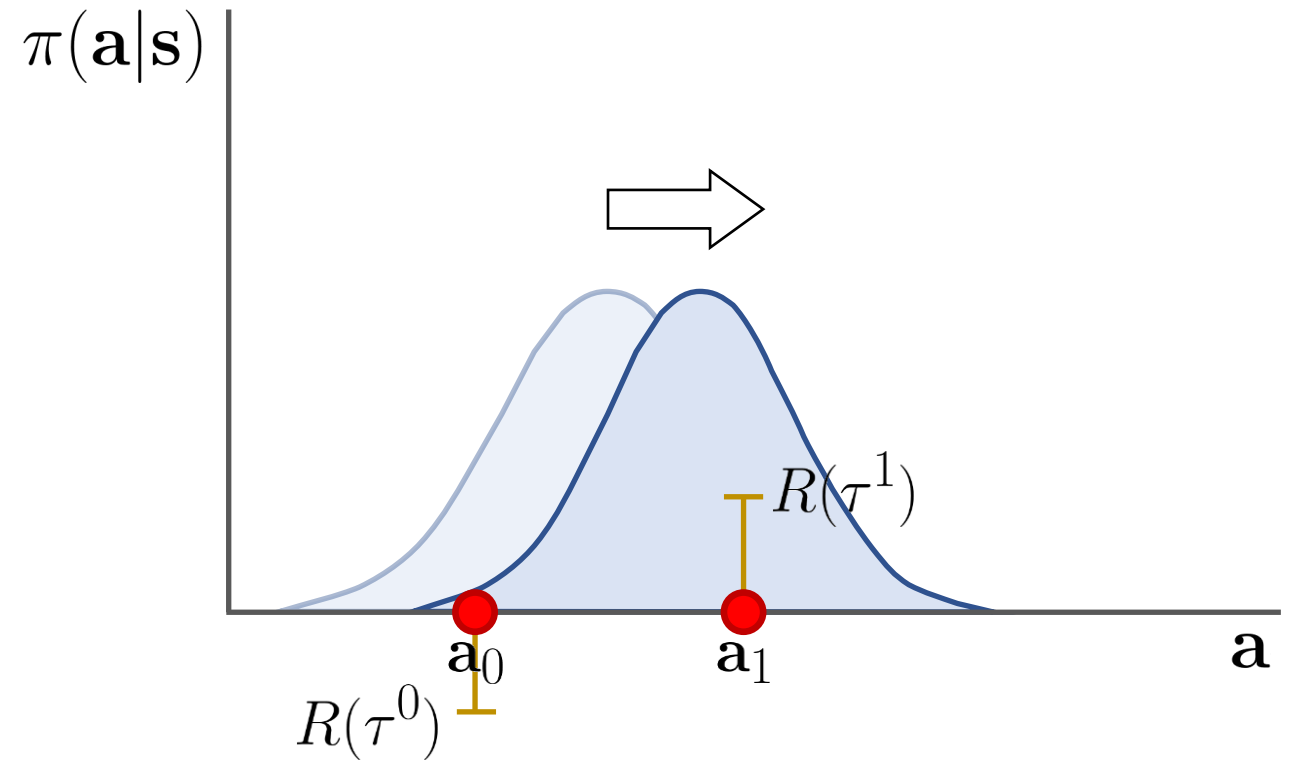$R(\tau^0) + \delta$

$\mathbf{a}_0$

$\mathbf{a}_1$

$\mathbf{a}$

Problems

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t | \mathbf{s}_t) \right]$$

$\pi(\mathbf{a}|\mathbf{s})$

$\mathbf{a}_0$

$\mathbf{a}_1$

$R(\tau^1) + \delta$

$R(\tau^0) + \delta$

$\mathbf{a}$

Problems

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$
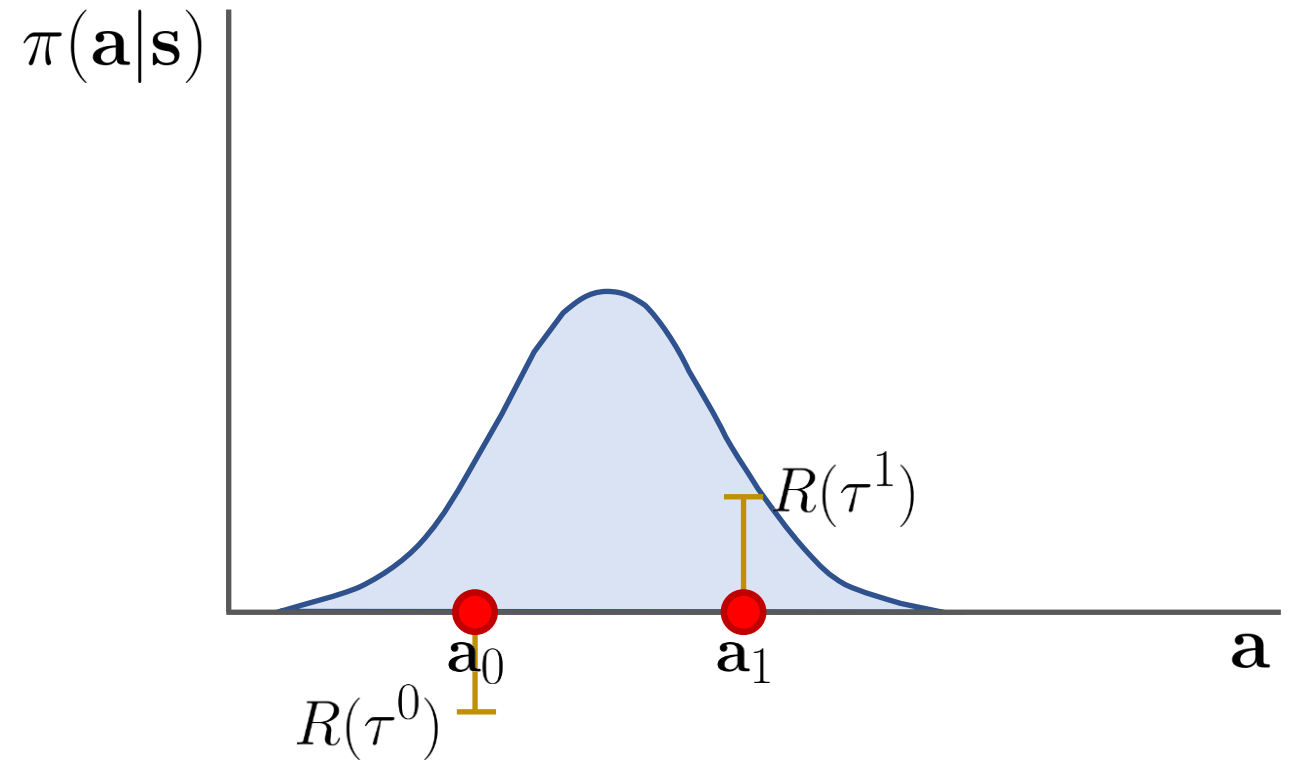
Problems

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$

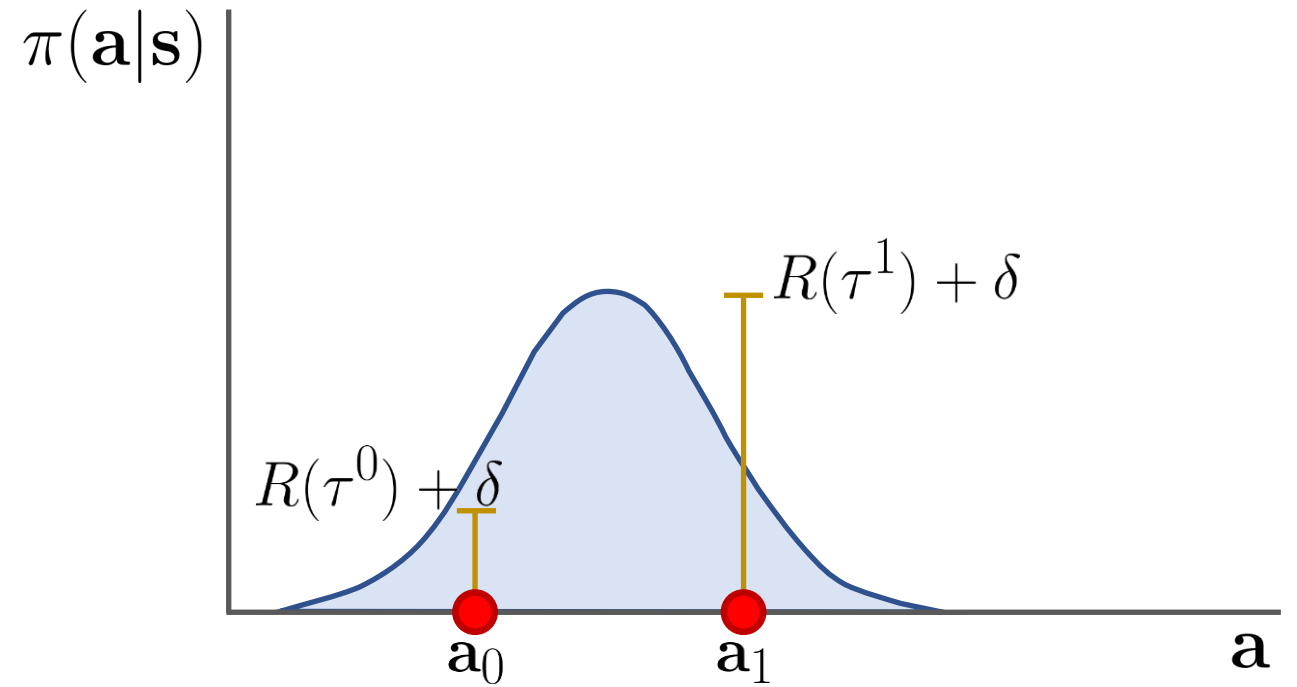**Problem:** Not invariant to reward translations

Reward Translation

- Optimal policy is invariant to reward translation

- Gradient estimator is *not* invariant to reward translation

- Problem: Variance
  - Monte-Carlo estimate with finite samples
  - Goes away in expectation with infinite samples

Variance Reduction

- Baselines
- Causality
- Bootstrapping
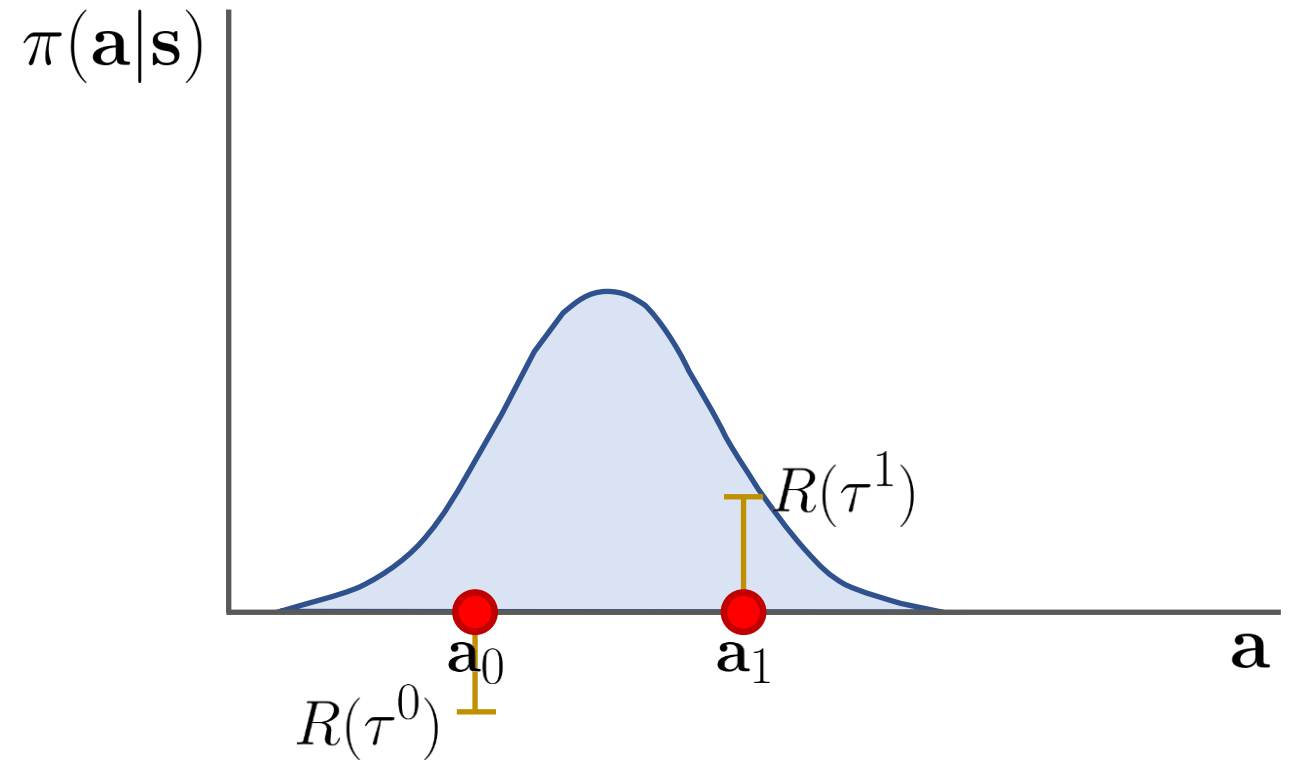
Variance Reduction: Baseline

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t | \mathbf{s}_t) \right]$$
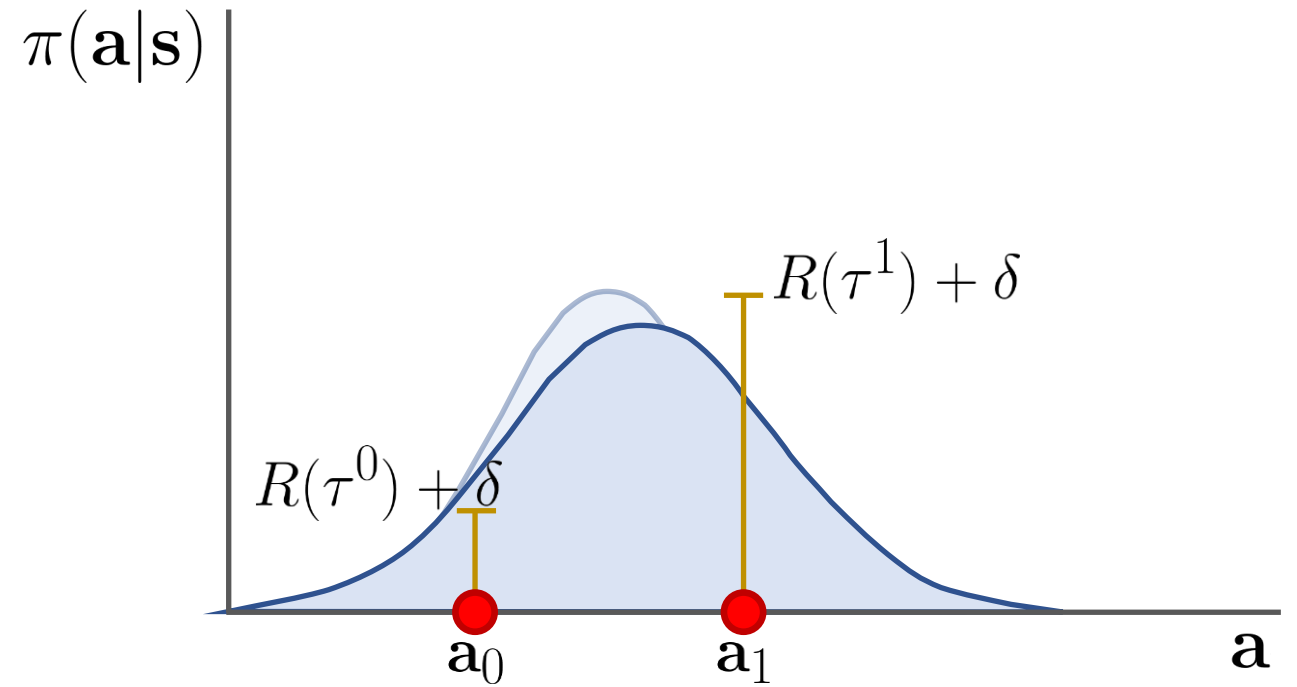
Variance Reduction: Baseline

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$

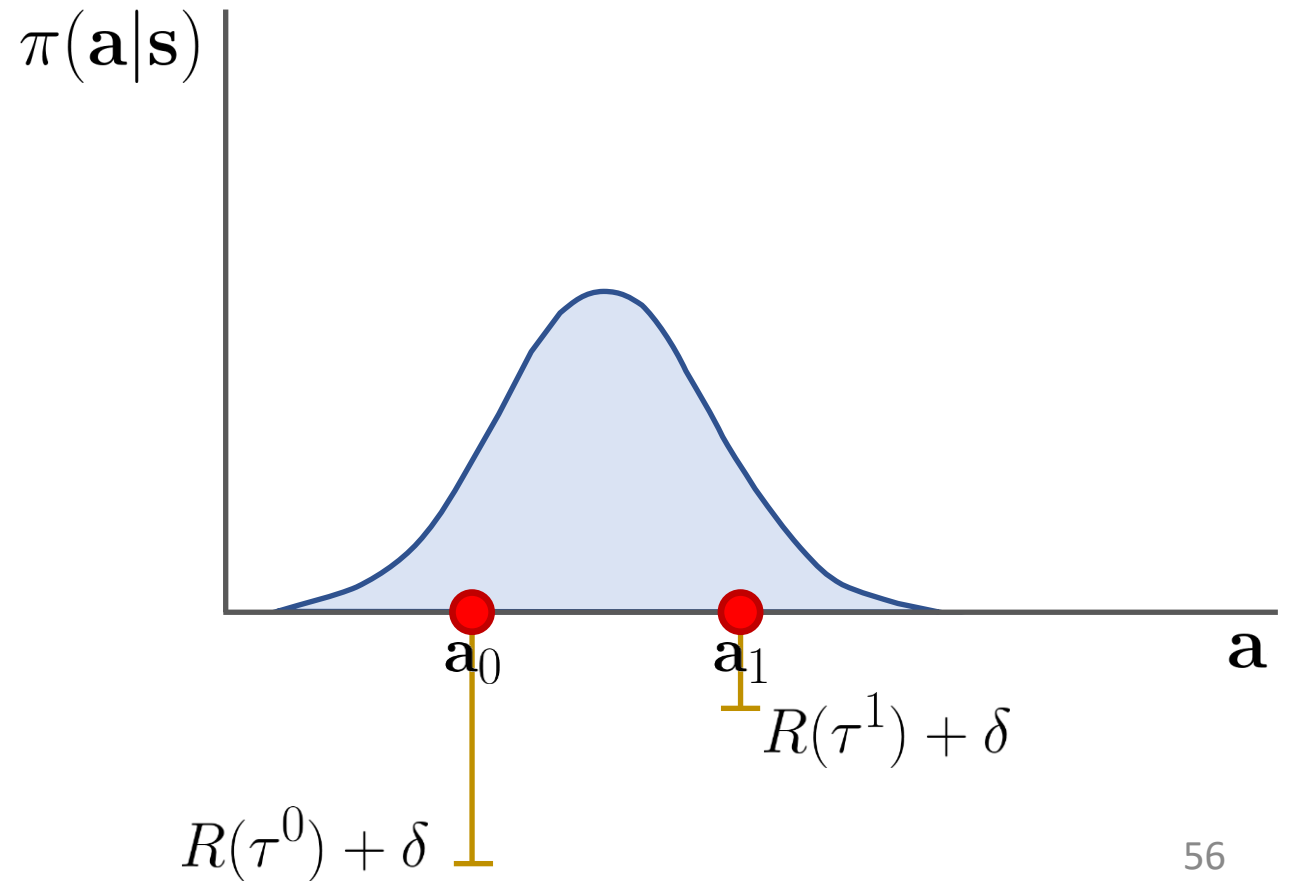$\pi(\mathbf{a}|\mathbf{s})$

$R(\tau^1) + \delta$

$R(\tau^0) + \delta$

$\mathbf{a}_0$ $\mathbf{a}_1$ $\mathbf{a}$

Variance Reduction: Baseline

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[(R(\tau) - b) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$
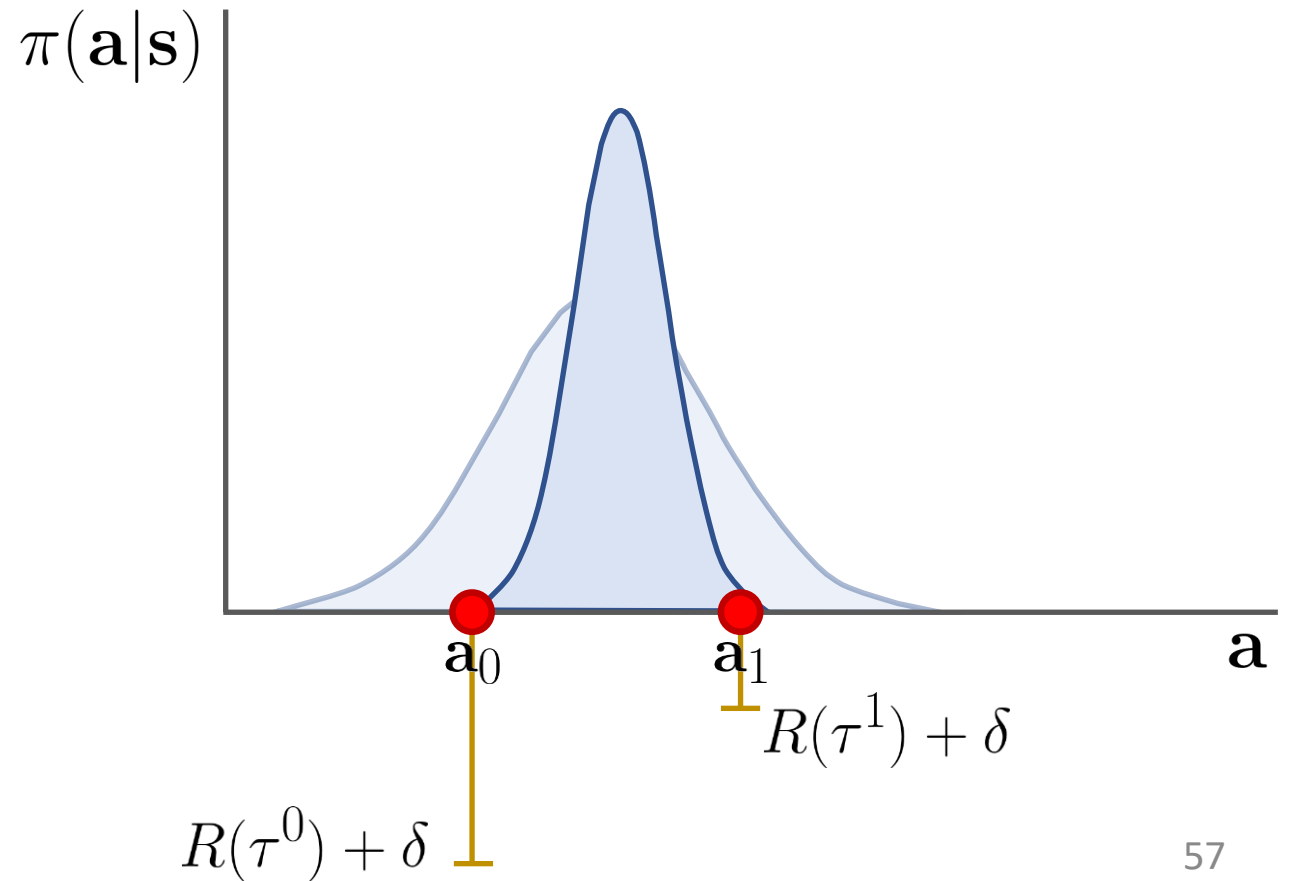
baseline

e.g. $b = \delta$

$\pi(\mathbf{a}|\mathbf{s})$

$R(\tau^1) + \delta$

$R(\tau^0) + \delta$

$\mathbf{a}_0$ $\mathbf{a}_1$

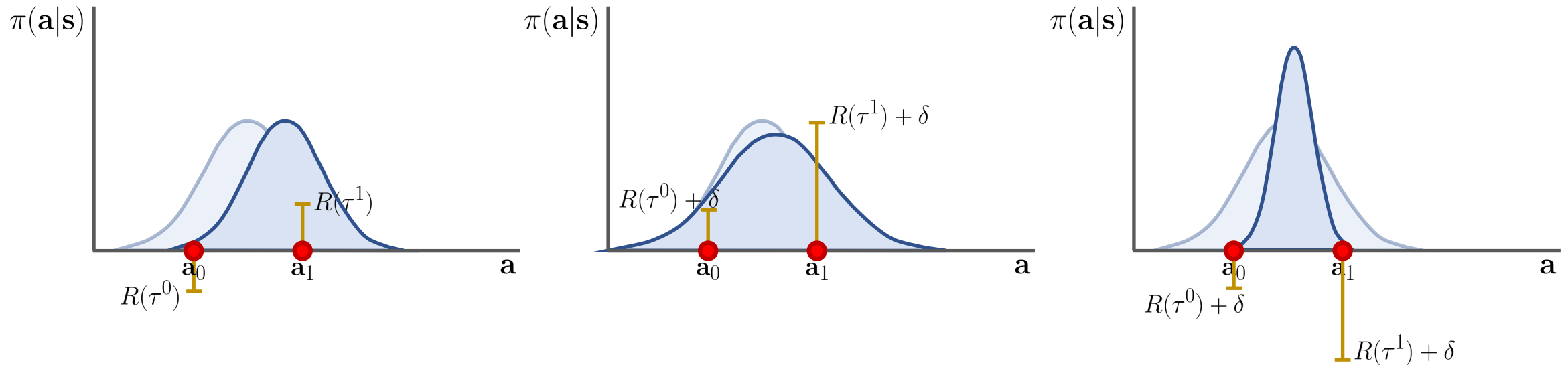$\mathbf{a}$
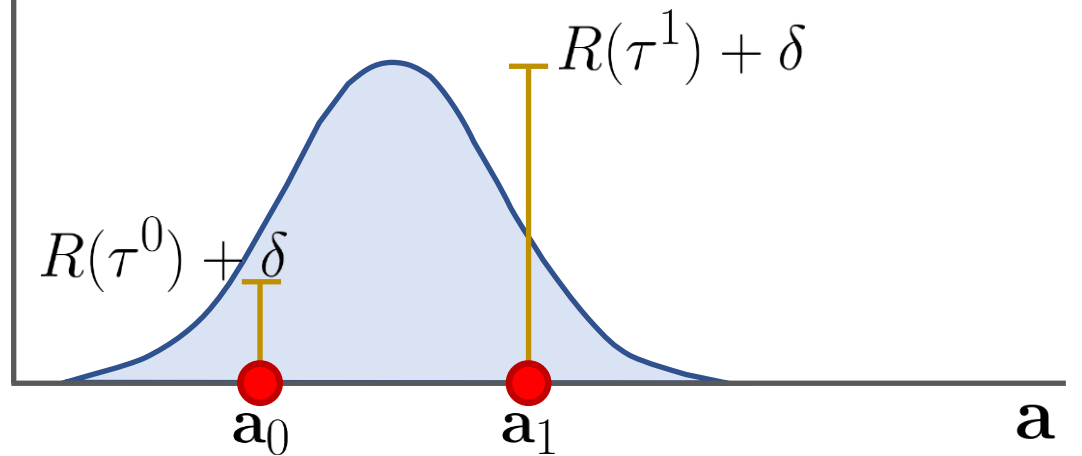
Variance Reduction: Baseline

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[(R(\tau) - b) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$
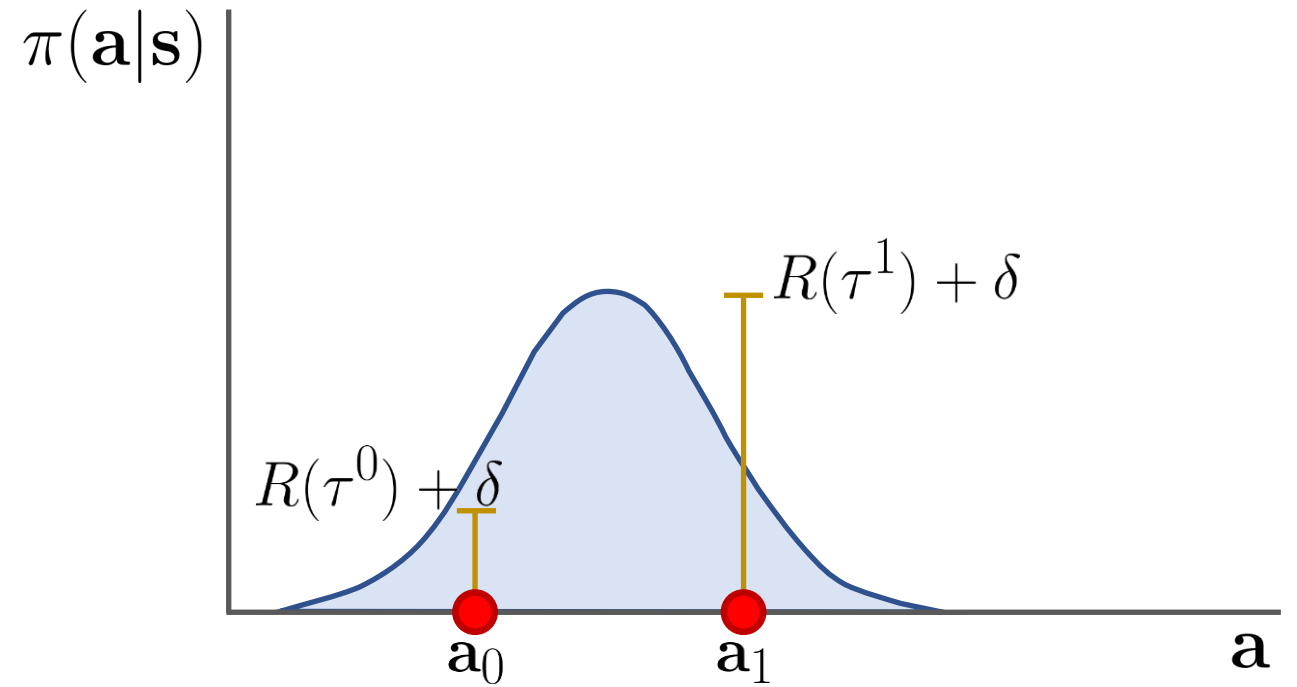
baseline

e.g. $b = \delta$

- Baseline reduces variance
- Is this allowed?
- What is the optimal baseline?
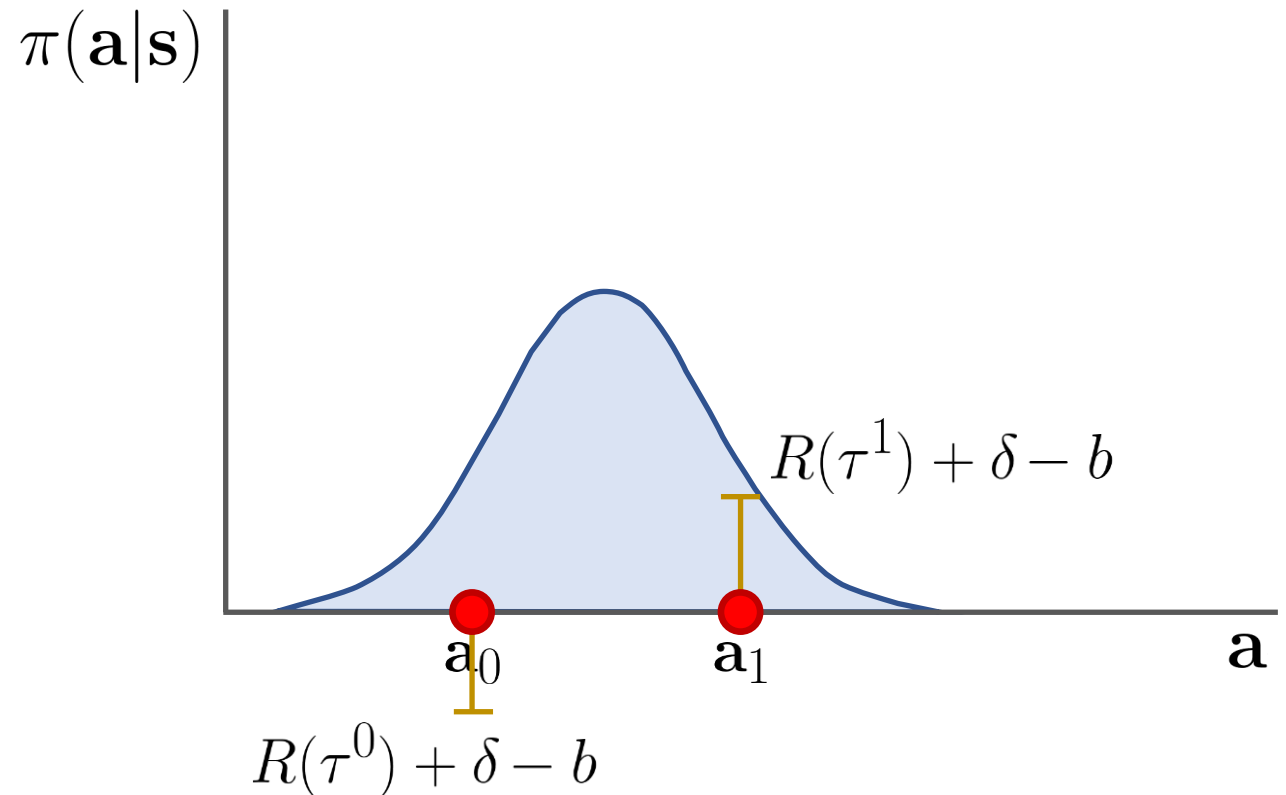
$\pi(\mathbf{a}|\mathbf{s})$

$R(\tau^1) + \delta - b$

$\mathbf{a}_0$

$\mathbf{a}_1$

$\mathbf{a}$

$R(\tau^0) + \delta - b$

Variance Reduction: Baseline

- How does the baseline effect the gradient?

$$R(\tau) \implies \hat{R}(\tau) = R(\tau) - b$$

$$\nabla_\pi \hat{J}(\pi) = \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\hat{R}(\tau) \right] = \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) - b \right]$$

$$= \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \right] - \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[b \right]$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \nabla_\pi \mathrm{log} p(\tau|\pi) \right] - \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[b \right]$$

score function

Variance Reduction: Baseline

- How does the baseline effect the gradient?

$$R(\tau) \Longrightarrow \hat{R}(\tau) = R(\tau) - b$$

$$\nabla_\pi \hat{J}(\pi) = \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\hat{R}(\tau) \right] = \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) - b \right]$$

$$= \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \right] - \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[b \right]$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \nabla_\pi \log p(\tau|\pi) \right] - \nabla_\pi \underline{\mathbb{E}_{\tau \sim p(\tau|\pi)} \left[b \right]}$$

$$= b$$

Variance Reduction: Baseline

- How does the baseline effect the gradient?

$$R(\tau) \Longrightarrow \hat{R}(\tau) = R(\tau) - b$$

$$\nabla_\pi \hat{J}(\pi) = \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\hat{R}(\tau) \right] = \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) - b \right]$$

$$= \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \right] - \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[b \right]$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \nabla_\pi \log p(\tau|\pi) \right] - \underline{\nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[b \right]}$$

$$\nabla_\pi b = 0$$

Variance Reduction: Baseline

- How does the baseline effect the gradient?

$$R(\tau) \Longrightarrow \hat{R}(\tau) = R(\tau) - b$$

$$\nabla_\pi \hat{J}(\pi) = \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\hat{R}(\tau)\right] = \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) - b\right]$$

$$= \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau)\right] - \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[b\right]$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \nabla_\pi \log p(\tau|\pi)\right] - \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[b\right] \quad 0$$

Variance Reduction: Baseline

- How does the baseline effect the gradient?

$$R(\tau) \Longrightarrow \hat{R}(\tau) = R(\tau) - b$$

$$\nabla_\pi \hat{J}(\pi) = \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\hat{R}(\tau) \right] = \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) - b \right]$$

$$= \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \right] - \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[b \right]$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \nabla_\pi \log p(\tau|\pi) \right] - \nabla_\pi \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[b \right]$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \nabla_\pi \log p(\tau|\pi) \right]$$

$$= \nabla_\pi J(\pi)$$

Variance Reduction: Baseline

- How does the baseline effect the gradient?

$$R(\tau) \Longrightarrow \hat{R}(\tau) = R(\tau) - b$$

- Baseline does not change the gradient!

- Reduces variance without introducing bias

- Any constant value for the baseline will preserve policy gradient

Variance Reduction: Baseline
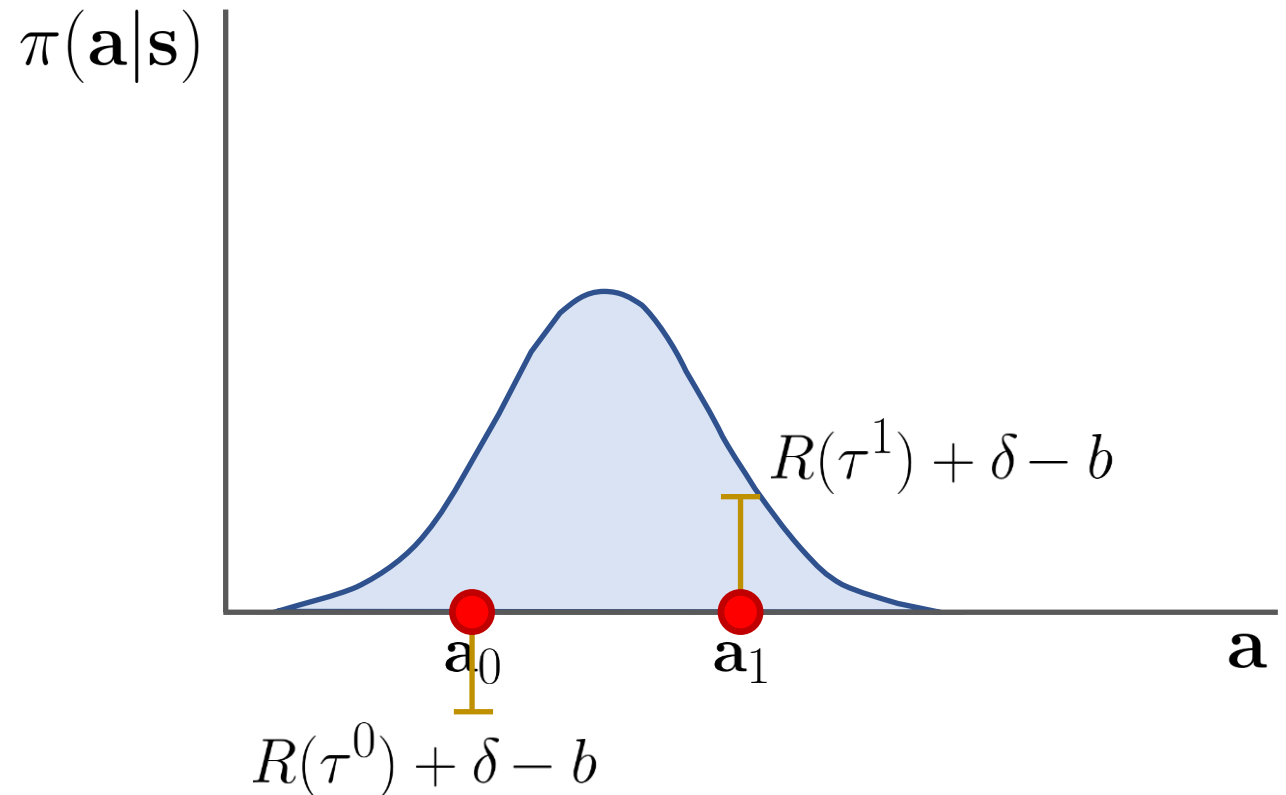
$$\nabla_{\pi} J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[(R(\tau) - b) \sum_{t=0}^{T-1} \nabla_{\pi} \log \pi(\mathbf{a}_t | \mathbf{s}_t) \right]$$

baseline

e.g. $b = \delta$

$\pi(\mathbf{a}|\mathbf{s})$

- Baseline reduces variance
- Is this allowed?
- What is the optimal baseline?

$R(\tau^1) + \delta - b$

$\mathbf{a}_0$ $\mathbf{a}_1$ $\mathbf{a}$

$R(\tau^0) + \delta - b$

Optimal Baseline

- Minimize variance of gradient estimator

$$\mathrm{Var}\left[x\right] = \mathbb{E}[x^2] - (\mathbb{E}\left[x\right])^2$$

$$\mathrm{Var}\left[\nabla_\pi J(\pi)\right] = \mathrm{Var}\left[(R(\tau) - b)\,\nabla_\pi \log p(\tau|\pi)\right]$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)}\left[((R(\tau) - b)\,\nabla_\pi \log p(\tau|\pi))^2\right] - \left(\underbrace{\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[(R(\tau) - b)\,\nabla_\pi \log p(\tau|\pi)\right]}\right)^2$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)}\left[R(\tau)\nabla_\pi \log p(\tau|\pi)\right]$$

$$= \nabla_\pi J(\pi)$$

independent of baseline

Optimal Baseline

- Minimize variance of gradient estimator

$$\text{Var}\,[x] = \mathbb{E}[x^2] - (\mathbb{E}\,[x])^2$$

$$\text{Var}\,[\nabla_\pi J(\pi)] = \text{Var}\,[(R(\tau) - b)\,\nabla_\pi \log p(\tau|\pi)]$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)}\left[((R(\tau) - b)\,\nabla_\pi \log p(\tau|\pi))^2\right] - \left(\mathbb{E}_{\tau \sim p(\tau|\pi)}\,[(R(\tau) - b)\,\nabla_\pi \log p(\tau|\pi)]\right)^2$$

Optimal Baseline

$$\frac{d\text{Var}}{db} = \frac{d}{db}\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[(R(\tau) - b)^2 \left(\nabla_\pi \log p(\tau|\pi)\right)^2\right] = 0$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)}\left[-2\left(R(\tau) - b\right)\left(\nabla_\pi \log p(\tau|\pi)\right)^2\right]$$

$$= -2\,\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[R(\tau)\left(\nabla_\pi \log p(\tau|\pi)\right)^2\right] + 2b\,\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[\left(\nabla_\pi \log p(\tau|\pi)\right)^2\right]$$

$$2b\,\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[\left(\nabla_\pi \log p(\tau|\pi)\right)^2\right] = 2\,\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[R(\tau)\left(\nabla_\pi \log p(\tau|\pi)\right)^2\right]$$

$$b = \frac{\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[R(\tau)\left(\nabla_\pi \log p(\tau|\pi)\right)^2\right]}{\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[\left(\nabla_\pi \log p(\tau|\pi)\right)^2\right]}$$

$$w(\tau) = \left(\nabla_\pi \log p(\tau|\pi)\right)^2 \implies b = \frac{\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[R(\tau)w(\tau)\right]}{\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[w(\tau)\right]}$$

Optimal Baseline

$$R(\tau) \Longrightarrow \hat{R}(\tau) = R(\tau) - b$$

- Optimal baseline:

$$b = \frac{\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[R(\tau)w(\tau)\right]}{\mathbb{E}_{\tau \sim p(\tau|\pi)}\left[w(\tau)\right]} \quad \text{where} \quad w(\tau) = \left(\nabla_\pi \log p(\tau|\pi)\right)^2$$

- In practice:

$$b = \mathbb{E}_{\tau \sim p(\tau|\pi)}\left[R(\tau)\right]$$

easier to estimate

Optimal Baseline

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[(R(\tau) - b) \sum_{t=0}^{T-1} \nabla_\pi \log\pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$

where

$$b = \mathbb{E}_{\tau \sim p(\tau|\pi)} [R(\tau)]$$

- Interpretation:
  - Increase likelihood of trajectories that do *better* than average
  - Decrease likelihood of trajectories that do *worse* than average

Optimal Baseline

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\underbrace{(R(\tau) - b)}_{> 0} \sum_{t=0}^{T-1} \underbrace{\nabla_\pi \mathrm{log} \pi(\mathbf{a}_t|\mathbf{s}_t)}_{\text{increase likelihood}} \right]$$

where

$$b = \mathbb{E}_{\tau \sim p(\tau|\pi)} [R(\tau)]$$

- Interpretation:
  - Increase likelihood of trajectories that do *better* than average
  - Decrease likelihood of trajectories that do *worse* than average

Optimal Baseline

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau | \pi)} \left[\underbrace{(R(\tau) - b)}_{< 0} \sum_{t=0}^{T-1} \underbrace{\nabla_\pi \mathrm{log} \pi(\mathbf{a}_t | \mathbf{s}_t)}_{\text{decrease likelihood}} \right]$$

where

$$b = \mathbb{E}_{\tau \sim p(\tau | \pi)} \left[R(\tau) \right]$$

- Interpretation:
  - Increase likelihood of trajectories that do *better* than average
  - Decrease likelihood of trajectories that do *worse* than average

Policy Gradient

ALGORITHM: Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters

2: **while** not done **do**
3:      Sample trajectories $\{\tau^i\}$ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
4:      Estimate baseline $b = \frac{1}{N} R\left(\tau^i\right)$
5:      Estimate policy gradient
        $\nabla_\theta J(\pi_\theta) \approx \frac{1}{N} \sum_i \left(R(\tau^i) - b\right) \sum_t \nabla_\theta \log \pi_\theta(\mathbf{a}_t^i|\mathbf{s}_t^i)$
6:      Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
7: **end while**

8: return policy π_θ

Policy Gradient Methods for Reinforcement Learning with Function Approximation [Sutton et al. 1999]

Policy Gradient

ALGORITHM: Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters

2: **while** not done **do**
3:      Sample trajectories $\{\tau^i\}$ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
4:      Estimate baseline $b = \frac{1}{N} R(\tau^i)$
5:      Estimate policy gradient
        $\nabla_\theta J(\pi_\theta) \approx \frac{1}{N} \sum_i (R(\tau^i) - b) \sum_t \nabla_\theta \log \pi_\theta(\mathbf{a}_t^i|\mathbf{s}_t^i)$
6:      Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
7: **end while**

8: return policy π_θ

Policy Gradient Methods for Reinforcement Learning with Function Approximation [Sutton et al. 1999]

Variance Reduction

- Baselines
- <u>Causality</u>
- Bootstrapping

Variance Reduction: Causality

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[(R(\tau) - b) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t | \mathbf{s}_t) \right]$$

$$= \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t | \mathbf{s}_t) \right]$$

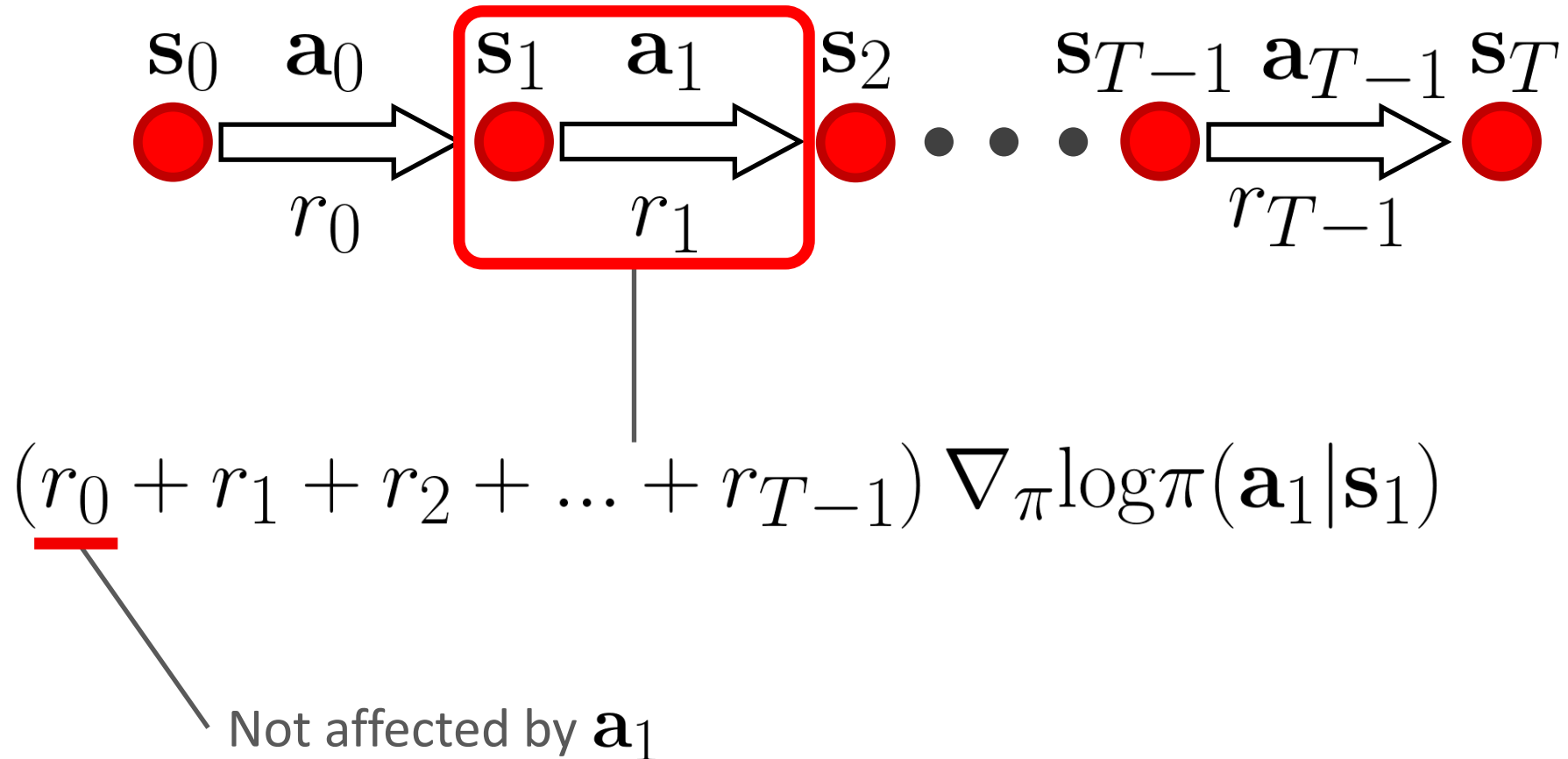rewards across all timesteps

Variance Reduction: Causality

- Gradient at single timestep t

$$\left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \nabla_\pi \log \pi(\mathbf{a}_t | \mathbf{s}_t)$$
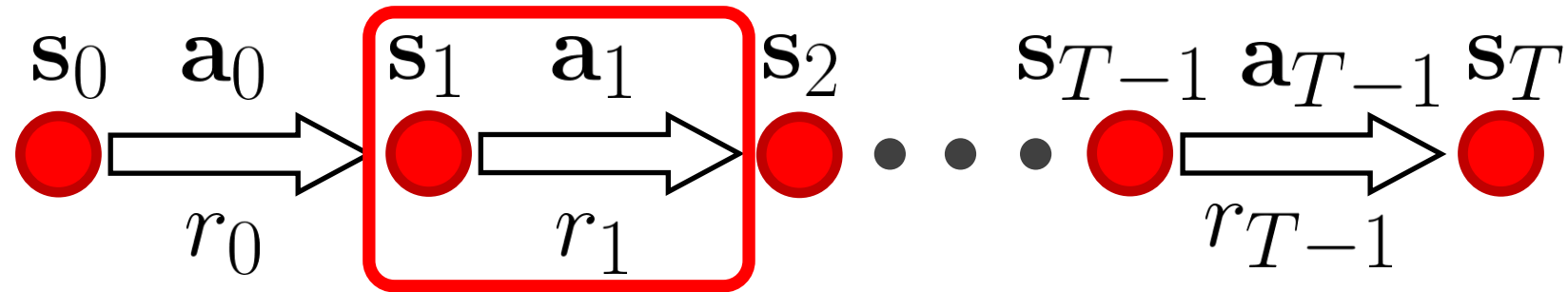
sum rewards across all timesteps

- Current action *does not* affect past rewards
- $r_{t'}$ is independent of $\mathbf{a}_t$ for all $t' < t$

Variance Reduction: Causality

$$(r_0 + r_1 + r_2 + ... + r_{T-1}) \nabla_\pi \log \pi(\mathbf{a}_1 | \mathbf{s}_1)$$

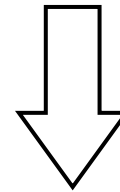Not affected by $\mathbf{a}_1$

Variance Reduction: Causality

$$(r_1 + r_2 + ... + r_{T-1}) \nabla_\pi \log \pi(\mathbf{a}_1 | \mathbf{s}_1)$$

Generally:

$$(r_t + r_{t+1} + ... + r_{T-1}) \nabla_\pi \log \pi(\mathbf{a}_t | \mathbf{s}_t)$$

Variance Reduction: Causality

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\sum_{t=0}^{T-1} \nabla_\pi \mathrm{log} \pi(\mathbf{a}_t|\mathbf{s}_t) \left(\underline{\sum_{t'=0}^{T-1}} \gamma^{t'} r_{t'} - b \right) \right]$$

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\sum_{t=0}^{T-1} \nabla_\pi \mathrm{log} \pi(\mathbf{a}_t|\mathbf{s}_t) \left(\underline{\sum_{t'=t}^{T-1}} \gamma^{t'} r_{t'} - b \right) \right]$$

"reward-to-go"
fewer reward terms → lower variance

Variance Reduction: Causality

- Trajectory-based estimator:

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

- Reward-to-Go estimator:

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0=\mathbf{s}, \mathbf{a}_0=\mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

Variance Reduction: Causality

- Trajectory-based estimator:

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau | \pi)} \left[\sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t | \mathbf{s}_t) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

- Reward-to-Go estimator:

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a} | \mathbf{s})} \mathbb{E}_{\tau \sim p(\tau | \pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a} | \mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

treat every state as start of
a new trajectory

Variance Reduction: Causality

- Trajectory-based estimator:

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[\sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

- Reward-to-Go estimator:

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0=\mathbf{s}, \mathbf{a}_0=\mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

"discounted" state distribution
of the policy π

sum future rewards

Discounted State Distribution

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim p(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})\left(\sum_{t'=0}^{T-1}\gamma^{t'}r_{t'}-b\right)\right]$$

"discounted" state distribution

$$d_\pi(\mathbf{s}) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t p(\mathbf{s}_t=\mathbf{s}|\pi)$$

Discounted State Distribution

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0=\mathbf{s}, \mathbf{a}_0=\mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

"discounted" state distribution

$$d_\pi(\mathbf{s}) = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t p(\mathbf{s}_t = \mathbf{s} | \pi)$$

probability of being in $\mathbf{s}$ after
following π for t timesteps

Discounted State Distribution

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0=\mathbf{s}, \mathbf{a}_0=\mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

"discounted" state distribution

$$d_\pi(\mathbf{s}) = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t p(\mathbf{s}_t = \mathbf{s}|\pi)$$
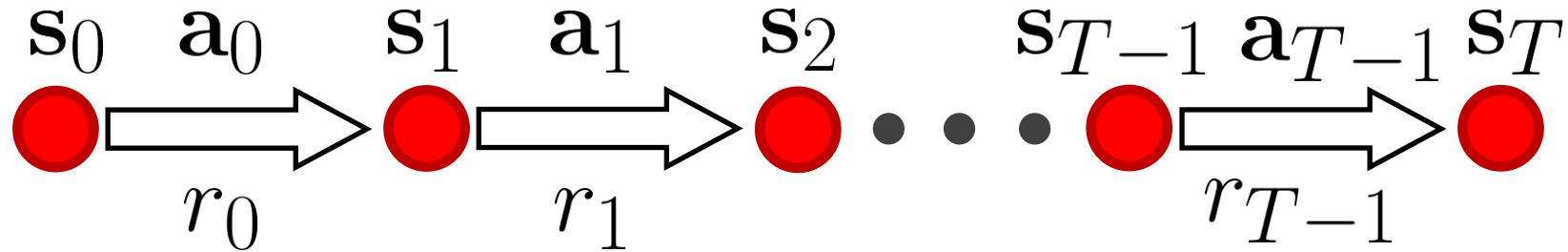
Discounted State Distribution

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

"discounted" state distribution

$$d_\pi(\mathbf{s}) = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t p(\mathbf{s}_t = \mathbf{s}|\pi)$$

Discounted State Distribution

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

"discounted" state distribution

$$d_\pi(\mathbf{s}) = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t p(\mathbf{s}_t = \mathbf{s}|\pi) \implies p(\mathbf{s}|\pi)$$

In practice, just use the marginal state distribution instead

Reward-to-Go Gradient Estimator

$$\mathbf{s}_0 \quad \mathbf{a}_0 \quad \mathbf{s}_1 \quad \mathbf{a}_1 \quad \mathbf{s}_2 \quad \quad \mathbf{s}_{T-1} \quad \mathbf{a}_{T-1} \quad \mathbf{s}_T$$

$$r_0 \qquad\qquad r_1 \qquad\qquad\qquad\qquad r_{T-1}$$

$$\nabla_0 = \left(r_0 + \gamma r_1 + \gamma^2 r_2 + ... + \gamma^{T-1} r_{T-1} \right) \nabla_\pi \mathrm{log}\pi(\mathbf{a}_0|\mathbf{s}_0)$$

$$\nabla_1 = \left(r_1 + \gamma r_2 + \gamma^2 r_3 + ... + \gamma^{T-2} r_{T-1} \right) \nabla_\pi \mathrm{log}\pi(\mathbf{a}_1|\mathbf{s}_1)$$

$$\vdots$$

$$\nabla_{T-1} = \left(r_{T-1} \right) \nabla_\pi \mathrm{log}\pi(\mathbf{a}_{T-1}|\mathbf{s}_{T-1})$$

average grads

$$\approx \nabla_\pi J(\pi)$$

State-Based Baseline

- Reward-to-Go estimator:

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

Can use a better baseline for even lower variance

State-Based Baseline

- Reward-to-Go estimator:

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - V^\pi(\mathbf{s}) \right) \right]$$

Value function baseline

State-Based Baseline

- Reward-to-Go estimator:

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - b \right) \right]$$

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\underbrace{\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - V^\pi(\mathbf{s})} \right) \right]$$

"Advantage"

- Advantage > 0: Action is better than average
- Advantage < 0: Action is worse than average

Value Function Baseline

$$\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim p(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\mathrm{log}\pi(\mathbf{a}|\mathbf{s})\left(\sum_{t'=0}^{T-1}\gamma^{t'}r_{t'}-V^\pi(\mathbf{s})\right)\right]$$

$$=\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim p(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\mathrm{log}\pi(\mathbf{a}|\mathbf{s})\sum_{t'=0}^{T-1}\gamma^{t'}r_{t'}\right]$$

$$-\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim\pi(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\mathrm{log}\pi(\mathbf{a}|\mathbf{s})V^\pi(\mathbf{s})\right]$$

Value Function Baseline

$$\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim p(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})\left(\sum_{t'=0}^{T-1}\gamma^{t'}r_{t'}-V^\pi(\mathbf{s})\right)\right]$$

$$=\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim p(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})\sum_{t'=0}^{T-1}\gamma^{t'}r_{t'}\right]$$

$$-\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim\pi(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})V^\pi(\mathbf{s})\right]$$

Value Function Baseline

$$\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim p(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})\left(\sum_{t'=0}^{T-1}\gamma^{t'}r_{t'}-V^\pi(\mathbf{s})\right)\right]$$

$$= \mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim p(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})\sum_{t'=0}^{T-1}\gamma^{t'}r_{t'}\right]$$

$$-\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim\pi(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})V^\pi(\mathbf{s})\right]$$

Value Function Baseline

$$\mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0=\mathbf{s}, \mathbf{a}_0=\mathbf{a})} \left[\nabla_\pi \log\pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - V^\pi(\mathbf{s}) \right) \right]$$

$$= \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0=\mathbf{s}, \mathbf{a}_0=\mathbf{a})} \left[\nabla_\pi \log\pi(\mathbf{a}|\mathbf{s}) \sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} \right]$$

$$- \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim \pi(\tau|\pi, \mathbf{s}_0=\mathbf{s}, \mathbf{a}_0=\mathbf{a})} \left[\nabla_\pi \log\pi(\mathbf{a}|\mathbf{s}) V^\pi(\mathbf{s}) \right]$$

$$= \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0=\mathbf{s}, \mathbf{a}_0=\mathbf{a})} \left[\nabla_\pi \log\pi(\mathbf{a}|\mathbf{s}) \sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} \right]$$

$$- \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \left[V^\pi(\mathbf{s}) \ \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \left[\nabla_\pi \log\pi(\mathbf{a}|\mathbf{s}) \right] \right]$$

Value Function Baseline

$$\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim p(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})\left(\sum_{t'=0}^{T-1}\gamma^{t'}r_{t'}-V^\pi(\mathbf{s})\right)\right]$$

$$=\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim p(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})\sum_{t'=0}^{T-1}\gamma^{t'}r_{t'}\right]$$

$$-\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim\pi(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})V^\pi(\mathbf{s})\right]$$

$$=\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\mathbb{E}_{\tau\sim p(\tau|\pi,\mathbf{s}_0=\mathbf{s},\mathbf{a}_0=\mathbf{a})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})\sum_{t'=0}^{T-1}\gamma^{t'}r_{t'}\right]$$

$$-\mathbb{E}_{\mathbf{s}\sim d_\pi(\mathbf{s})}\left[V^\pi(\mathbf{s})\underline{\mathbb{E}_{\mathbf{a}\sim\pi(\mathbf{a}|\mathbf{s})}\left[\nabla_\pi\log\pi(\mathbf{a}|\mathbf{s})\right]}\right]$$

$$=\nabla_\pi\sum_{\mathbf{a}}\pi(\mathbf{a}|\mathbf{s})=\nabla_\pi 1$$

Value Function Baseline

$$\mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - V^\pi(\mathbf{s}) \right) \right]$$

$$= \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} \right]$$

$$- \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim \pi(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) V^\pi(\mathbf{s}) \right]$$

$$= \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} \right]$$

$$- \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \left[V^\pi(\mathbf{s}) \underbrace{\mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \right]}_{0} \right]$$

Value function baseline is unbiased!

Value Function Baseline

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau | \pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - V^\pi(\mathbf{s}) \right) \right]$$

- Value function baseline is unbiased!
- Substantial variance reduction
- Any baseline that is only a function of the state is unbiased
  [Sutton et al. 1990]

Policy Gradient Methods for Reinforcement Learning with Function Approximation
[Sutton et al. 1999]

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:     Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:     Fit value function $V(\mathbf{s})$

6:     **for** every timestep t **do**
7:         $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:     **end for**

9:     $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:    Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:     Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:     Fit value function $V(\mathbf{s})$

6:     **for** every timestep t **do**
7:         $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:     **end for**

9:     $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:    Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:    Fit value function $V(\mathbf{s})$

6:    **for** every timestep t **do**
7:        $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:    **end for**

9:    $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:   Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:     Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:     Fit value function $V(\mathbf{s})$

6:     **for** every timestep t **do**
7:         $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:     **end for**

9:     $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:     Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:     Fit value function $V(\mathbf{s})$

6:     **for** every timestep t **do**
7:         $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:     **end for**

9:     $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:      Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:      Fit value function $V(\mathbf{s})$

6:      **for** every timestep t **do**
7:          $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:      **end for**

9:      $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:     Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:     Fit value function $V(\mathbf{s})$

6:     **for** every timestep t **do**
7:         $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:     **end for**

9:     $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:     Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:     Fit value function $V(\mathbf{s})$

6:     **for** every timestep t **do**
7:         $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:     **end for**

9:     $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:     Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:     Fit value function $V(\mathbf{s})$

6:     **for** every timestep t **do**
7:         $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:     **end for**

9:     $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:     Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:     Fit value function $V(\mathbf{s})$

6:     **for** every timestep t **do**
7:         $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - \underline{V(\mathbf{s}_t)} \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:     **end for**

9:     $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:      Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:      Fit value function $V(\mathbf{s})$

6:      **for** every timestep t **do**
7:          $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:      **end for**

9:      $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:      Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:      Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:      Fit value function $V(\mathbf{s})$

6:      **for** every timestep t **do**
7:         $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:      **end for**

9:      $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Reward-to-Go Policy Gradient

ALGORITHM: Reward-to-Go Policy Gradient

1: $\theta \leftarrow$ initialize policy parameters
2: $V \leftarrow$ initialize value function parameters

3: **while** not done **do**
4:     Sample trajectory τ from policy $\pi_\theta(\mathbf{a}|\mathbf{s})$
5:     Fit value function $V(\mathbf{s})$

6:     **for** every timestep t **do**
7:         $\nabla_t \leftarrow \left(\sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'} - V(\mathbf{s}_t) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$
8:     **end for**

9:     $\nabla_\theta J(\pi_\theta) \approx \frac{1}{T} \sum_{t=0}^{T-1} \nabla_t$
10:     Update policy $\theta \leftarrow \theta + \alpha \nabla_\theta J(\pi_\theta)$
11: **end while**

12: return policy π_θ

Variance Reduction

- Baselines

- Causality

- <u>Bootstrapping</u>

Variance Reduction: Bootstrapping

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - V^\pi(\mathbf{s}) \right) \right]$$

sum of random variables → high variance

$$r_0 + \gamma r_1 + \gamma^2 r_2 + \ldots$$

Variance Reduction: Bootstrapping

$$\nabla_{\pi} J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_{\pi}(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_{\pi} \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - V^{\pi}(\mathbf{s}) \right) \right]$$

sum of random variables → high variance

n-step return: $r_0 + \gamma r_1 + \gamma^2 r_2 + \ldots + \gamma^{k-1} r_{k-1}$

Variance Reduction: Bootstrapping

$$\nabla_\pi J(\pi) = \mathbb{E}_{\mathbf{s} \sim d_\pi(\mathbf{s})} \mathbb{E}_{\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})} \mathbb{E}_{\tau \sim p(\tau|\pi, \mathbf{s}_0 = \mathbf{s}, \mathbf{a}_0 = \mathbf{a})} \left[\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t'=0}^{T-1} \gamma^{t'} r_{t'} - V^\pi(\mathbf{s}) \right) \right]$$

sum of random variables → high variance

n-step return: $\quad r_0 + \gamma r_1 + \gamma^2 r_2 + ... + \gamma^{k-1} r_{k-1} + \gamma^k V^\pi(\mathbf{s}_k)$

bootstrap

N-Step Bootstrapping

1-step bootstrap: $\quad y = r_0 + \gamma \hat{V}^\pi(\mathbf{s}_1)$

2-step bootstrap: $\quad y = r_0 + \gamma r_1 + \gamma^2 \hat{V}^\pi(\mathbf{s}_2)$

3-step bootstrap: $\quad y = r_0 + \gamma r_1 + \gamma^2 r_2 + \gamma^3 \hat{V}^\pi(\mathbf{s}_3)$

$\bullet$
$\bullet$
$\bullet$

n-step bootstrap: $\quad y = \sum_{t=0}^{n-1} \gamma^t r_t + \gamma^n \hat{V}^\pi(\mathbf{s}_n)$

High variance $\qquad\qquad$ Biased

TD(λ)

- Use TD(λ) to estimate return

$$\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t=0} \gamma^t r_t - V^\pi(\mathbf{s}) \right)$$

$$\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(R^\lambda(\mathbf{s}, \mathbf{a}) - V^\pi(\mathbf{s}) \right)$$

λ-return

TD(λ)

$1 - \lambda$

$(1 - \lambda)\lambda$

$(1 - \lambda)\lambda^2$

$\sum = 1$

S_t
A_t
$S_{t+1} \ R_{t+1}$
A_{t+1}
$S_{t+2} \ R_{t+2}$
A_{t+2}
A_{T-1}
$S_T \ R_T$
λ^{T-t-1}
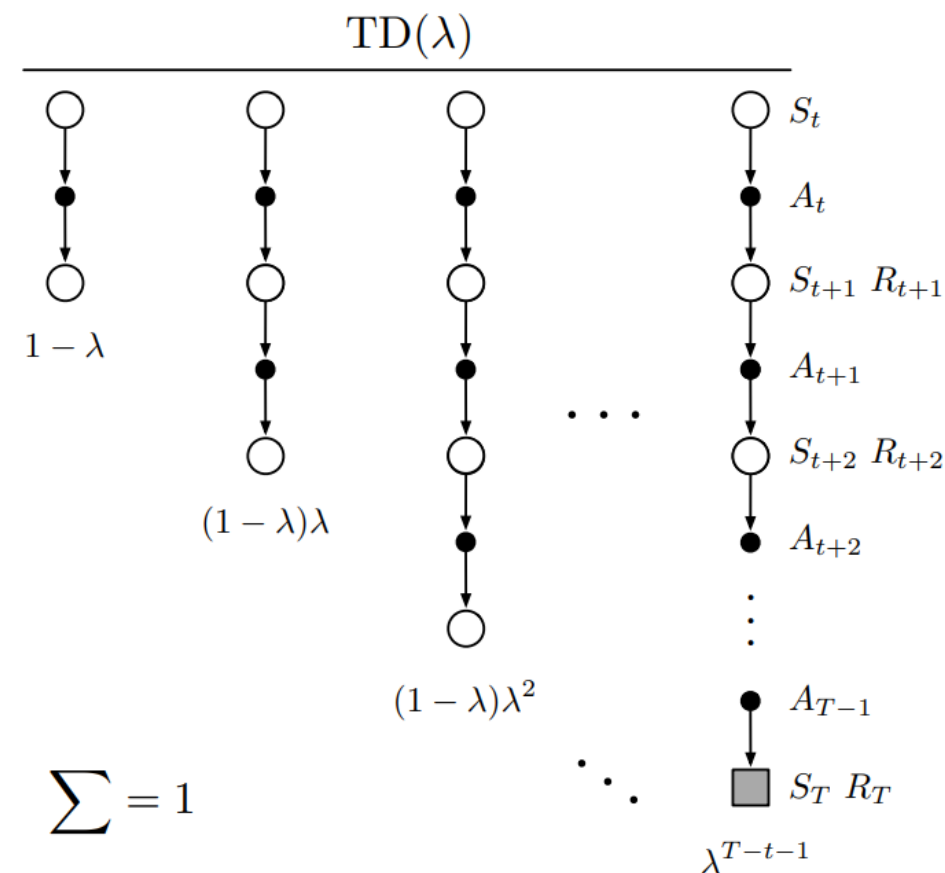
Reinforcement Learning: An Introduction
[Sutton and Barto 1998]

TD(λ)

- Use TD(λ) to estimate return

$$\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(\sum_{t=0} \gamma^t r_t - V^\pi(\mathbf{s}) \right)$$
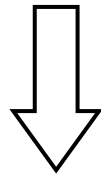
$$\nabla_\pi \log \pi(\mathbf{a}|\mathbf{s}) \left(R^\lambda(\mathbf{s}, \mathbf{a}) - V^\pi(\mathbf{s}) \right)$$

Generalized Advantage Estimation (GAE)

High-Dimensional Continuous Control Using
Generalized Advantage Estimation
[Schulman et al. 2016]

TD(λ)

$$\sum = 1$$
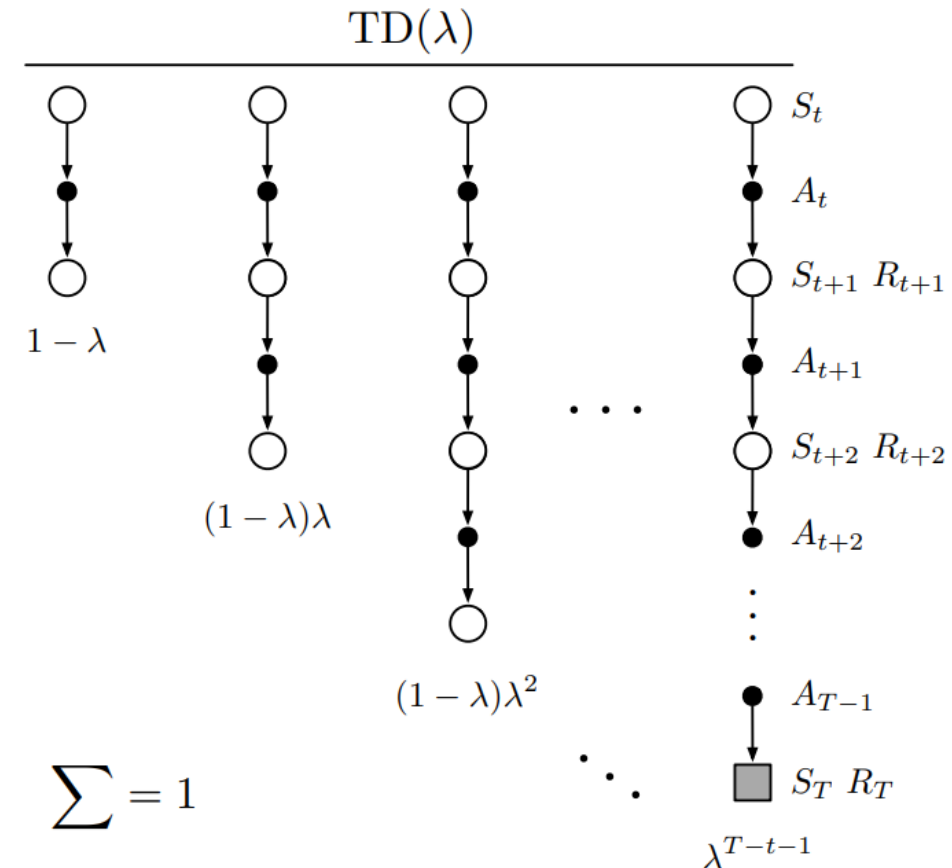
Reinforcement Learning: An Introduction
[Sutton and Barto 1998]

Variance Reduction

- Baselines

- Causality

- Bootstrapping

Applications

Visual Navigation
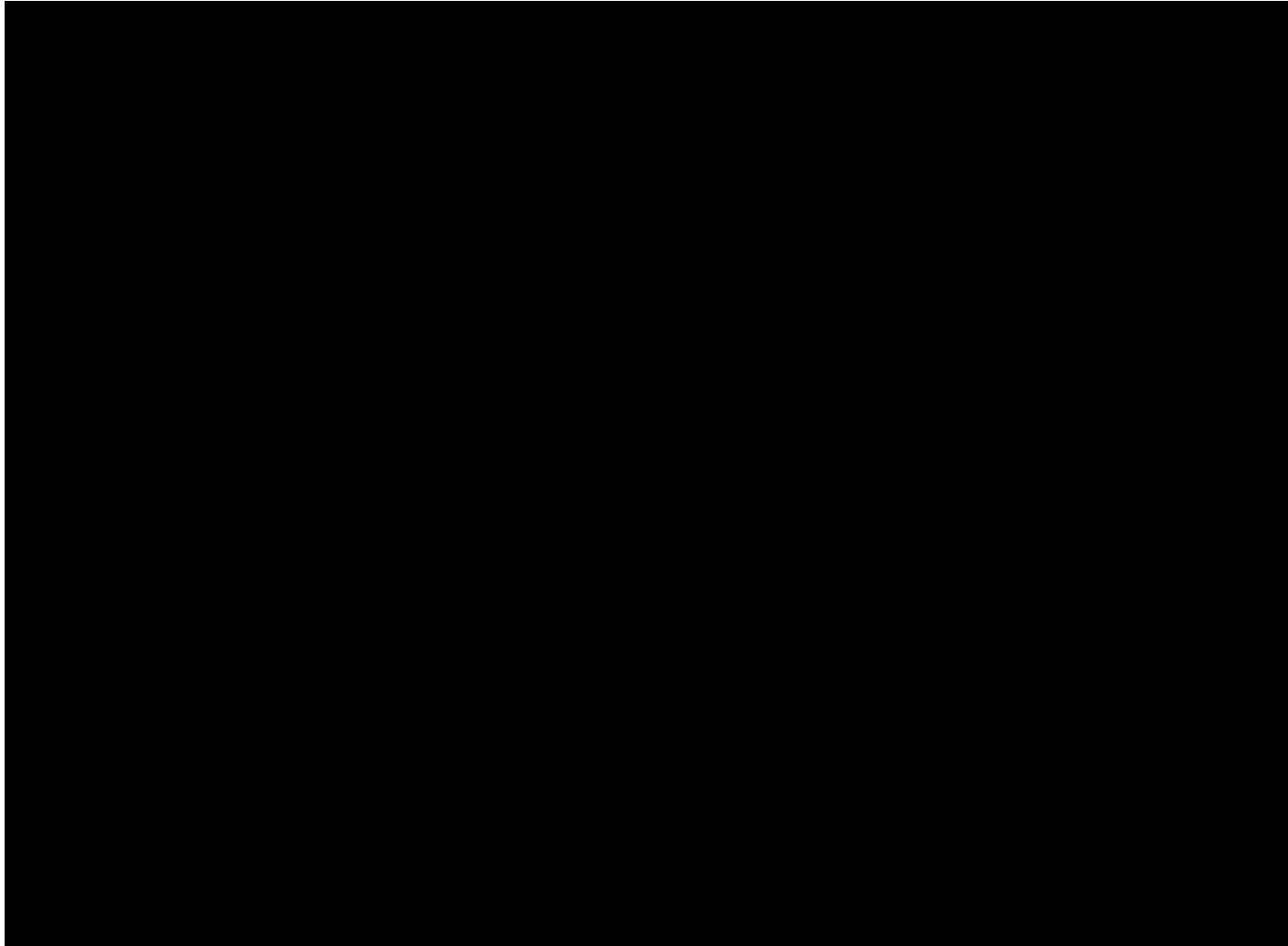
Asynchronous Methods for Deep Reinforcement Learning
[Mnih et al. 2016]

Visual Navigation
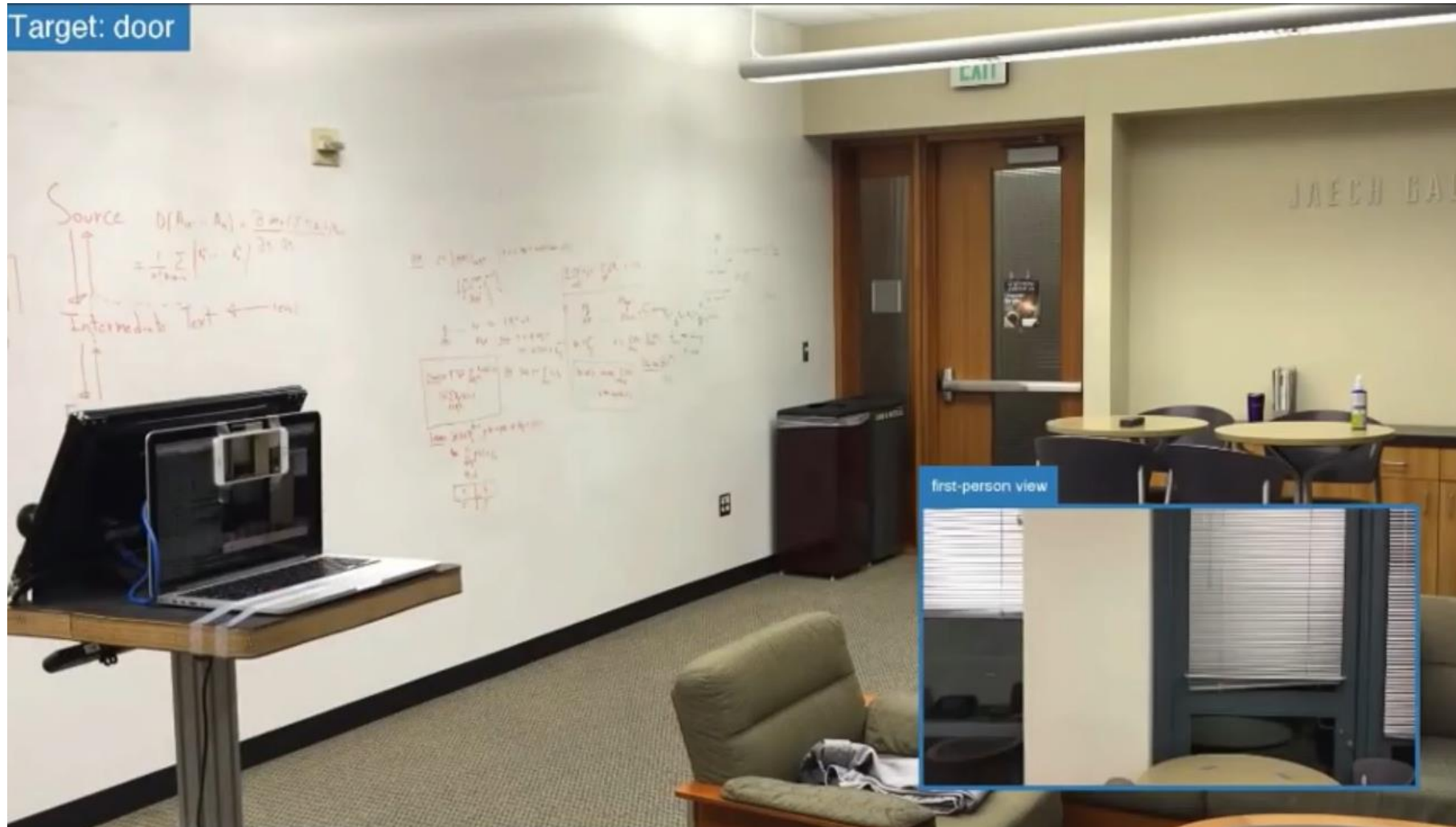
Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning
[Zhu et al. 2017]

Robotic Locomotion
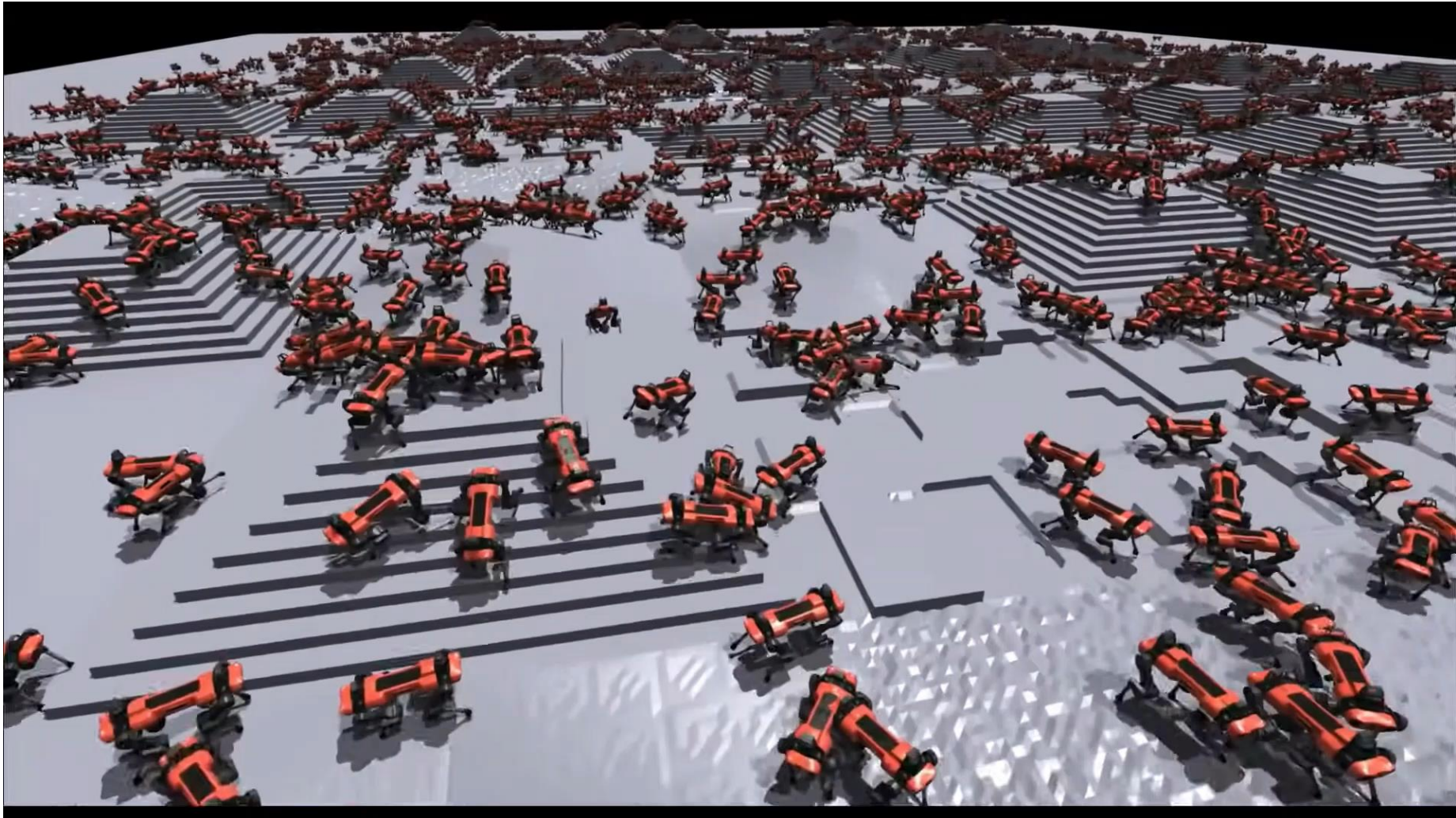
Learning to Walk in Minutes Using Massively Parallel Deep Reinforcement Learning
[Rudin et al. 2022]

Robotic Locomotion
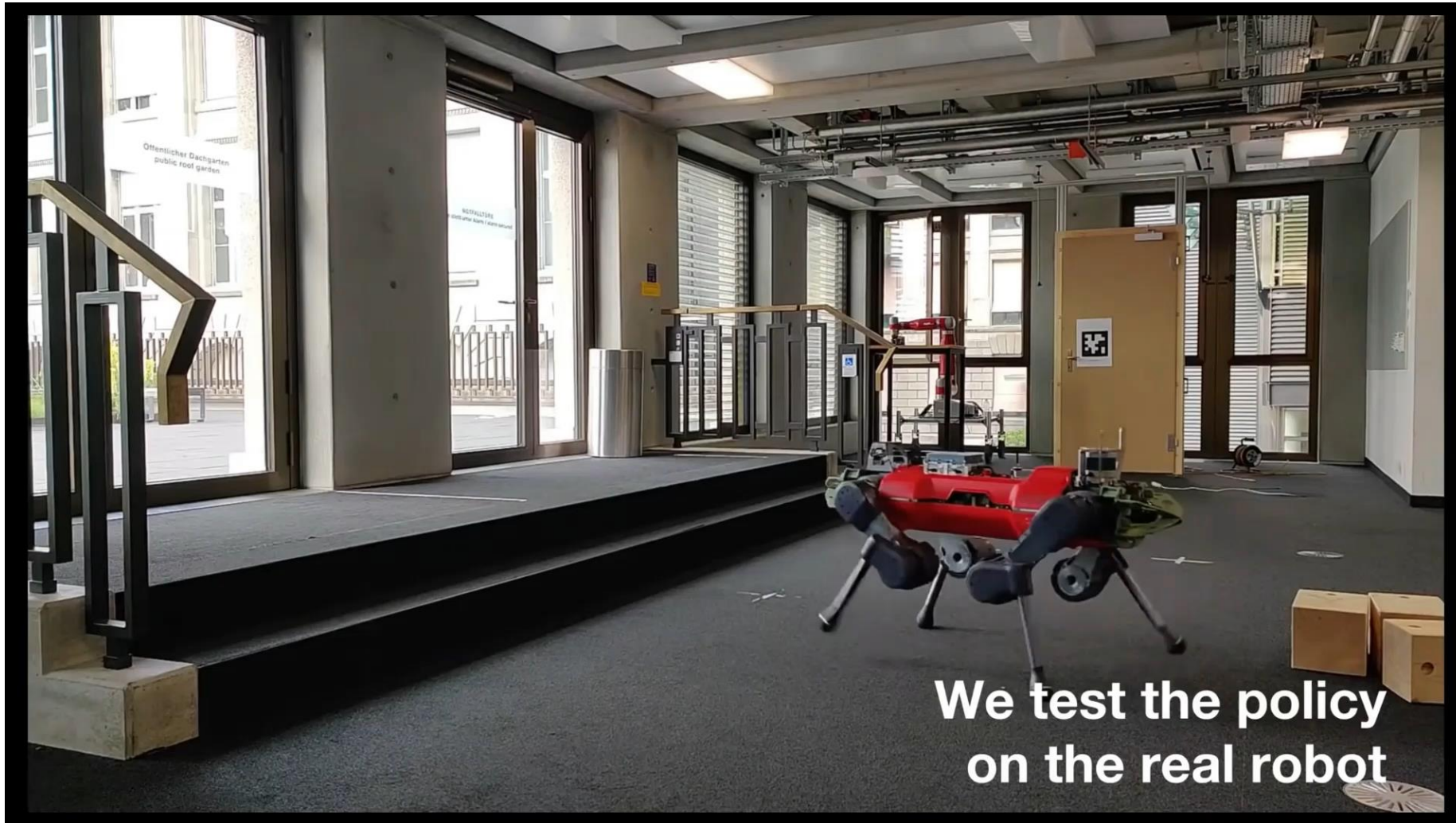
Learning to Walk in Minutes Using Massively Parallel Deep Reinforcement Learning
[Rudin et al. 2022]

Policy Gradient

✓ Directly optimize $J(\pi)$ by estimating gradient $\nabla_\pi J(\pi)$

✓ General: can be applied to continuous and discrete states and actions

✗ High-variance gradient estimator $\rightarrow$ unstable/slow convergence

✗ Very sample inefficient
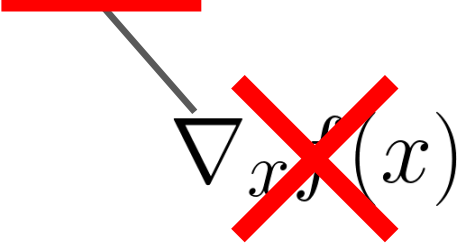
General View of PG

Nondifferentiable Functions

- Why does PG allow us to calculate gradients for a nondifferentiable function?
  - Gradient exists but unknown
  - Gradient does not exist

$$\nabla_\pi J(\pi) = \mathbb{E}_{\tau \sim p(\tau|\pi)} \left[R(\tau) \sum_{t=0}^{T-1} \nabla_\pi \log \pi(\mathbf{a}_t|\mathbf{s}_t) \right]$$

Nondifferentiable Functions

$$\arg\max_x \; \underline{f(x)}$$

$$\nabla_x f(x)$$

Nondifferentiable Functions

$$\arg\max_{x} \ f(x)$$

$f(x)$

x

Nondifferentiable Functions

$$\arg \max_x \ f(x) \implies \arg \max_p \ \mathbb{E}_{x \sim p(x)} \left[f(x) \right]$$

$f(x)$

x

Nondifferentiable Functions

$$\arg\max_x f(x) \implies \arg\max_p \mathbb{E}_{x \sim p(x)}[f(x)]$$

$f(x)$

x

Nondifferentiable Functions

$$\arg \max_x f(x) \implies \arg \max_p \mathbb{E}_{x \sim p(x)} [f(x)]$$

Score
Function

$$\nabla_p = \mathbb{E}_{x \sim p(x)} [f(x) \nabla_p \log p(x)]$$

$f(x)$

x

Nondifferentiable Functions

$$\arg \max_{x} f(x) \implies \arg \max_{p} \mathbb{E}_{x \sim p(x)}[f(x)]$$

Score Function

$$\nabla_p = \mathbb{E}_{x \sim p(x)}[f(x)\nabla_p \log p(x)]$$

$f(x)$

x

Nondifferentiable Functions

$$\arg \max_p \; \mathbb{E}_{x \sim p(x)} \left[f(x) \right]$$

Nondifferentiable Functions

$$\arg\max_{p} \ \underline{\mathbb{E}_{x \sim p(x)}\left[f(x)\right]}$$

$$= \underbrace{\sum_{x} p(x)f(x)}$$

This is a convolution!

Nondifferentiable Functions
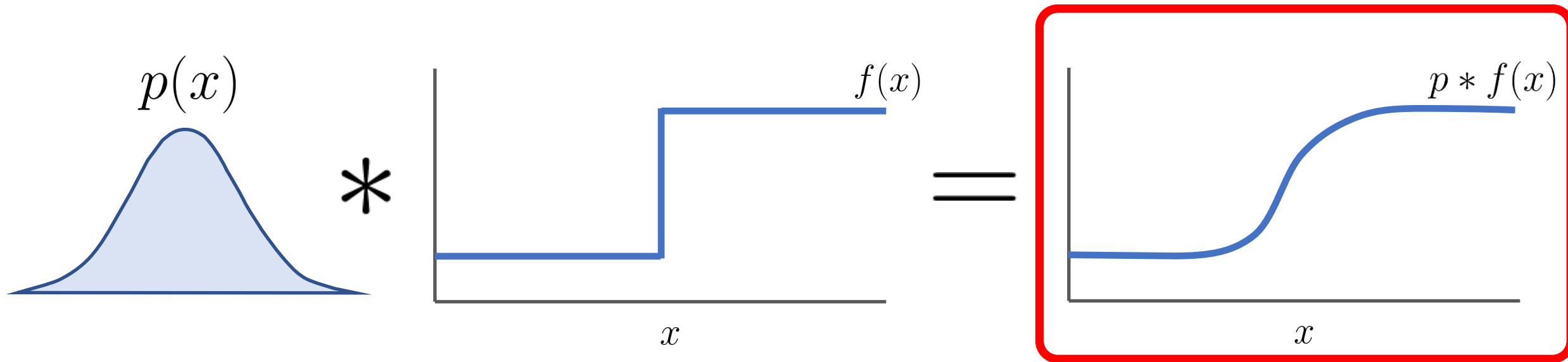
$$\arg \max_{p} \ \mathbb{E}_{x \sim p(x)}\left[f(x)\right]$$

Nondifferentiable Functions

$$\arg \max_{p} \mathbb{E}_{x \sim p(x)} [f(x)]$$
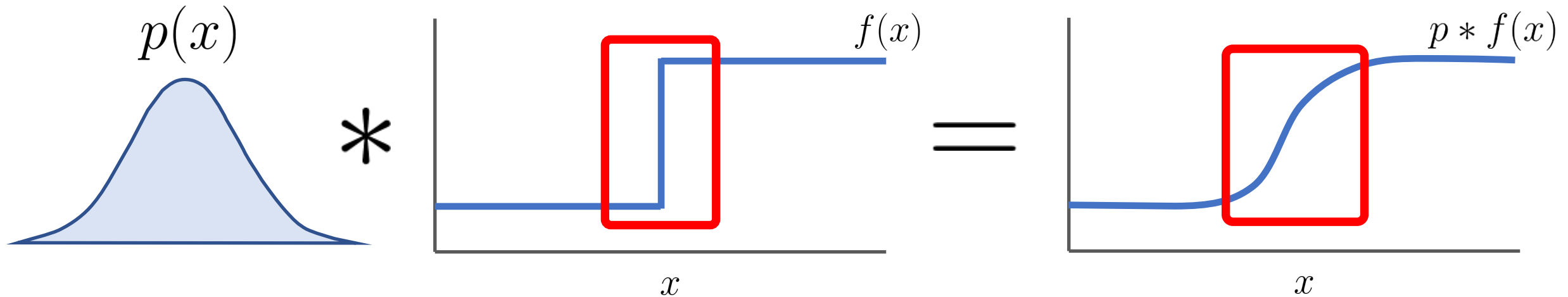
Score Function

- Score function can be applied to estimate gradients for _any_ nondifferentiable function
  - Converts an optimization of a _deterministic_ variable into an optimization of a _stochastic_ distribution

- Policy gradient only works for stochastic policies

$$\mathbf{a} \sim \pi(\mathbf{a}|\mathbf{s})$$

$$\mathbf{a} = \pi(\mathbf{s})$$
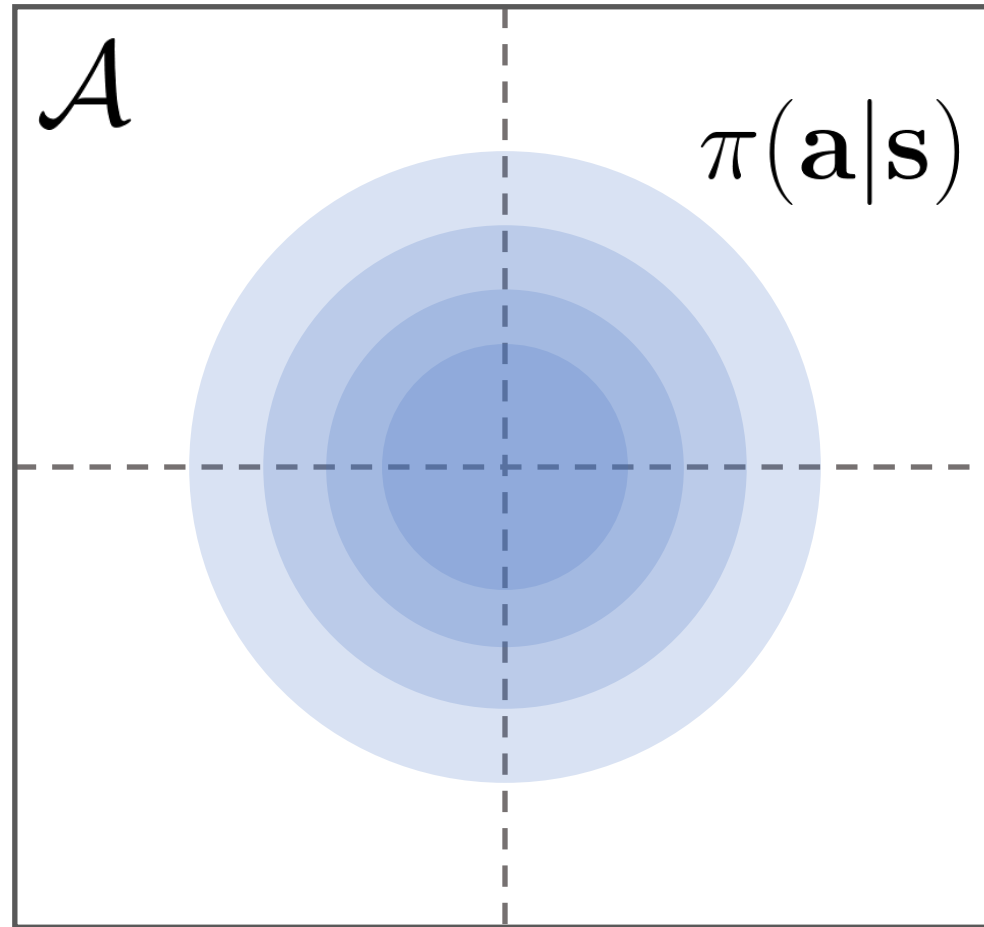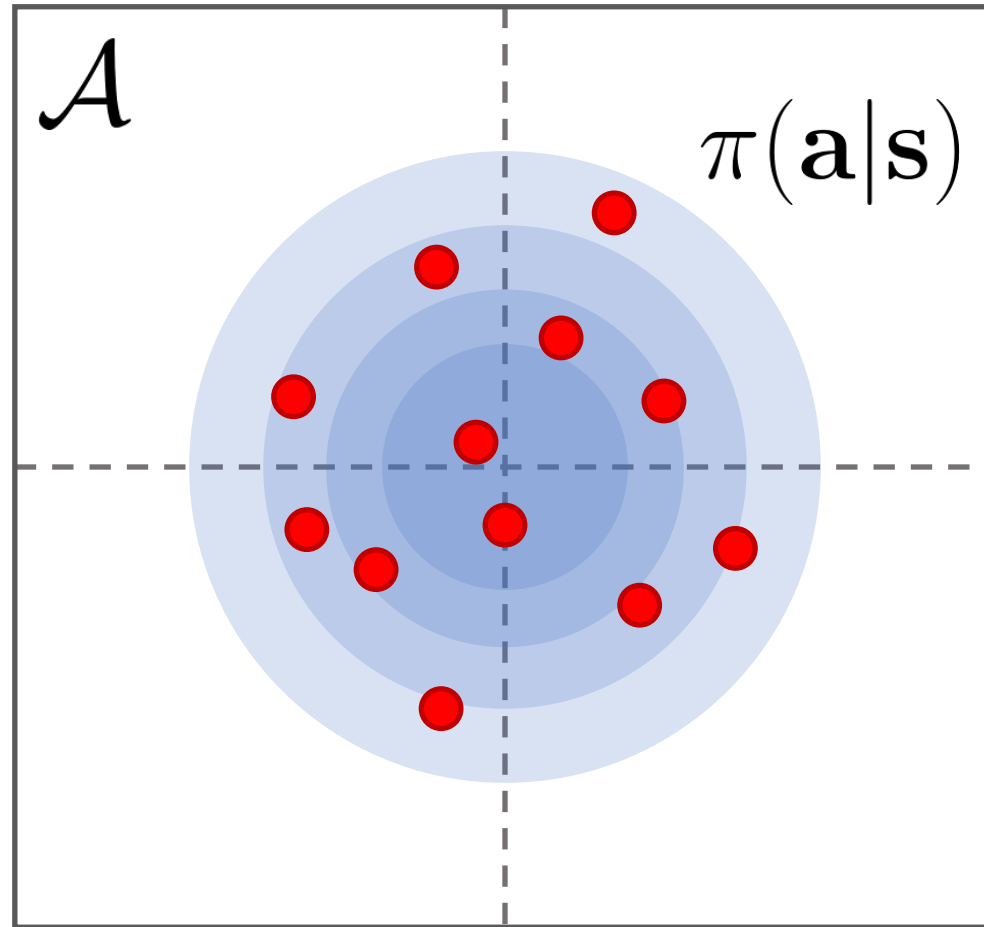
Gradient Estimation Using Stochastic Computation Graphs
[Schulman et al. 2015]

Evolutionary Strategies

Evolutionary Strategies

Evolutionary Strategies
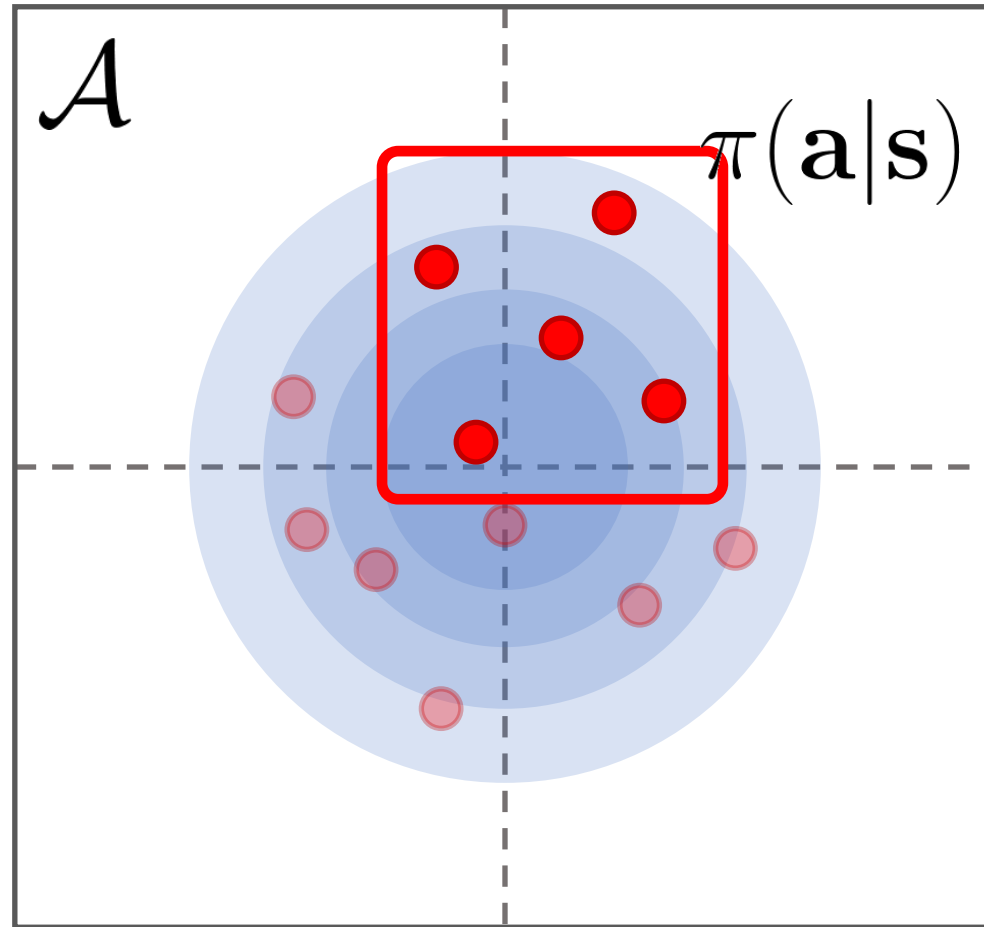
Evolutionary Strategies
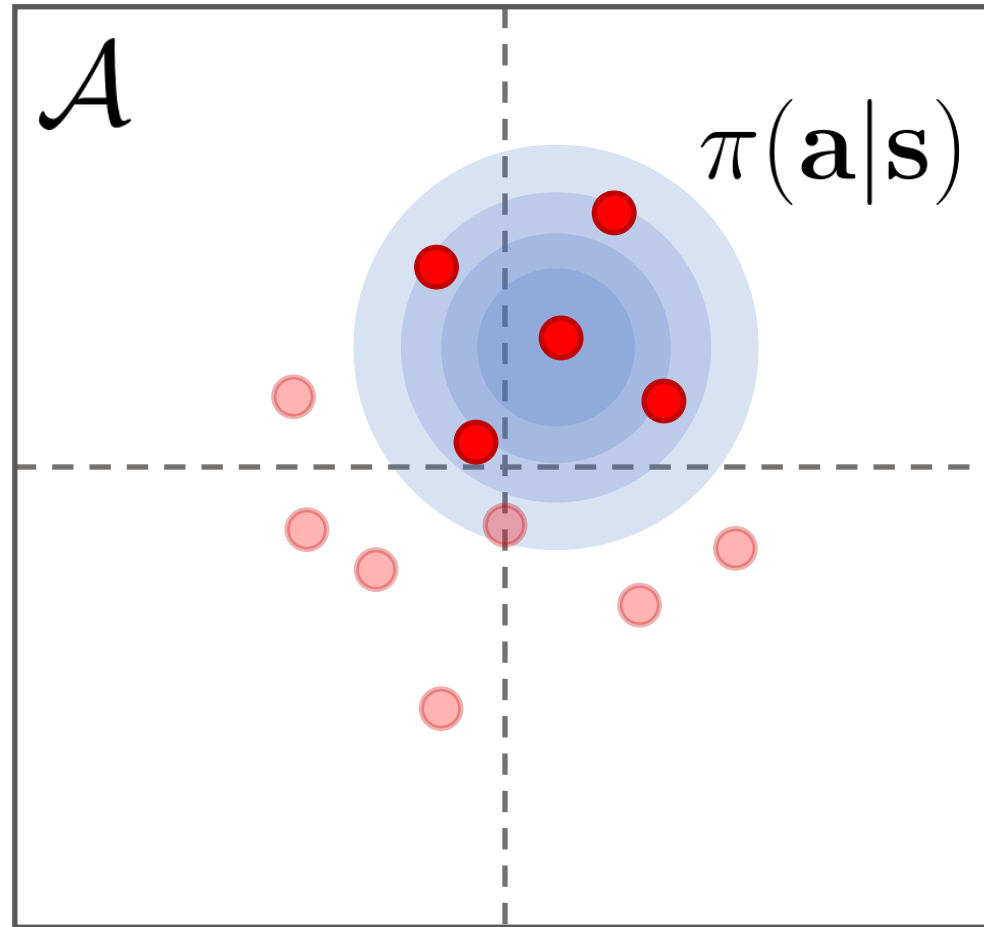
Evolutionary Strategies
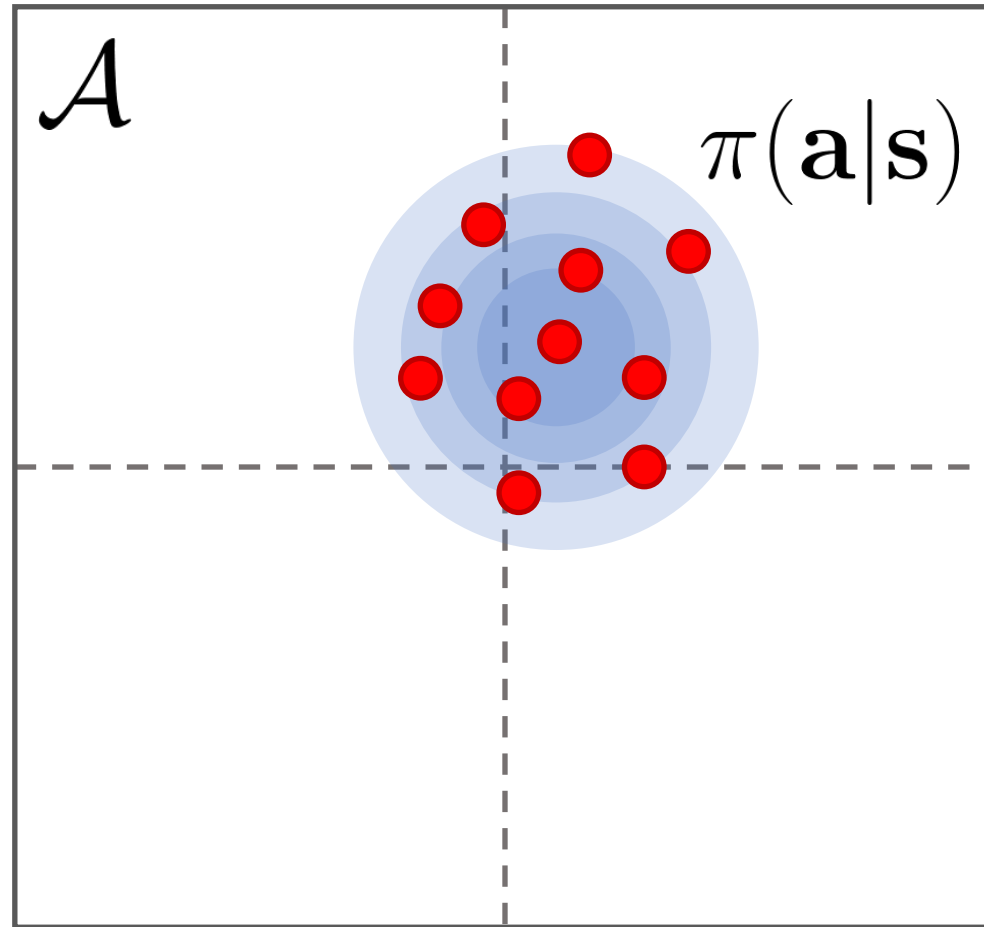
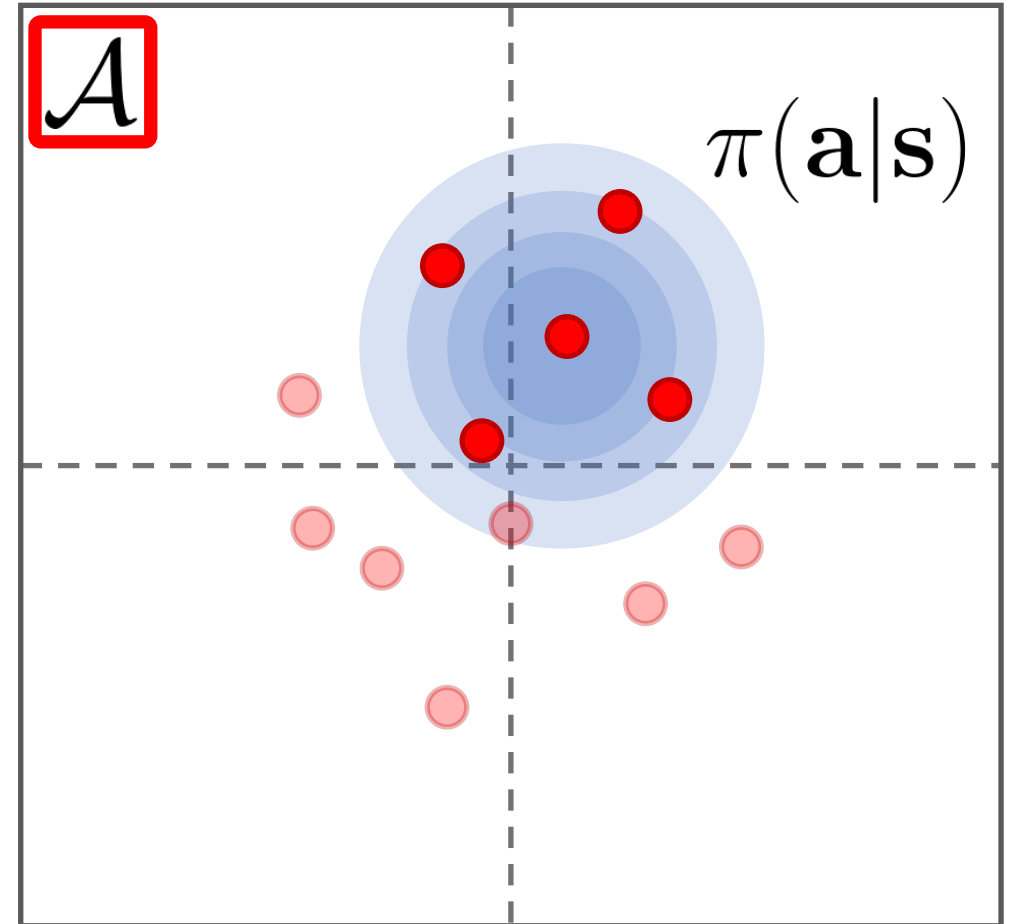# Evolutionary Strategies
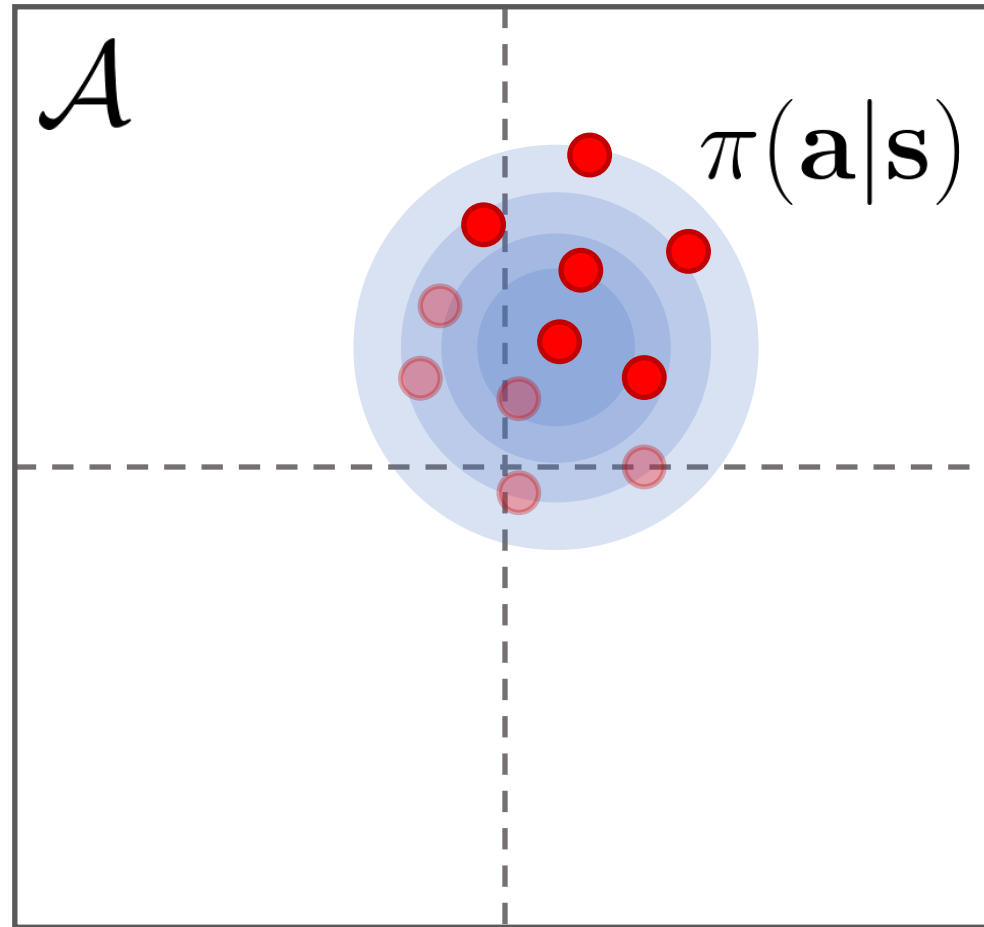
Evolutionary Strategies
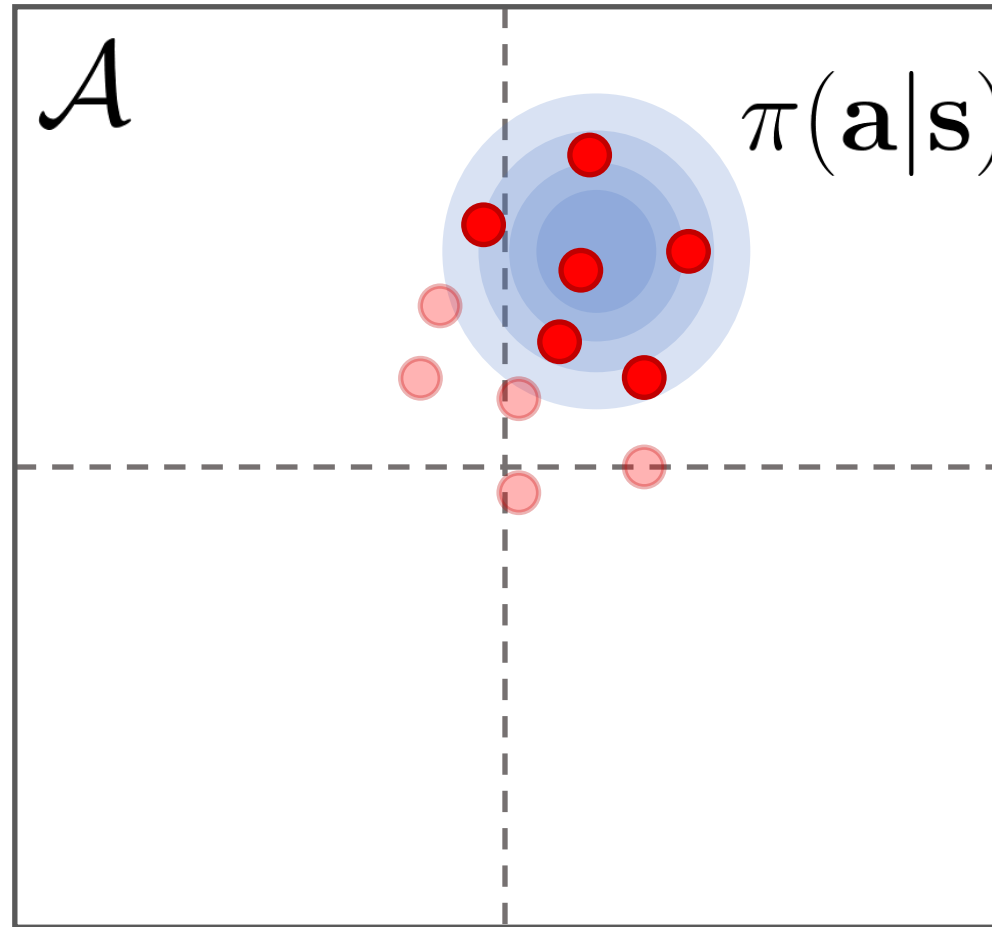
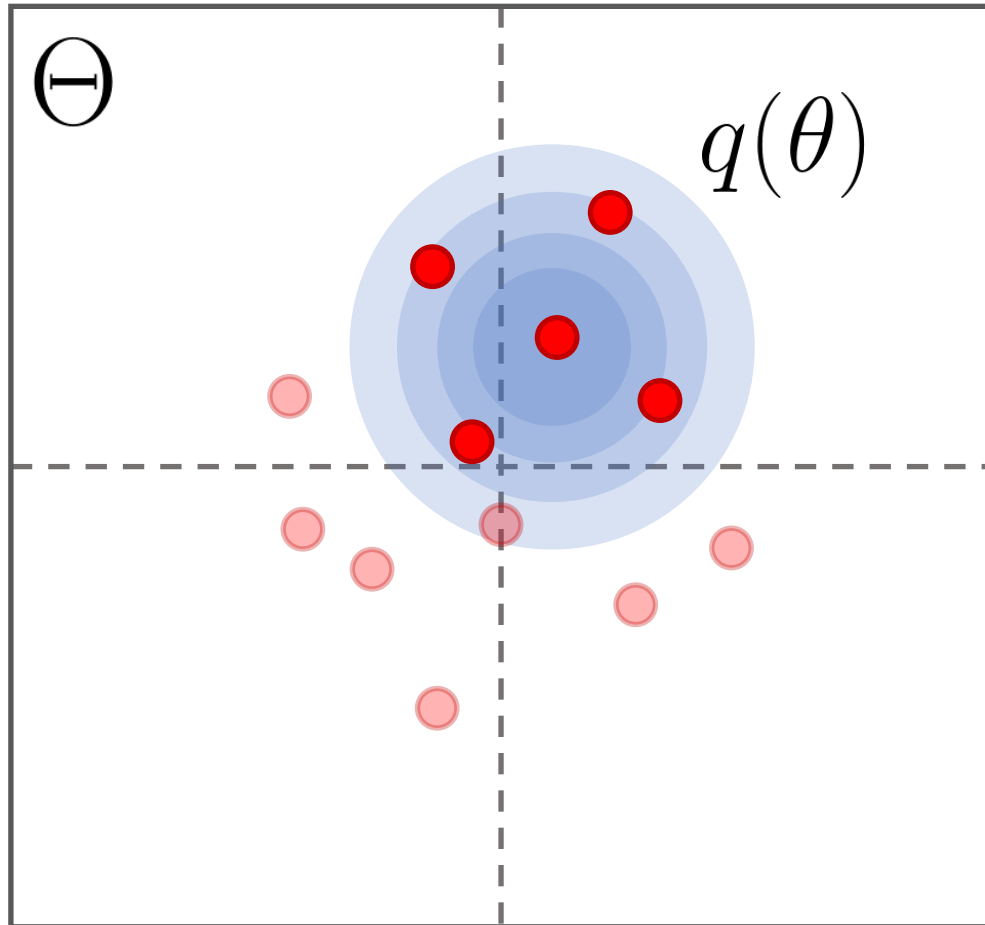# Evolutionary Strategies

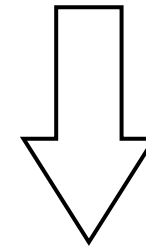Cross-Entropy Method
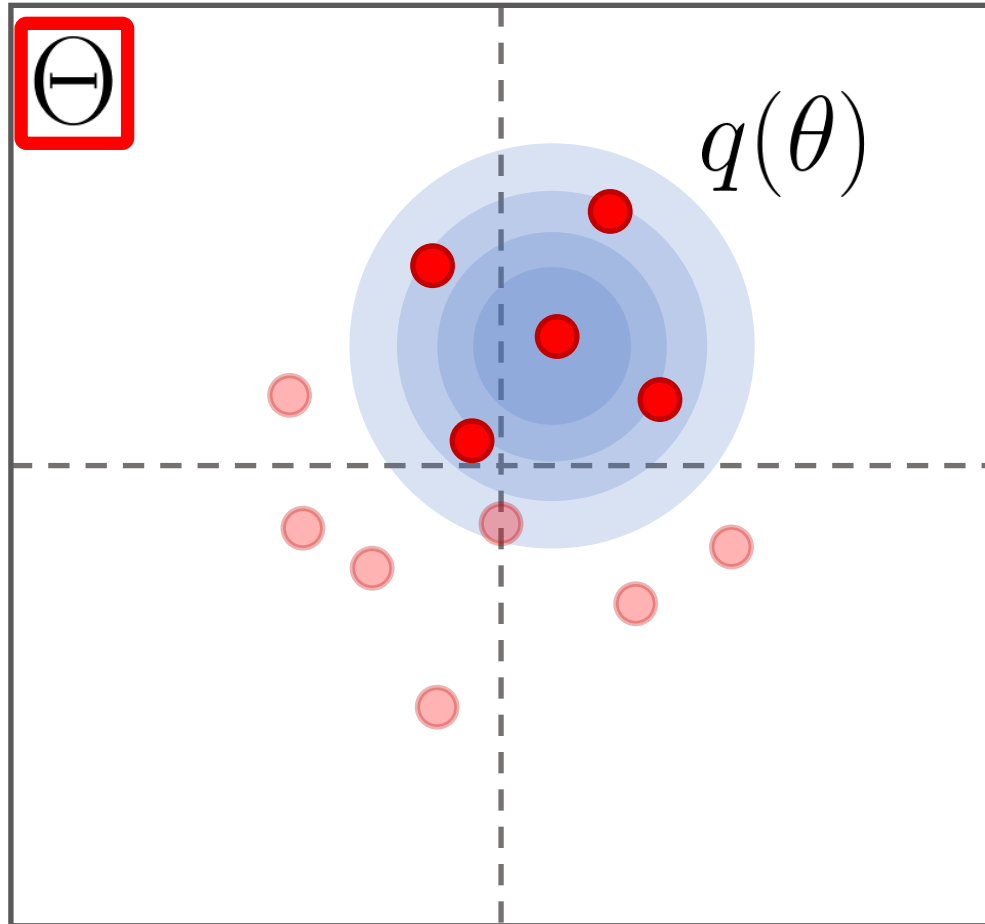
Policy Gradient

Evolutionary Strategies
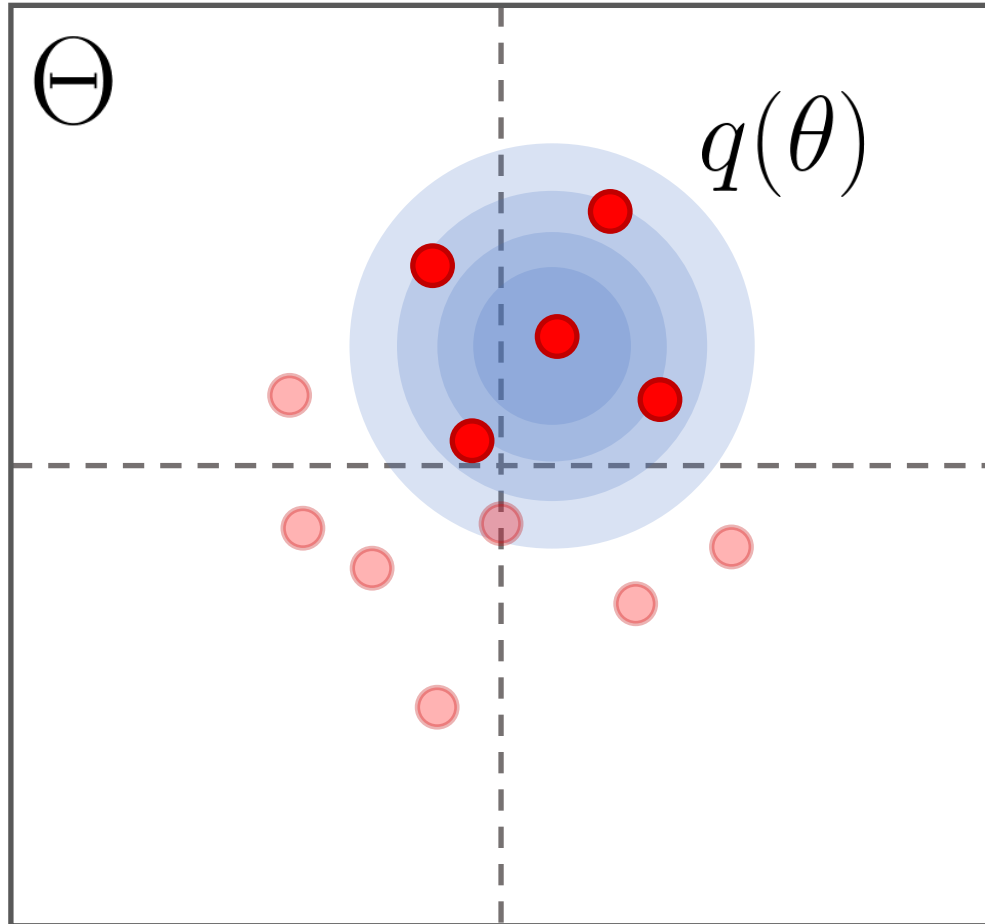
$$J(q) = \underline{\mathbb{E}_{\theta \sim q(\theta)}} \left[J(\pi_\theta) \right]$$

Evolutionary Strategies

$$J(q) = \mathbb{E}_{\theta \sim q(\theta)} \left[J(\pi_\theta) \right]$$

⇓

$$\nabla_q J(q) = \mathbb{E}_{\theta \sim q(\theta)} \left[J(\pi_\theta) \nabla_q \log q(\theta) \right]$$

evolutionary strategy
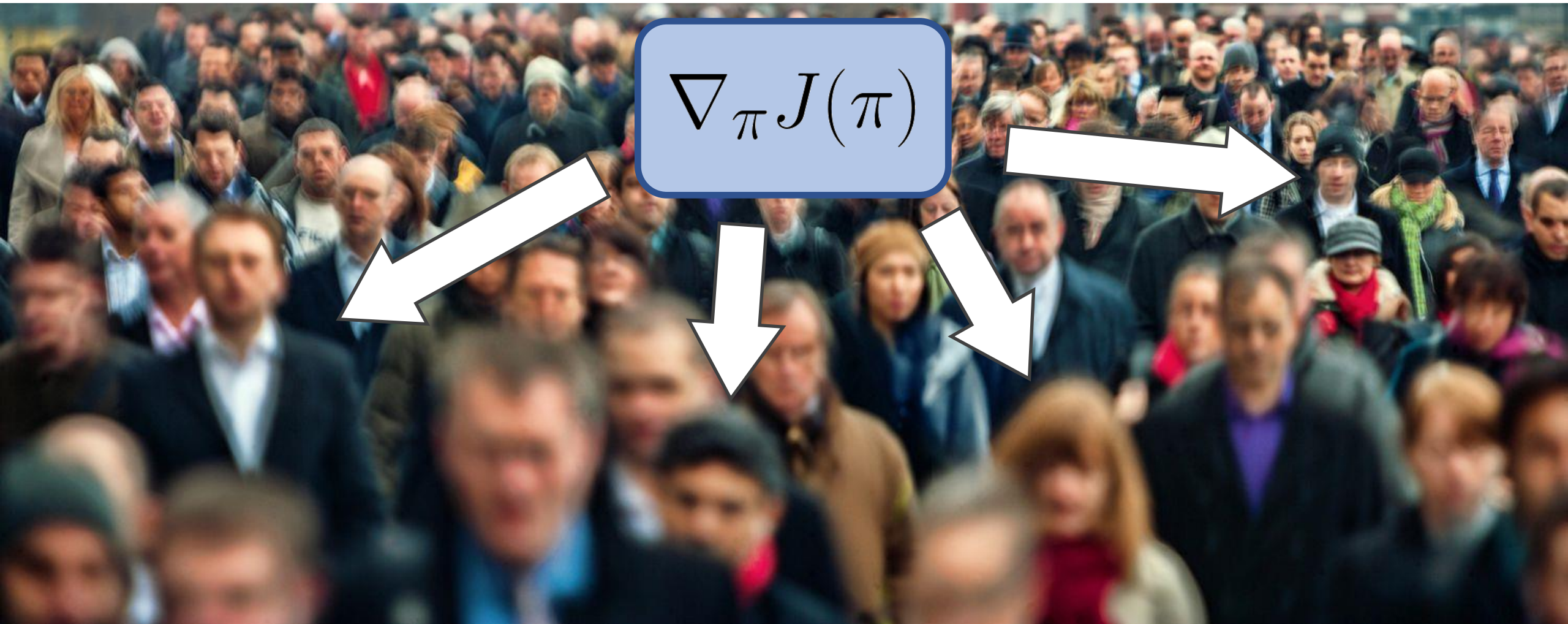
Evolution is doing gradient ascent!

Evolutionary Strategies

Cross-Entropy Method:

- Optimize distribution over parameters

Policy Gradient:

- Optimize distribution over actions

Evolution

$$\nabla_\pi J(\pi)$$

Summary

- Policy Gradient

- Derivation

- Variance Reduction

- Applications

- General View of PG