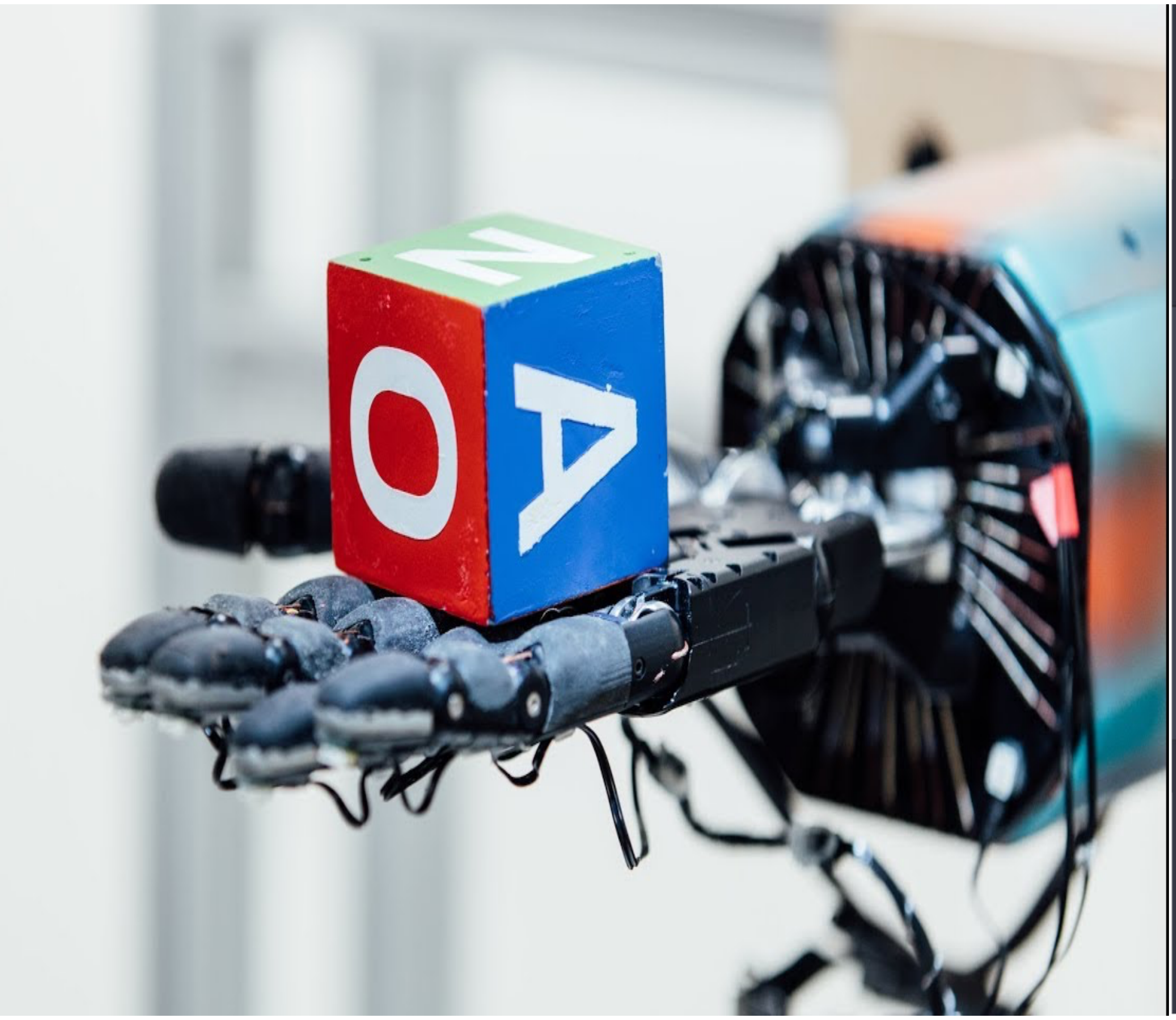
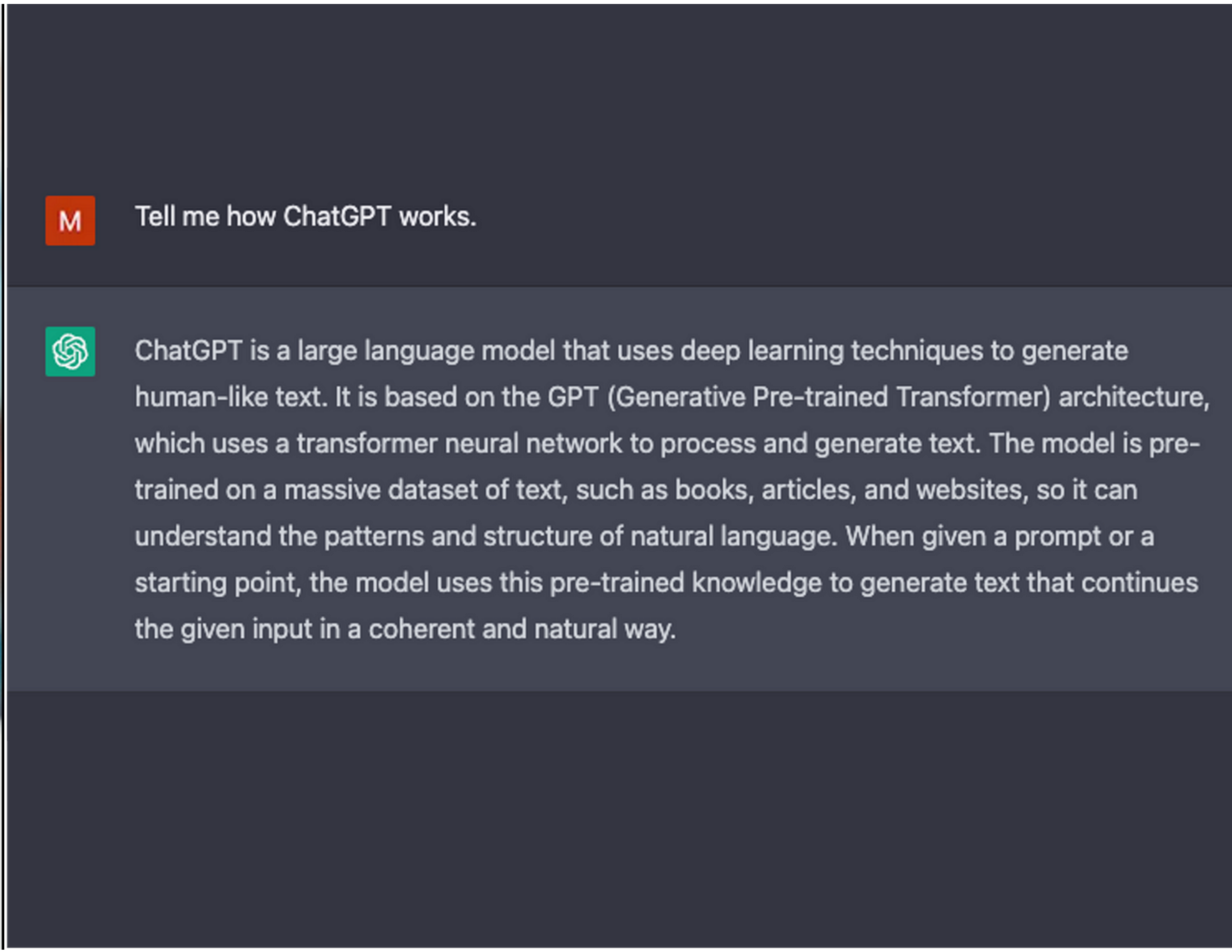
Policy Gradient

Wen Sun

Applications that use policy gradient methods





Applications that use policy gradient methods

Recap: MDPs

$$\mathcal{M} = \{\mathcal{T}, r, H, \mu, S, A\}$$

where $s_0 \sim \mu$

Objective: $J(\pi) := \mathbb{E}\left[\sum_{h=0}^{H-1} r(s_h, a_h) \,\middle|\, s_0 \sim \mu, s_{h+1} \sim \mathcal{T}(s_h, a_h), a_h \sim \pi(\cdot \mid s_h)\right]$

Recap: MDPs

$$\mathcal{M} = \{\mathcal{T}, r, H, \mu, S, A\}$$

where $s_0 \sim \mu$

Objective: $J(\pi) := \mathbb{E}\left[\sum_{h=0}^{H-1} r(s_h, a_h) \middle| s_0 \sim \mu, s_{h+1} \sim \mathcal{T}(s_h, a_h), a_h \sim \pi(\cdot \mid s_h)\right]$

LQR & MPC: find the (near) optimal policy **given the transition P**

Today

How to **learn** a good policy when transition $\mathcal{T}$ is unknown?

Outline for today

1. Recap on Gradient descent and stochastic gradient descent

2. Warm up: computing gradient using importance weighting

3. Policy Gradient formulations

Stochastic Gradient Descent

Given an objective function $J(\theta) : \mathbb{R}^d \mapsto \mathbb{R}$, (e.g., $J(\theta) = \mathbb{E}_{x,y}(f_\theta(x) - y)^2$)

SGD minimizes the above objective function as follows:

Stochastic Gradient Descent

Given an objective function $J(\theta) : \mathbb{R}^d \mapsto \mathbb{R}$, (e.g., $J(\theta) = \mathbb{E}_{x,y}(f_\theta(x) - y)^2$)

SGD minimizes the above objective function as follows:

Initialize θ_0, for t = 0, … :

Stochastic Gradient Descent

Given an objective function $J(\theta) : \mathbb{R}^d \mapsto \mathbb{R}$, (e.g., $J(\theta) = \mathbb{E}_{x,y}(f_\theta(x) - y)^2$)

SGD minimizes the above objective function as follows:

Initialize θ_0, for t = 0, … :

$$\theta_{t+1} = \theta_t - \eta g_t$$

Stochastic Gradient Descent

Given an objective function $J(\theta) : \mathbb{R}^d \mapsto \mathbb{R}$, (e.g., $J(\theta) = \mathbb{E}_{x,y}(f_\theta(x) - y)^2$)

SGD minimizes the above objective function as follows:

Initialize θ_0, for t = 0, … :

$$\theta_{t+1} = \theta_t - \eta g_t$$

where $\mathbb{E}[g_t] = \nabla_\theta J(\theta_t)$

Stochastic Gradient Descent

Given an objective function $J(\theta) : \mathbb{R}^d \mapsto \mathbb{R}$, (e.g., $J(\theta) = \mathbb{E}_{x,y}(f_\theta(x) - y)^2$)

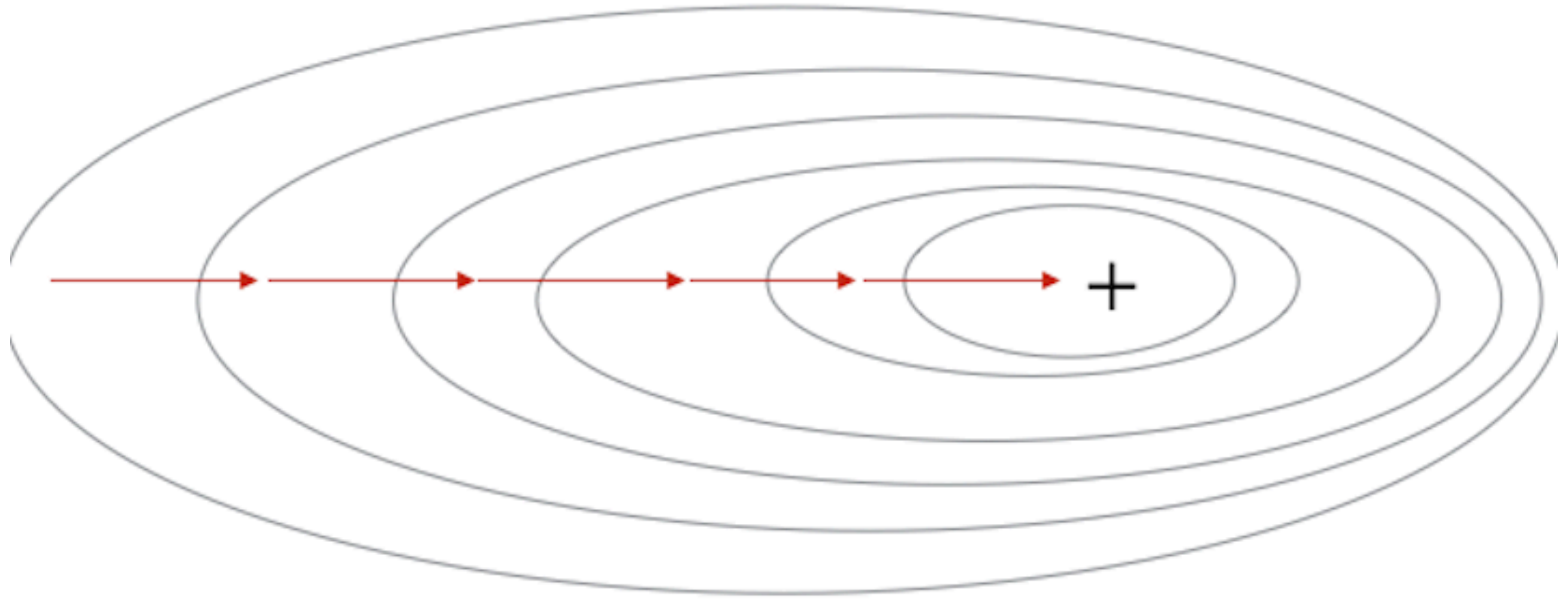
SGD minimizes the above objective function as follows:

Initialize θ_0, for t = 0, … :

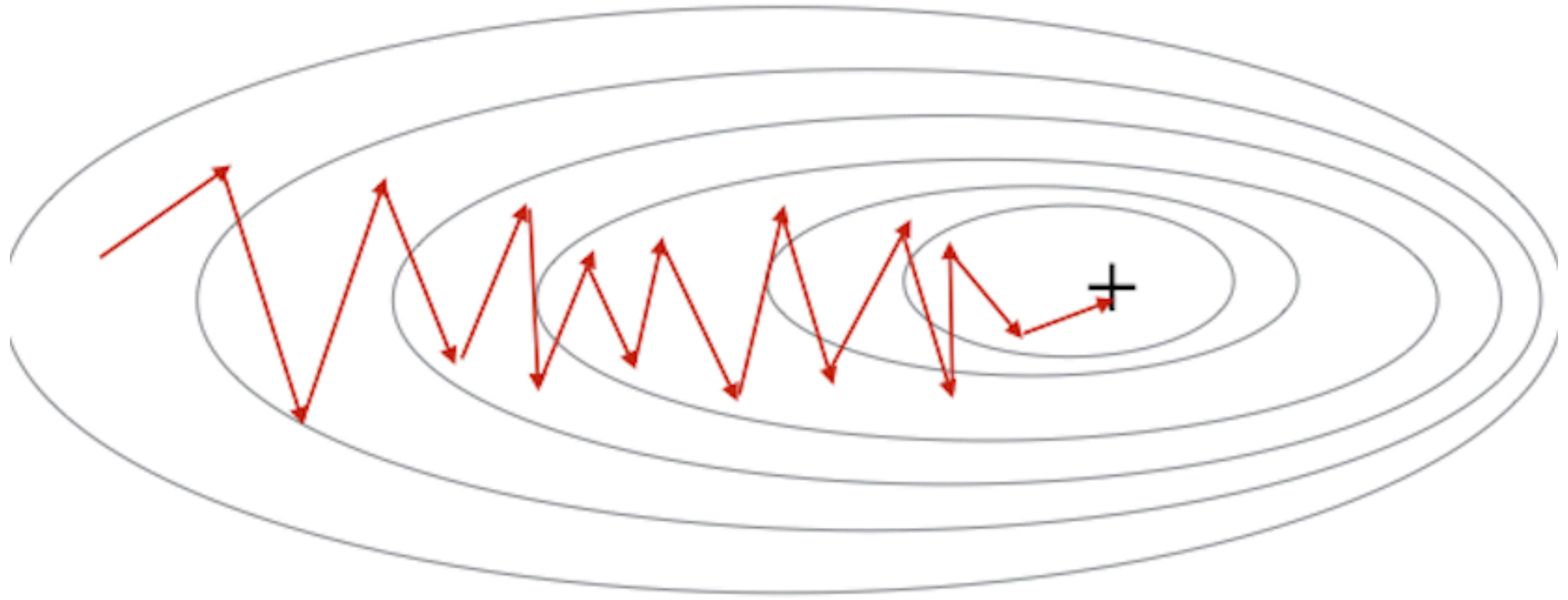
$$\theta_{t+1} = \theta_t - \eta g_t$$

where $\mathbb{E}[g_t] = \nabla_\theta J(\theta_t)$

Gradient Descent



Stochastic Gradient Descent

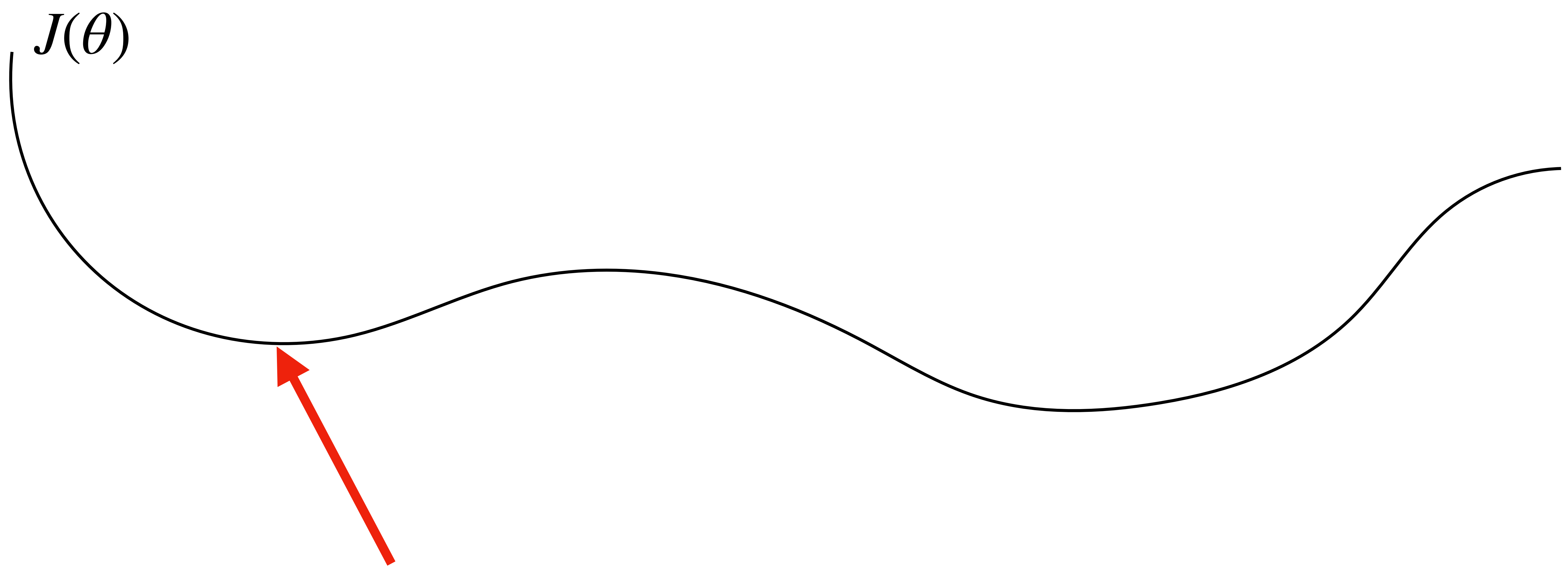


Convergence of SGD

Under some regularity condition of the objective, SGD converges to a stationary point, i.e.,

Convergence of SGD

Under some regularity condition of the objective, SGD converges to a stationary point, i.e.,

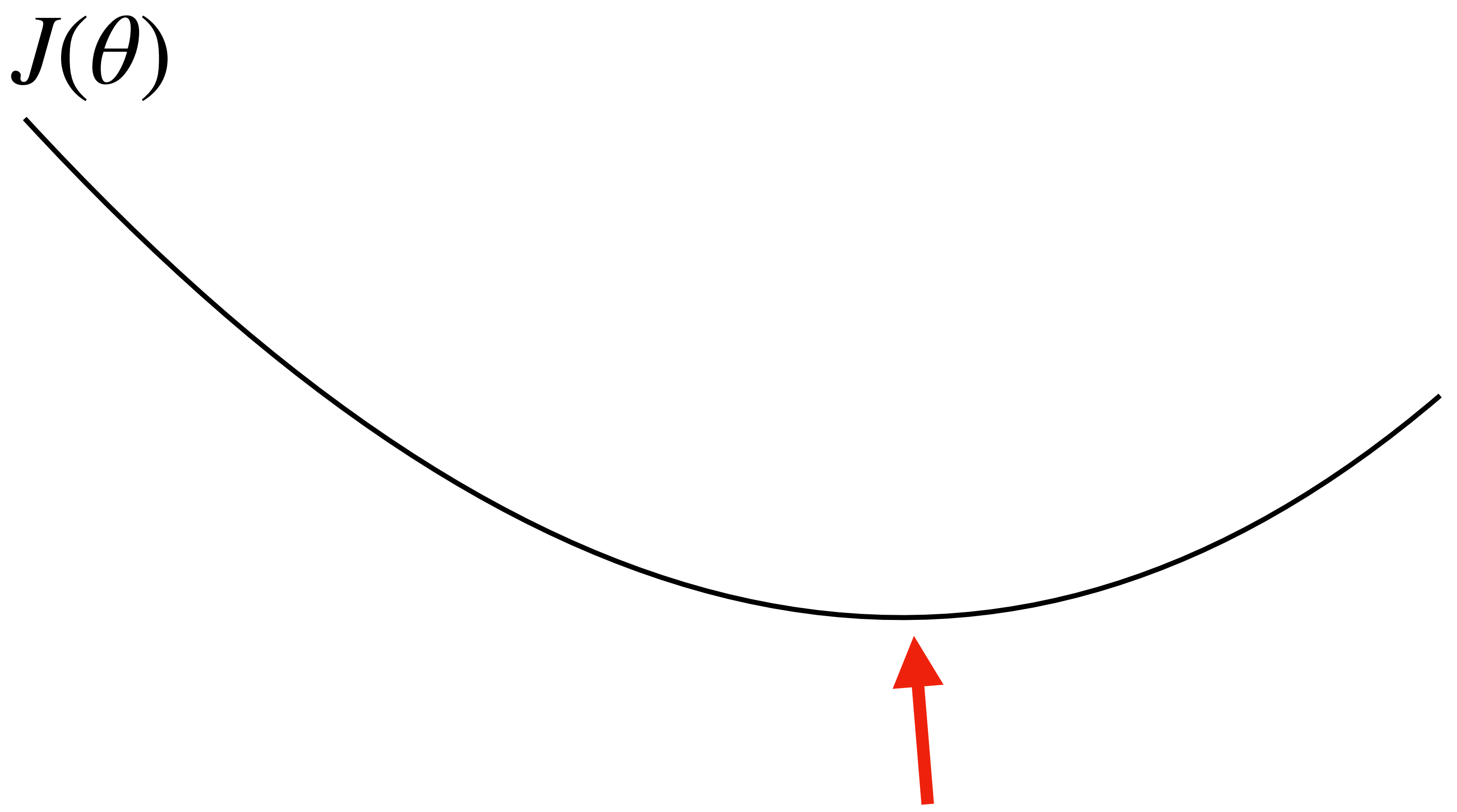


Stationary point: $||\nabla J(\theta)||_2 = 0$

Convergence of SGD

Under some regularity condition of the objective, SGD converges to a stationary point, i.e.,

For convex function, it guarantees convergence to the global optimal



Stationary point = global optimal point

SGD in general is amazing!

Works really well for training large neural networks, desipte non-convexity!

implicit regularization — models trained via SGD can generalize better

Easy to implement, take advtange of modern GPUs

Question:

Can we develop something like SGD for RL?

Outline for today

1. Recap on Gradient descent and stochastic gradient descent

2. Warm up: computing gradient using importance weighting

3. Policy Gradient formulations

Warm Up: Importance Weighting

$$J(\theta) = \mathbb{E}_{x \sim P_\theta}\left[f(x)\right]$$

Warm Up: Importance Weighting

$$J(\theta) = \mathbb{E}_{x \sim P_\theta}\left[f(x)\right]$$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x)$$

Warm Up: Importance Weighting

$$J(\theta) = \mathbb{E}_{x \sim P_\theta}\left[f(x)\right]$$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x)$$

Suppose that I have a sampling distribution ρ, s.t., $\max_x P_\theta(x)/\rho(x) < \infty$

Warm Up: Importance Weighting

$$J(\theta) = \mathbb{E}_{x \sim P_\theta}\left[f(x)\right]$$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x)$$

Suppose that I have a sampling distribution ρ, s.t., $\max_x P_\theta(x)/\rho(x) < \infty$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x) = \nabla_\theta \mathbb{E}_{x \sim \rho} \frac{P_\theta(x)}{\rho(x)} f(x)$$

Warm Up: Importance Weighting

$$J(\theta) = \mathbb{E}_{x \sim P_\theta}\left[f(x)\right]$$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x)$$

Suppose that I have a sampling distribution ρ, s.t., $\max_x P_\theta(x)/\rho(x) < \infty$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x) = \nabla_\theta \mathbb{E}_{x \sim \rho} \frac{P_\theta(x)}{\rho(x)} f(x) \quad = \mathbb{E}_{x \sim \rho} \frac{\nabla_\theta P_\theta(x)}{\rho(x)} f(x)$$

Warm Up: Importance Weighting

$$J(\theta) = \mathbb{E}_{x \sim P_\theta} \left[f(x) \right]$$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x)$$

Suppose that I have a sampling distribution ρ, s.t., $\max_x P_\theta(x) / \rho(x) < \infty$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x) = \nabla_\theta \mathbb{E}_{x \sim \rho} \frac{P_\theta(x)}{\rho(x)} f(x) = \mathbb{E}_{x \sim \rho} \frac{\nabla_\theta P_\theta(x)}{\rho(x)} f(x) \approx \frac{1}{N} \sum_{i=1}^{N} \frac{\nabla_\theta P_\theta(x_i)}{\rho(x_i)} f(x_i)$$

Warm Up: Importance Weighting

$$J(\theta) = \mathbb{E}_{x \sim P_\theta}\left[f(x)\right]$$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x)$$

Suppose that I have a sampling distribution ρ, s.t., $\max_x P_\theta(x)/\rho(x) < \infty$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x) = \nabla_\theta \mathbb{E}_{x \sim \rho} \frac{P_\theta(x)}{\rho(x)} f(x) = \mathbb{E}_{x \sim \rho} \frac{\nabla_\theta P_\theta(x)}{\rho(x)} f(x) \approx \frac{1}{N} \sum_{i=1}^{N} \frac{\nabla_\theta P_\theta(x_i)}{\rho(x_i)} f(x_i)$$

To compute gradient at θ_0: $\nabla_\theta J(\theta_0)$ (in short of $\nabla_\theta J(\theta)\,|_{\theta=\theta_0}$)

Warm Up: Importance Weighting

$$J(\theta) = \mathbb{E}_{x \sim P_\theta} \left[f(x) \right]$$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x)$$

Suppose that I have a sampling distribution ρ, s.t., $\max_x P_\theta(x)/\rho(x) < \infty$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x) = \nabla_\theta \mathbb{E}_{x \sim \rho} \frac{P_\theta(x)}{\rho(x)} f(x) \quad = \mathbb{E}_{x \sim \rho} \frac{\nabla_\theta P_\theta(x)}{\rho(x)} f(x) \quad \approx \frac{1}{N} \sum_{i=1}^{N} \frac{\nabla_\theta P_\theta(x_i)}{\rho(x_i)} f(x_i)$$

To compute gradient at θ_0: $\nabla_\theta J(\theta_0)$ (in short of $\nabla_\theta J(\theta)\,|_{\theta=\theta_0}$)

We can set sampling distribution $\rho = P_{\theta_0}$

Warm Up: Importance Weighting

$$J(\theta) = \mathbb{E}_{x \sim P_\theta}\left[f(x)\right]$$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x)$$

Suppose that I have a sampling distribution ρ, s.t., $\max_x P_\theta(x)/\rho(x) < \infty$

$$\nabla_\theta J(\theta) = \nabla_\theta \mathbb{E}_{x \sim P_\theta} f(x) = \nabla_\theta \mathbb{E}_{x \sim \rho} \frac{P_\theta(x)}{\rho(x)} f(x) \;\; = \mathbb{E}_{x \sim \rho} \frac{\nabla_\theta P_\theta(x)}{\rho(x)} f(x) \;\; \approx \frac{1}{N} \sum_{i=1}^{N} \frac{\nabla_\theta P_\theta(x_i)}{\rho(x_i)} f(x_i)$$

To compute gradient at θ_0: $\nabla_\theta J(\theta_0)$ (in short of $\nabla_\theta J(\theta)\,|_{\theta=\theta_0}$)

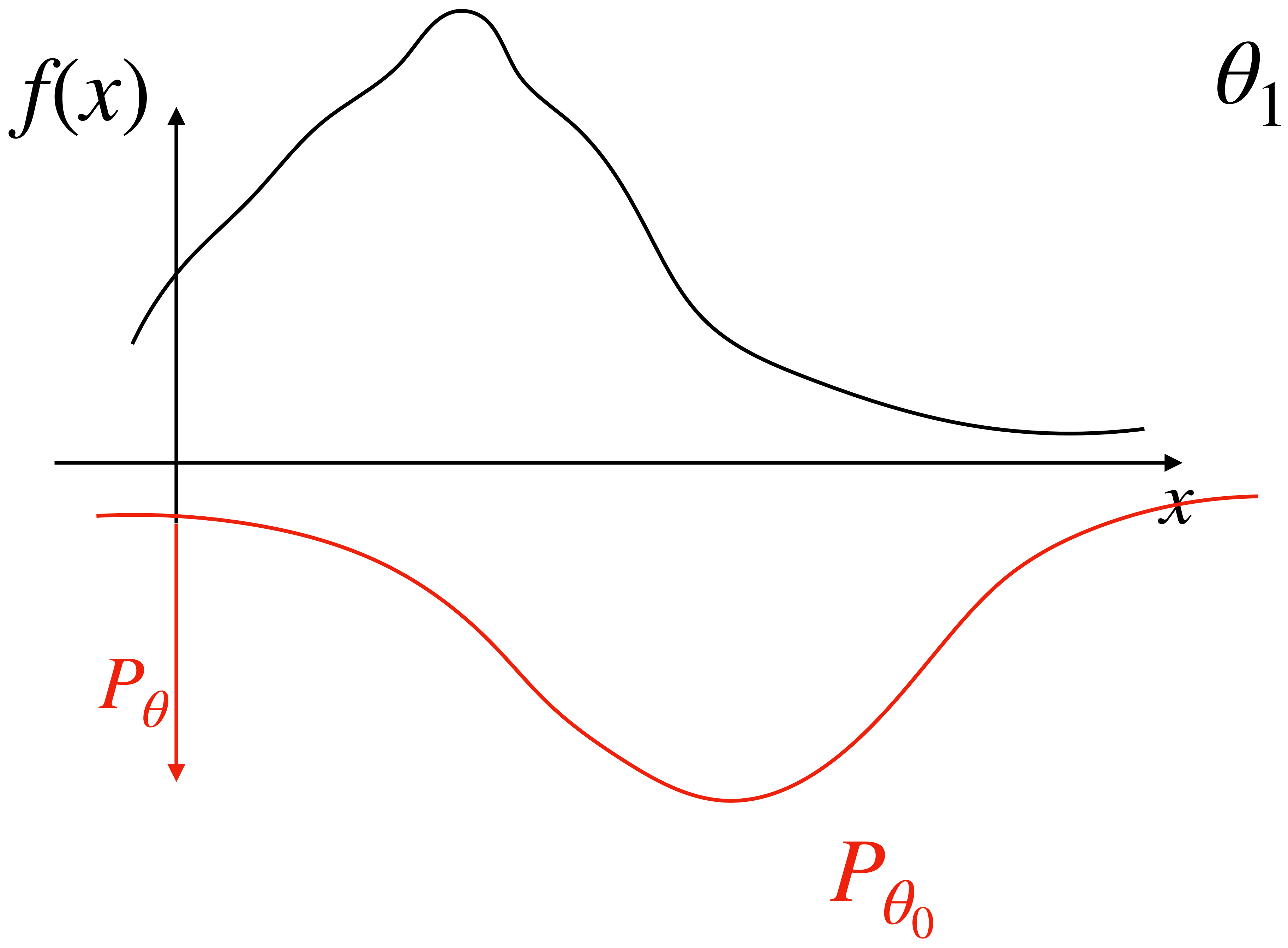
We can set sampling distribution $\rho = P_{\theta_0}$

$$\nabla_\theta J(\theta_0) = \mathbb{E}_{x \sim P_{\theta_0}}\left[\nabla_\theta \ln P_{\theta_0}(x) \cdot f(x)\right]$$

Warm Up

$$\nabla_\theta J(\theta)\,|_{\theta=\theta_0} = \mathbb{E}_{x\sim P_{\theta_0}} \nabla_\theta \ln P_{\theta_0}(x) \cdot f(x)$$

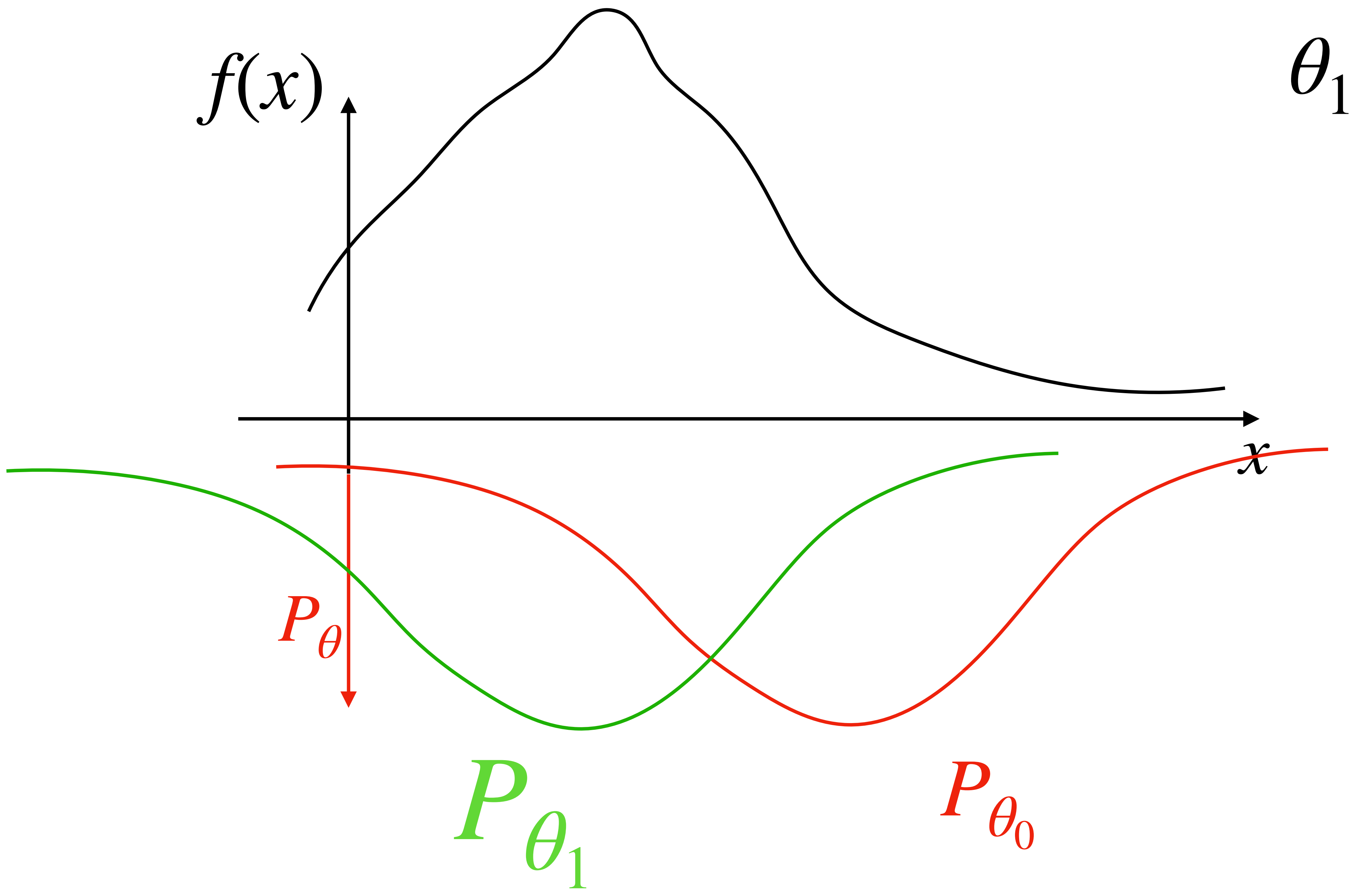
$$\theta_1 = \theta_0 + \eta \nabla_\theta J(\theta_0)$$



Warm Up

$$\nabla_\theta J(\theta)\,|_{\theta=\theta_0} = \mathbb{E}_{x\sim P_{\theta_0}} \nabla_\theta \ln P_{\theta_0}(x) \cdot f(x)$$

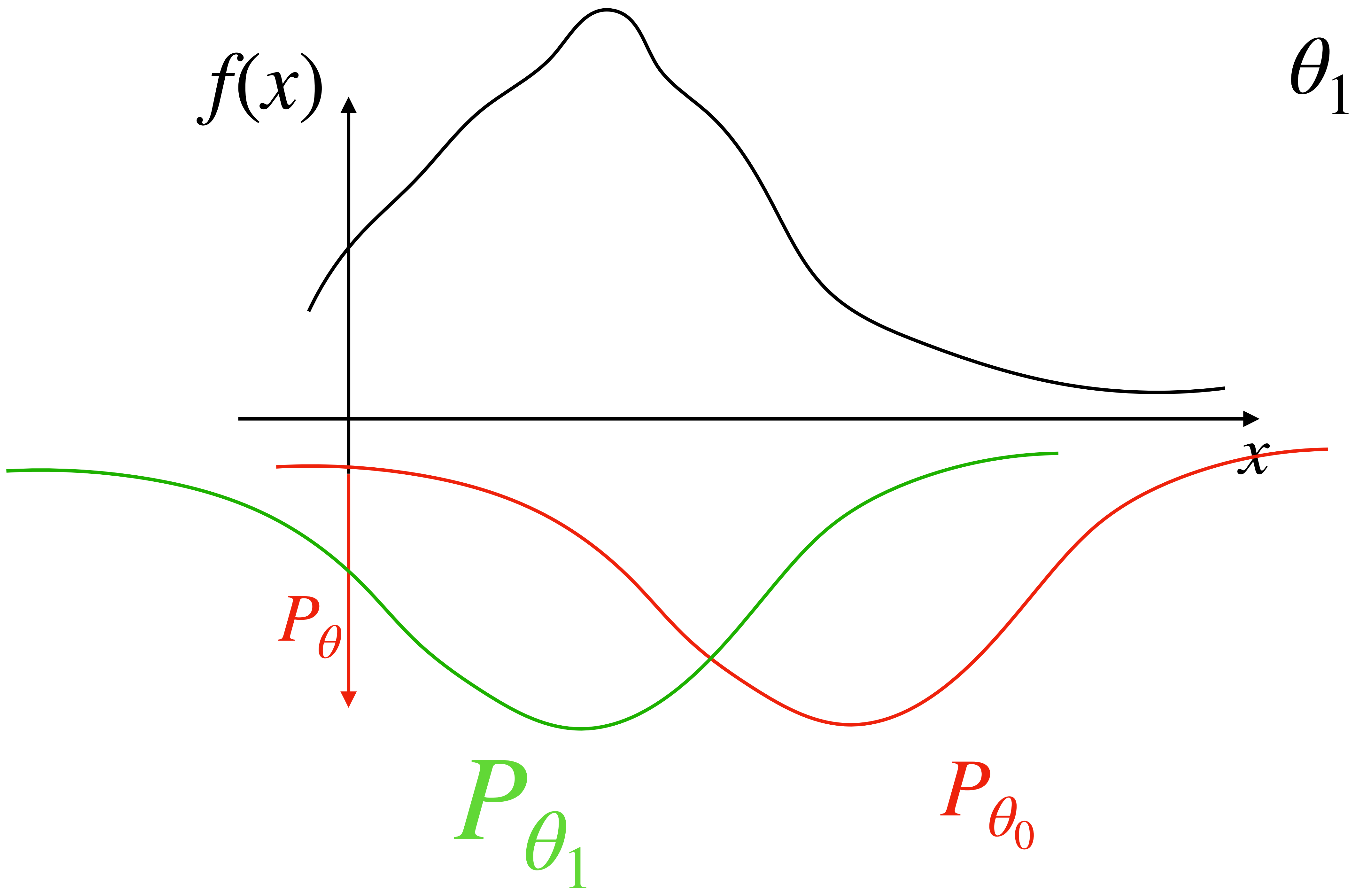
$$\theta_1 = \theta_0 + \eta \nabla_\theta J(\theta_0)$$



Warm Up

$$\nabla_\theta J(\theta)\,|_{\theta=\theta_0} = \mathbb{E}_{x\sim P_{\theta_0}} \nabla_\theta \ln P_{\theta_0}(x) \cdot f(x)$$

$$\theta_1 = \theta_0 + \eta \nabla_\theta J(\theta_0)$$

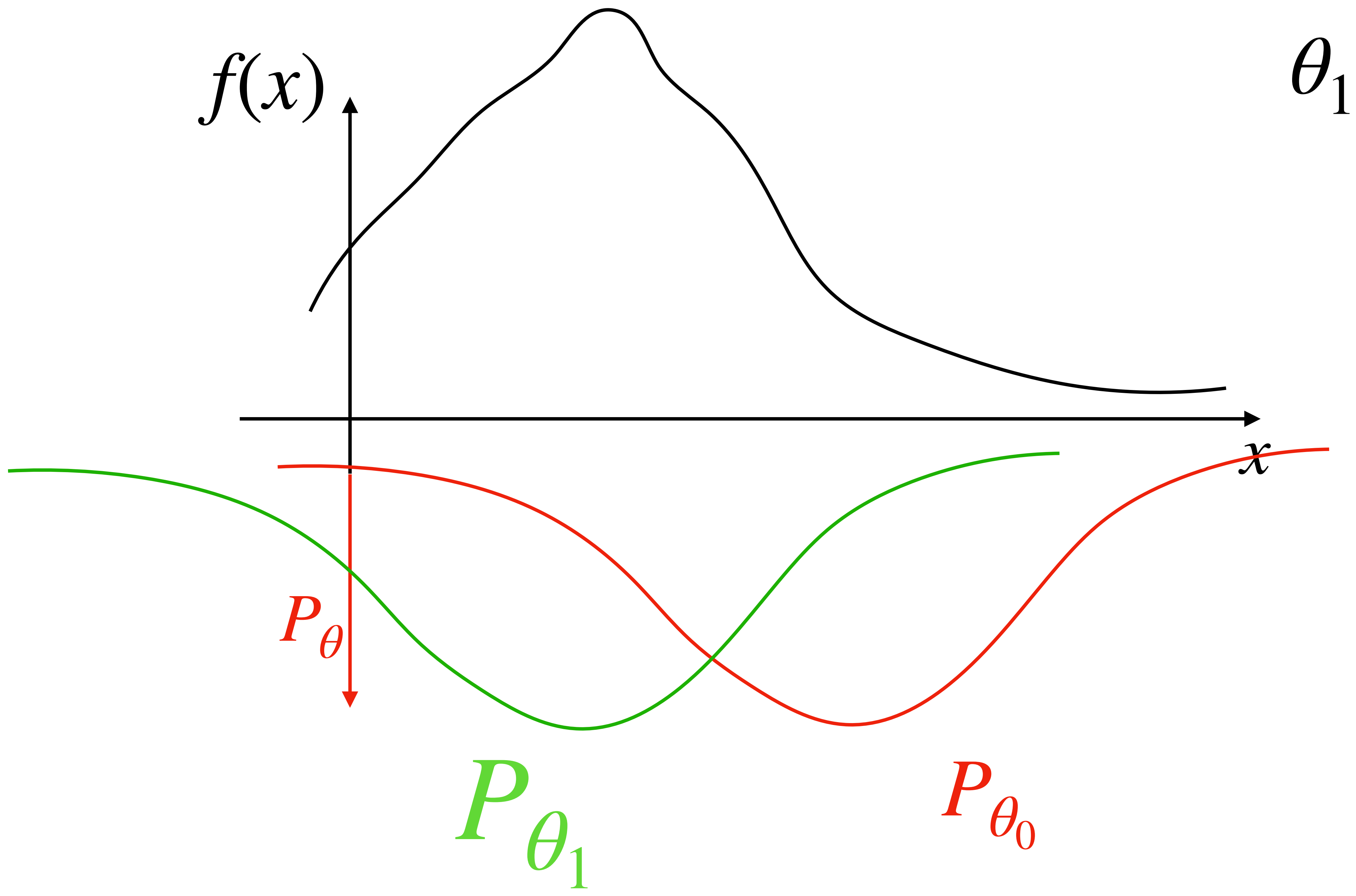


Update distribution (via updating θ) such that P_θ has high probability mass at regions where $f(x)$ is large

Warm Up

$$\nabla_\theta J(\theta)\,|_{\theta=\theta_0} = \mathbb{E}_{x\sim P_{\theta_0}} \nabla_\theta \ln P_{\theta_0}(x) \cdot f(x)$$

$$\theta_1 = \theta_0 + \eta \nabla_\theta J(\theta_0)$$



Update distribution (via updating θ) such that P_θ has high probability mass at regions where $f(x)$ is large

Using same idea, now let's move on to RL…

Outline for today

1. Recap on Gradient descent and stochastic gradient descent

2. Warm up: computing gradient using importance weighting

3. Policy Gradient formulations

Examples of Policy Parameterization

Parameterized policy $\pi_\theta(\cdot \mid s) \in \Delta(A), \forall s$

Discrete actions (e.g., LLM)

Continues actions (e.g., control, diffusion model)

Examples of Policy Parameterization

Parameterized policy $\pi_\theta(\cdot \mid s) \in \Delta(A), \forall s$

Discrete actions (e.g., LLM)

$f_\theta : S \times A \mapsto \mathbb{R}$, e.g., MLP, transformer

Continues actions (e.g., control, diffusion model)

Examples of Policy Parameterization

Parameterized policy $\pi_\theta(\cdot \mid s) \in \Delta(A), \forall s$

Discrete actions (e.g., LLM)

$f_\theta : S \times A \mapsto \mathbb{R}$, e.g., MLP, transformer

$$\pi_\theta(a \mid s) = \frac{\exp(f_\theta(s, a))}{\sum_{a'} \exp(f_\theta(s, a'))}$$

Continues actions (e.g., control, diffusion model)

Examples of Policy Parameterization

Parameterized policy $\pi_\theta(\cdot \mid s) \in \Delta(A), \forall s$

Discrete actions (e.g., LLM)

$f_\theta : S \times A \mapsto \mathbb{R}$, e.g., MLP, transformer

$$\pi_\theta(a \mid s) = \frac{\exp(f_\theta(s, a))}{\sum_{a'} \exp(f_\theta(s, a'))}$$

Continues actions (e.g., control, diffusion model)

$$\pi(\cdot \mid s) = \mathcal{N}(\mu_\theta(s), \sigma^2 I)$$

Examples of Policy Parameterization

Parameterized policy $\pi_\theta(\cdot \mid s) \in \Delta(A), \forall s$

Discrete actions (e.g., LLM)

$f_\theta : S \times A \mapsto \mathbb{R}$, e.g., MLP, transformer

$$\pi_\theta(a \mid s) = \frac{\exp(f_\theta(s, a))}{\sum_{a'} \exp(f_\theta(s, a'))}$$

Continues actions (e.g., control, diffusion model)

$$\pi(\cdot \mid s) = \mathcal{N}(\mu_\theta(s), \sigma^2 I)$$

Mean is modeled by MLP

Examples of Policy Parameterization

Parameterized policy $\pi_\theta(\cdot \mid s) \in \Delta(A), \forall s$

Discrete actions (e.g., LLM)

$f_\theta : S \times A \mapsto \mathbb{R}$, e.g., MLP, transformer

$$\pi_\theta(a \mid s) = \frac{\exp(f_\theta(s, a))}{\sum_{a'} \exp(f_\theta(s, a'))}$$

Continues actions (e.g., control, diffusion model)

$$\pi(\cdot \mid s) = \mathcal{N}(\mu_\theta(s), \sigma^2 I)$$

Mean is modeled by MLP

STD

Derivation of Policy Gradient: REINFORCE

$$\tau = \{s_0, a_0, s_1, a_1, \ldots\}$$

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)\mathcal{T}(s_1 \mid s_0, a_0)\pi_\theta(a_1 \mid s_1)\ldots\mathcal{T}(s_{H-1} \mid s_{H-2}, a_{H-2})\pi(a_{H-1} \mid s_{H-1})$$

Derivation of Policy Gradient: REINFORCE

$$\tau = \{s_0, a_0, s_1, a_1, \ldots\}$$

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \,|\, s_0)\mathcal{T}(s_1 \,|\, s_0, a_0)\pi_\theta(a_1 \,|\, s_1)\ldots\mathcal{T}(s_{H-1} \,|\, s_{H-2}, a_{H-2})\pi(a_{H-1} \,|\, s_{H-1})$$

$$J(\pi_\theta) = \mathbb{E}_{\tau\sim\rho_\theta(\tau)} \underbrace{\left[\sum_{h=0}^{H-1} r(s_h, a_h)\right]}_{R(\tau)}$$

Derivation of Policy Gradient: REINFORCE

$$\tau = \{s_0, a_0, s_1, a_1, \ldots\}$$

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)\mathcal{T}(s_1 \mid s_0, a_0)\pi_\theta(a_1 \mid s_1)\ldots\mathcal{T}(s_{H-1} \mid s_{H-2}, a_{H-2})\pi(a_{H-1} \mid s_{H-1})$$

$$J(\pi_\theta) = \mathbb{E}_{\tau\sim\rho_\theta(\tau)} \underbrace{\left[\sum_{h=0}^{H-1} r(s_h, a_h)\right]}_{R(\tau)}$$

$$\nabla_\theta J(\pi_{\theta_0}) = \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\nabla_\theta \ln \rho_{\theta_0}(\tau) R(\tau)\right]$$

Derivation of Policy Gradient: REINFORCE

$$\tau = \{s_0, a_0, s_1, a_1, \ldots\}$$

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \,|\, s_0)\mathscr{T}(s_1 \,|\, s_0, a_0)\pi_\theta(a_1 \,|\, s_1)\ldots\mathscr{T}(s_{H-1} \,|\, s_{H-2}, a_{H-2})\pi(a_{H-1} \,|\, s_{H-1})$$

$$J(\pi_\theta) = \mathbb{E}_{\tau\sim\rho_\theta(\tau)} \underbrace{\left[\sum_{h=0}^{H-1} r(s_h, a_h)\right]}_{R(\tau)}$$

$$\nabla_\theta J(\pi_{\theta_0}) = \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)} \left[\nabla_\theta \ln \rho_{\theta_0}(\tau) R(\tau)\right]$$

$$= \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)} \left[\nabla_\theta \Big(\ln \rho(s_0) + \ln \pi_{\theta_0}(a_0 \,|\, s_0) + \ln \mathscr{T}(s_1 \,|\, s_0, a_0) + \ldots \Big) R(\tau)\right]$$

Derivation of Policy Gradient: REINFORCE

$$\tau = \{s_0, a_0, s_1, a_1, \ldots\}$$

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)\mathcal{T}(s_1 \mid s_0, a_0)\pi_\theta(a_1 \mid s_1)\ldots\mathcal{T}(s_{H-1} \mid s_{H-2}, a_{H-2})\pi(a_{H-1} \mid s_{H-1})$$

$$J(\pi_\theta) = \mathbb{E}_{\tau\sim\rho_\theta(\tau)} \underbrace{\left[\sum_{h=0}^{H-1} r(s_h, a_h)\right]}_{R(\tau)}$$

$$\nabla_\theta J(\pi_{\theta_0}) = \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\nabla_\theta \ln \rho_{\theta_0}(\tau)R(\tau)\right]$$

$$= \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\nabla_\theta\Big(\ln \rho(s_0) + \ln \pi_{\theta_0}(a_0 \mid s_0) + \ln \mathcal{T}(s_1 \mid s_0, a_0) + \ldots\Big) R(\tau)\right]$$

$$= \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\nabla_\theta\Big(\ln \pi_{\theta_0}(a_0 \mid s_0) + \ln \pi_{\theta_0}(a_1 \mid s_1)\ldots\Big) R(\tau)\right]$$

Derivation of Policy Gradient: REINFORCE

$$\tau = \{s_0, a_0, s_1, a_1, \dots\}$$

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)\mathcal{T}(s_1 \mid s_0, a_0)\pi_\theta(a_1 \mid s_1)\dots\mathcal{T}(s_{H-1} \mid s_{H-2}, a_{H-2})\pi(a_{H-1} \mid s_{H-1})$$

$$J(\pi_\theta) = \mathbb{E}_{\tau\sim\rho_\theta(\tau)} \underbrace{\left[\sum_{h=0}^{H-1} r(s_h, a_h)\right]}_{R(\tau)}$$

$$\nabla_\theta J(\pi_{\theta_0}) = \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\nabla_\theta \ln \rho_{\theta_0}(\tau) R(\tau)\right]$$

$$= \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\nabla_\theta\left(\ln \rho(s_0) + \ln \pi_{\theta_0}(a_0 \mid s_0) + \ln \mathcal{T}(s_1 \mid s_0, a_0) + \dots\right) R(\tau)\right]$$

$$= \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\nabla_\theta\left(\ln \pi_{\theta_0}(a_0 \mid s_0) + \ln \pi_{\theta_0}(a_1 \mid s_1)\dots\right) R(\tau)\right] = \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_{\theta_0}(a_h \mid s_h)\right) R(\tau)\right]$$

Derivation of Policy Gradient: REINFORCE

$$\tau = \{s_0, a_0, s_1, a_1, \ldots\}$$

$$\rho_\theta(\tau) = \mu(s_0)\pi_\theta(a_0 \mid s_0)\mathcal{T}(s_1 \mid s_0, a_0)\pi_\theta(a_1 \mid s_1)\ldots\mathcal{T}(s_{H-1} \mid s_{H-2}, a_{H-2})\pi(a_{H-1} \mid s_{H-1})$$

$$J(\pi_\theta) = \mathbb{E}_{\tau\sim\rho_\theta(\tau)} \underbrace{\left[\sum_{h=0}^{H-1} r(s_h, a_h)\right]}_{R(\tau)}$$

$$\nabla_\theta J(\pi_{\theta_0}) = \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\nabla_\theta \ln \rho_{\theta_0}(\tau) R(\tau)\right]$$

Adjust policy's parameters s.t. larger reward traj has higher likelihood

$$= \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\nabla_\theta\Big(\ln \rho(s_0) + \ln \pi_{\theta_0}(a_0 \mid s_0) + \ln \mathcal{T}(s_1 \mid s_0, a_0) + \ldots\Big) R(\tau)\right]$$

$$= \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\nabla_\theta\Big(\ln \pi_{\theta_0}(a_0 \mid s_0) + \ln \pi_{\theta_0}(a_1 \mid s_1)\ldots\Big) R(\tau)\right] = \mathbb{E}_{\tau\sim\rho_{\theta_0}(\tau)}\left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_{\theta_0}(a_h \mid s_h)\right) R(\tau)\right]$$

Summary so far for Policy Gradients

We derived the most classic PG formulation:

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta(\tau)} \left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) \right) R(\tau) \right]$$

Summary so far for Policy Gradients

We derived the most classic PG formulation:

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta(\tau)} \left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) \right) R(\tau) \right]$$

Increase the likelihood of sampling an trajectory with higher total reward

Further simplification on PG

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta(\tau)} \left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) \left(\sum_{t=h}^{H-1} r(s_t, a_t) \right) \right]$$

Reward-to-go

Further simplification on PG

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta(\tau)} \left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) \left(\sum_{t=h}^{H-1} r(s_t, a_t) \right) \right]$$

Reward-to-go

(Change action distribution at h only affects rewards later on…)

Put things together — Policy Gradient Algorithm

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta(\tau)} \left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) \left(\sum_{t=h}^{H-1} r(s_t, a_t) \right) \right]$$

Initialize a policy π_{θ_0} (e.g., random initialization)

For t = 0 to T:

Put things together — Policy Gradient Algorithm

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta(\tau)} \left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) \left(\sum_{t=h}^{H-1} r(s_t, a_t) \right) \right]$$

Initialize a policy π_{θ_0} (e.g., random initialization)

For t = 0 to T:

Sample K i.i.d traj $\tau^1, \ldots, \tau^k$ from π_{θ_t}

Put things together — Policy Gradient Algorithm

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta(\tau)} \left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) \left(\sum_{t=h}^{H-1} r(s_t, a_t) \right) \right]$$

Initialize a policy π_{θ_0} (e.g., random initialization)

For t = 0 to T:

Sample K i.i.d traj $\tau^1, \ldots, \tau^k$ from π_{θ_t}

Form SG: $g_t = \sum_{i=1}^{K} \left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_{\theta_t}(a_h \mid s_h) \left(\sum_{t=h}^{H-1} r(s_t^i, a_t^i) \right) \right] / K$

Put things together — Policy Gradient Algorithm

$$\nabla J(\theta) = \mathbb{E}_{\tau \sim \rho_\theta(\tau)} \left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_\theta(a_h \mid s_h) \left(\sum_{t=h}^{H-1} r(s_t, a_t) \right) \right]$$

Initialize a policy π_{θ_0} (e.g., random initialization)

For t = 0 to T:

Sample K i.i.d traj $\tau^1, \ldots, \tau^k$ from π_{θ_t}

Form SG: $g_t = \sum_{i=1}^{K} \left[\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_{\theta_t}(a_h \mid s_h) \left(\sum_{t=h}^{H-1} r(s_t^i, a_t^i) \right) \right] / K$

SG ascent: $\theta_{t+1} = \theta_t + \eta g_t$ (or other off-shelf optimizers like AdaGrad / Adam)

Summary for today

1. Importance Weighting Trick

2. Policy Gradient:

REINFORCE (a direct application of our warm up example):

Summary for today

1. Importance Weighting Trick

2. Policy Gradient:

REINFORCE (a direct application of our warm up example):

$$\nabla J(\theta_t) = \mathbb{E}_{\tau \sim \rho_{\theta_t}(\tau)} \left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_{\theta_t}(a_h \mid s_h) \right) R(\tau) \right]$$

Summary for today

1. Importance Weighting Trick

2. Policy Gradient:

REINFORCE (a direct application of our warm up example):

$$\nabla J(\theta_t) = \mathbb{E}_{\tau \sim \rho_{\theta_t}(\tau)} \left[\left(\sum_{h=0}^{H-1} \nabla_\theta \ln \pi_{\theta_t}(a_h \mid s_h) \right) R(\tau) \right]$$

3. Known result on SGD implies Policy Gradient at least converges to stationary points