Policy Gradient Theorem

Ashwin Rao

October 4, 2018

Overview

Why do we care about Policy Gradient Theorem (PGT)?

Why do we care about Policy Gradient Theorem (PGT)?

- Let us review how we got here

Why do we care about Policy Gradient Theorem (PGT)?

- Let us review how we got here
- We started with Markov Decision Processes and Bellman Equations

Why do we care about Policy Gradient Theorem (PGT)?

- Let us review how we got here
- We started with Markov Decision Processes and Bellman Equations
- We then studied several variants of DP and RL algorithms

Why do we care about Policy Gradient Theorem (PGT)?

- Let us review how we got here
- We started with Markov Decision Processes and Bellman Equations
- We then studied several variants of DP and RL algorithms
- We noted that the idea of *Generalized Policy Iteration* (GPI) is key

Why do we care about Policy Gradient Theorem (PGT)?

- Let us review how we got here
- We started with Markov Decision Processes and Bellman Equations
- We then studied several variants of DP and RL algorithms
- We noted that the idea of *Generalized Policy Iteration* (GPI) is key
- Policy Improvement step: $\pi(a|s)$ derived from $\text{argmax}_a\, Q(s,a)$

Why do we care about Policy Gradient Theorem (PGT)?

- Let us review how we got here
- We started with Markov Decision Processes and Bellman Equations
- We then studied several variants of DP and RL algorithms
- We noted that the idea of *Generalized Policy Iteration* (GPI) is key
- Policy Improvement step: $\pi(a|s)$ derived from $\text{argmax}_a\, Q(s,a)$
- How do we do argmax when action space is large or continuous?

Why do we care about Policy Gradient Theorem (PGT)?

- Let us review how we got here
- We started with Markov Decision Processes and Bellman Equations
- We then studied several variants of DP and RL algorithms
- We noted that the idea of *Generalized Policy Iteration* (GPI) is key
- Policy Improvement step: $\pi(a|s)$ derived from $\text{argmax}_a \, Q(s, a)$
- How do we do argmax when action space is large or continuous?
- Idea: Do Policy Improvement step with a Gradient Ascent instead

"Policy Improvement with a Gradient Ascent??"

"Policy Improvement with a Gradient Ascent??"

- We want to find the Policy that fetches the "Best Expected Returns"

"Policy Improvement with a Gradient Ascent??"

- We want to find the Policy that fetches the "Best Expected Returns"
- Gradient Ascent on "Expected Returns" w.r.t params of Policy func

"Policy Improvement with a Gradient Ascent??"

- We want to find the Policy that fetches the "Best Expected Returns"
- Gradient Ascent on "Expected Returns" w.r.t params of Policy func
- So we need a func approx for (stochastic) Policy Func: $\pi(s, a; \theta)$

"Policy Improvement with a Gradient Ascent??"

- We want to find the Policy that fetches the "Best Expected Returns"
- Gradient Ascent on "Expected Returns" w.r.t params of Policy func
- So we need a func approx for (stochastic) Policy Func: $\pi(s, a; \theta)$
- In addition to the usual func approx for Action Value Func: $Q(s, a; w)$

"Policy Improvement with a Gradient Ascent??"

- We want to find the Policy that fetches the "Best Expected Returns"
- Gradient Ascent on "Expected Returns" w.r.t params of Policy func
- So we need a func approx for (stochastic) Policy Func: $\pi(s, a; \theta)$
- In addition to the usual func approx for Action Value Func: $Q(s, a; w)$
- $\pi(s, a; \theta)$ func approx called *Actor*, $Q(s, a; w)$ func approx called *Critic*

"Policy Improvement with a Gradient Ascent??"

- We want to find the Policy that fetches the "Best Expected Returns"
- Gradient Ascent on "Expected Returns" w.r.t params of Policy func
- So we need a func approx for (stochastic) Policy Func: $\pi(s, a; \theta)$
- In addition to the usual func approx for Action Value Func: $Q(s, a; w)$
- $\pi(s, a; \theta)$ func approx called *Actor*, $Q(s, a; w)$ func approx called *Critic*
- Critic parameters w are optimized w.r.t $Q(s, a; w)$ loss function min

"Policy Improvement with a Gradient Ascent??"

- We want to find the Policy that fetches the "Best Expected Returns"
- Gradient Ascent on "Expected Returns" w.r.t params of Policy func
- So we need a func approx for (stochastic) Policy Func: $\pi(s, a; \theta)$
- In addition to the usual func approx for Action Value Func: $Q(s, a; w)$
- $\pi(s, a; \theta)$ func approx called *Actor*, $Q(s, a; w)$ func approx called *Critic*
- Critic parameters w are optimized w.r.t $Q(s, a; w)$ loss function min
- Actor parameters θ are optimized w.r.t Expected Returns max

"Policy Improvement with a Gradient Ascent??"

- We want to find the Policy that fetches the "Best Expected Returns"
- Gradient Ascent on "Expected Returns" w.r.t params of Policy func
- So we need a func approx for (stochastic) Policy Func: $\pi(s, a; \theta)$
- In addition to the usual func approx for Action Value Func: $Q(s, a; w)$
- $\pi(s, a; \theta)$ func approx called *Actor*, $Q(s, a; w)$ func approx called *Critic*
- Critic parameters w are optimized w.r.t $Q(s, a; w)$ loss function min
- Actor parameters θ are optimized w.r.t Expected Returns max
- We need to formally define "Expected Returns"

“Policy Improvement with a Gradient Ascent??”

- We want to find the Policy that fetches the “Best Expected Returns”
- Gradient Ascent on “Expected Returns” w.r.t params of Policy func
- So we need a func approx for (stochastic) Policy Func: $\pi(s, a; \theta)$
- In addition to the usual func approx for Action Value Func: $Q(s, a; w)$
- $\pi(s, a; \theta)$ func approx called *Actor*, $Q(s, a; w)$ func approx called *Critic*
- Critic parameters w are optimized w.r.t $Q(s, a; w)$ loss function min
- Actor parameters θ are optimized w.r.t Expected Returns max
- We need to formally define “Expected Returns”
- But we already see that this idea is appealing for continuous actions

"Policy Improvement with a Gradient Ascent??"

- We want to find the Policy that fetches the "Best Expected Returns"
- Gradient Ascent on "Expected Returns" w.r.t params of Policy func
- So we need a func approx for (stochastic) Policy Func: $\pi(s, a; \theta)$
- In addition to the usual func approx for Action Value Func: $Q(s, a; w)$
- $\pi(s, a; \theta)$ func approx called *Actor*, $Q(s, a; w)$ func approx called *Critic*
- Critic parameters w are optimized w.r.t $Q(s, a; w)$ loss function min
- Actor parameters θ are optimized w.r.t Expected Returns max
- We need to formally define "Expected Returns"
- But we already see that this idea is appealing for continuous actions
- GPI with Policy Improvement done as **Policy Gradient (Ascent)**

Other Advantages of Policy Gradient approach

Other Advantages of Policy Gradient approach

- Finds the best *stochastic* policy

Other Advantages of Policy Gradient approach

- Finds the best *stochastic* policy
- Unlike the deterministic policy-focused search of other RL algorithms

Other Advantages of Policy Gradient approach

- Finds the best *stochastic* policy
- Unlike the deterministic policy-focused search of other RL algorithms
- Naturally *explores* due to stochastic policy representation

Other Advantages of Policy Gradient approach

- Finds the best *stochastic* policy
- Unlike the deterministic policy-focused search of other RL algorithms
- Naturally *explores* due to stochastic policy representation
- Small changes in $\theta \Rightarrow$ small changes in π, and in state distribution

Other Advantages of Policy Gradient approach

- Finds the best *stochastic* policy
- Unlike the deterministic policy-focused search of other RL algorithms
- Naturally *explores* due to stochastic policy representation
- Small changes in $\theta \Rightarrow$ small changes in π, and in state distribution
- This avoids the convergence issues seen in argmax-based algorithms

Notation

Notation

- Discount Factor γ

PGT coverage will be quite similar for non-episodic, by considering average-reward objective (so we won't cover it)

Notation

- Discount Factor γ
- Assume episodic with $0 \leq \gamma \leq 1$ or non-episodic with $0 \leq \gamma < 1$

PGT coverage will be quite similar for non-episodic, by considering average-reward objective (so we won't cover it)

Notation

- Discount Factor γ
- Assume episodic with $0 \leq \gamma \leq 1$ or non-episodic with $0 \leq \gamma < 1$
- States $s_t \in \mathcal{S}$, Actions $a_t \in \mathcal{A}$, Rewards $r_t \in \mathbb{R}$, $\forall t \in \{0, 1, 2, \ldots\}$

PGT coverage will be quite similar for non-episodic, by considering average-reward objective (so we won't cover it)

Notation

- Discount Factor γ
- Assume episodic with $0 \leq \gamma \leq 1$ or non-episodic with $0 \leq \gamma < 1$
- States $s_t \in \mathcal{S}$, Actions $a_t \in \mathcal{A}$, Rewards $r_t \in \mathbb{R}$, $\forall t \in \{0, 1, 2, \ldots\}$
- State Transition Probabilities $\mathcal{P}_{s,s'}^a = Pr(s_{t+1} = s' | s_t = s, a_t = a)$

PGT coverage will be quite similar for non-episodic, by considering average-reward objective (so we won't cover it)

Notation

- Discount Factor γ
- Assume episodic with $0 \leq \gamma \leq 1$ or non-episodic with $0 \leq \gamma < 1$
- States $s_t \in \mathcal{S}$, Actions $a_t \in \mathcal{A}$, Rewards $r_t \in \mathbb{R}$, $\forall t \in \{0, 1, 2, \dots\}$
- State Transition Probabilities $\mathcal{P}_{s,s'}^a = Pr(s_{t+1} = s' | s_t = s, a_t = a)$
- Expected Rewards $\mathcal{R}_s^a = E[r_t | s_t = s, a_t = a]$

PGT coverage will be quite similar for non-episodic, by considering average-reward objective (so we won't cover it)

Notation

- Discount Factor γ
- Assume episodic with $0 \leq \gamma \leq 1$ or non-episodic with $0 \leq \gamma < 1$
- States $s_t \in \mathcal{S}$, Actions $a_t \in \mathcal{A}$, Rewards $r_t \in \mathbb{R}$, $\forall t \in \{0, 1, 2, \ldots\}$
- State Transition Probabilities $\mathcal{P}_{s,s'}^a = Pr(s_{t+1} = s' | s_t = s, a_t = a)$
- Expected Rewards $\mathcal{R}_s^a = E[r_t | s_t = s, a_t = a]$
- Initial State Probability Distribution $p_0 : \mathcal{S} \rightarrow [0, 1]$

PGT coverage will be quite similar for non-episodic, by considering average-reward objective (so we won't cover it)

Notation

- Discount Factor γ
- Assume episodic with $0 \leq \gamma \leq 1$ or non-episodic with $0 \leq \gamma < 1$
- States $s_t \in \mathcal{S}$, Actions $a_t \in \mathcal{A}$, Rewards $r_t \in \mathbb{R}$, $\forall t \in \{0, 1, 2, \ldots\}$
- State Transition Probabilities $\mathcal{P}_{s,s'}^a = Pr(s_{t+1} = s' | s_t = s, a_t = a)$
- Expected Rewards $\mathcal{R}_s^a = E[r_t | s_t = s, a_t = a]$
- Initial State Probability Distribution $p_0 : \mathcal{S} \rightarrow [0, 1]$
- Policy Func Approx $\pi(s, a; \theta) = Pr(a_t = a | s_t = s, \theta), \theta \in \mathbb{R}^k$

PGT coverage will be quite similar for non-episodic, by considering average-reward objective (so we won't cover it)

“Expected Returns” Objective

“Expected Returns” Objective

Now we formalize the “Expected Returns” Objective $J(\pi_\theta)$

$$J(\pi_\theta) = E[\sum_{t=0}^{\infty} \gamma^t r_t | \pi]$$

"Expected Returns" Objective

Now we formalize the "Expected Returns" Objective $J(\pi_\theta)$

$$J(\pi_\theta) = E[\sum_{t=0}^{\infty} \gamma^t r_t | \pi]$$

Value Function $V^\pi(s)$ and Action Value function $Q^\pi(s, a)$ defined as:

$$V^\pi(s) = E[\sum_{t=k}^{\infty} \gamma^{t-k} r_t | s_k = s, \pi], \forall k \in \{0, 1, 2, \ldots\}$$

$$Q^\pi(s, a) = E[\sum_{t=k}^{\infty} \gamma^{t-k} r_t | s_k = s, a_k = a, \pi], \forall k \in \{0, 1, 2, \ldots\}$$

"Expected Returns" Objective

Now we formalize the "Expected Returns" Objective $J(\pi_\theta)$

$$J(\pi_\theta) = E[\sum_{t=0}^{\infty} \gamma^t r_t | \pi]$$

Value Function $V^\pi(s)$ and Action Value function $Q^\pi(s,a)$ defined as:

$$V^\pi(s) = E[\sum_{t=k}^{\infty} \gamma^{t-k} r_t | s_k = s, \pi], \forall k \in \{0, 1, 2, \ldots\}$$

$$Q^\pi(s,a) = E[\sum_{t=k}^{\infty} \gamma^{t-k} r_t | s_k = s, a_k = a, \pi], \forall k \in \{0, 1, 2, \ldots\}$$

Advantage Function $A^\pi(s,a) = Q^\pi(s,a) - V^\pi(s)$

"Expected Returns" Objective

Now we formalize the "Expected Returns" Objective $J(\pi_\theta)$

$$J(\pi_\theta) = E[\sum_{t=0}^{\infty} \gamma^t r_t | \pi]$$

Value Function $V^\pi(s)$ and Action Value function $Q^\pi(s,a)$ defined as:

$$V^\pi(s) = E[\sum_{t=k}^{\infty} \gamma^{t-k} r_t | s_k = s, \pi], \forall k \in \{0, 1, 2, \ldots\}$$

$$Q^\pi(s,a) = E[\sum_{t=k}^{\infty} \gamma^{t-k} r_t | s_k = s, a_k = a, \pi], \forall k \in \{0, 1, 2, \ldots\}$$

Advantage Function $A^\pi(s,a) = Q^\pi(s,a) - V^\pi(s)$

Also, $p(s \rightarrow s', t, \pi)$ will be a key function for us - it denotes the probability of going from state s to s' in t steps by following policy π

Discounted State Visitation Measure

Discounted State Visitation Measure

$$J(\pi_\theta) = E[\sum_{t=0}^{\infty} \gamma^t r_t | \pi] = \sum_{t=0}^{\infty} \gamma^t E[r_t | \pi]$$

Discounted State Visitation Measure

$$J(\pi_\theta) = E[\sum_{t=0}^{\infty} \gamma^t r_t | \pi] = \sum_{t=0}^{\infty} \gamma^t E[r_t | \pi]$$

$$= \sum_{t=0}^{\infty} \gamma^t \int_{\mathcal{S}} (\int_{\mathcal{S}} p_0(s_0) \cdot p(s_0 \rightarrow s, t, \pi) \cdot ds_0) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot \mathcal{R}_s^a \cdot da \cdot ds$$

Discounted State Visitation Measure

$$J(\pi_\theta) = E[\sum_{t=0}^{\infty} \gamma^t r_t | \pi] = \sum_{t=0}^{\infty} \gamma^t E[r_t | \pi]$$

$$= \sum_{t=0}^{\infty} \gamma^t \int_{\mathcal{S}} (\int_{\mathcal{S}} p_0(s_0) \cdot p(s_0 \rightarrow s, t, \pi) \cdot ds_0) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot \mathcal{R}_s^a \cdot da \cdot ds$$

$$= \int_{\mathcal{S}} (\int_{\mathcal{S}} \sum_{t=0}^{\infty} \gamma^t \cdot p_0(s_0) \cdot p(s_0 \rightarrow s, t, \pi) \cdot ds_0) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot \mathcal{R}_s^a \cdot da \cdot ds$$

Discounted State Visitation Measure

$$J(\pi_\theta) = E[\sum_{t=0}^{\infty} \gamma^t r_t | \pi] = \sum_{t=0}^{\infty} \gamma^t E[r_t | \pi]$$

$$= \sum_{t=0}^{\infty} \gamma^t \int_{\mathcal{S}} (\int_{\mathcal{S}} p_0(s_0) \cdot p(s_0 \to s, t, \pi) \cdot ds_0) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot \mathcal{R}_s^a \cdot da \cdot ds$$

$$= \int_{\mathcal{S}} (\int_{\mathcal{S}} \sum_{t=0}^{\infty} \gamma^t \cdot p_0(s_0) \cdot p(s_0 \to s, t, \pi) \cdot ds_0) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot \mathcal{R}_s^a \cdot da \cdot ds$$

Definition

$$J(\pi_\theta) = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot \mathcal{R}_s^a \cdot da \cdot ds$$

Discounted State Visitation Measure

$$J(\pi_\theta) = E[\sum_{t=0}^{\infty} \gamma^t r_t | \pi] = \sum_{t=0}^{\infty} \gamma^t E[r_t | \pi]$$

$$= \sum_{t=0}^{\infty} \gamma^t \int_{\mathcal{S}} (\int_{\mathcal{S}} p_0(s_0) \cdot p(s_0 \rightarrow s, t, \pi) \cdot ds_0) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot \mathcal{R}_s^a \cdot da \cdot ds$$

$$= \int_{\mathcal{S}} (\int_{\mathcal{S}} \sum_{t=0}^{\infty} \gamma^t \cdot p_0(s_0) \cdot p(s_0 \rightarrow s, t, \pi) \cdot ds_0) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot \mathcal{R}_s^a \cdot da \cdot ds$$

Definition

$$J(\pi_\theta) = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot \mathcal{R}_s^a \cdot da \cdot ds$$

where $\rho^\pi(s) = \int_{\mathcal{S}} \sum_{t=0}^{\infty} \gamma^t \cdot p_0(s_0) \cdot p(s_0 \rightarrow s, t, \pi) \cdot ds_0$ is the key function (for PGT) that we refer to as the *Discounted State Visitation Measure*.

Policy Gradient Theorem

Policy Gradient Theorem

Theorem

$$\frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \frac{\partial \pi(s, a; \theta)}{\partial \theta} Q^\pi(s, a) \cdot da \cdot ds$$

Policy Gradient Theorem

Theorem

$$\frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \frac{\partial \pi(s,a;\theta)}{\partial \theta} Q^\pi(s,a) \cdot da \cdot ds$$

- Note: $\rho^\pi(s)$ depends on θ, but we don't have a $\frac{\partial \rho^\pi(s)}{\partial \theta}$ term in $\frac{\partial J(\pi_\theta)}{\partial \theta}$

Policy Gradient Theorem

Theorem

$$\frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \frac{\partial \pi(s, a; \theta)}{\partial \theta} Q^\pi(s, a) \cdot da \cdot ds$$

- Note: $\rho^\pi(s)$ depends on θ, but we don't have a $\frac{\partial \rho^\pi(s)}{\partial \theta}$ term in $\frac{\partial J(\pi_\theta)}{\partial \theta}$
- So we can simply sample simulation paths, and at each time step, we calculate $\frac{\partial \log \pi(s,a;\theta)}{\partial \theta} Q^\pi(s, a)$ (probabilities implicit in the paths)

Policy Gradient Theorem

Theorem

$$\frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \frac{\partial \pi(s, a; \theta)}{\partial \theta} Q^\pi(s, a) \cdot da \cdot ds$$

- Note: $\rho^\pi(s)$ depends on θ, but we don't have a $\frac{\partial \rho^\pi(s)}{\partial \theta}$ term in $\frac{\partial J(\pi_\theta)}{\partial \theta}$
- So we can simply sample simulation paths, and at each time step, we calculate $\frac{\partial \log \pi(s,a;\theta)}{\partial \theta} Q^\pi(s, a)$ (probabilities implicit in the paths)
- We will estimate $Q^\pi(s, a)$ with a func approx $Q(s, a; w)$

Policy Gradient Theorem

Theorem

$$\frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \frac{\partial \pi(s, a; \theta)}{\partial \theta} Q^\pi(s, a) \cdot da \cdot ds$$

- Note: $\rho^\pi(s)$ depends on θ, but we don't have a $\frac{\partial \rho^\pi(s)}{\partial \theta}$ term in $\frac{\partial J(\pi_\theta)}{\partial \theta}$
- So we can simply sample simulation paths, and at each time step, we calculate $\frac{\partial \log \pi(s,a;\theta)}{\partial \theta} Q^\pi(s, a)$ (probabilities implicit in the paths)
- We will estimate $Q^\pi(s, a)$ with a func approx $Q(s, a; w)$
- We will later show how to avoid the estimate bias of $Q(s, a; w)$

Policy Gradient Theorem

Theorem

$$\frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \frac{\partial \pi(s,a;\theta)}{\partial \theta} Q^\pi(s,a) \cdot da \cdot ds$$

- Note: $\rho^\pi(s)$ depends on θ, but we don't have a $\frac{\partial \rho^\pi(s)}{\partial \theta}$ term in $\frac{\partial J(\pi_\theta)}{\partial \theta}$
- So we can simply sample simulation paths, and at each time step, we calculate $\frac{\partial \log \pi(s,a;\theta)}{\partial \theta} Q^\pi(s,a)$ (probabilities implicit in the paths)
- We will estimate $Q^\pi(s,a)$ with a func approx $Q(s,a;w)$
- We will later show how to avoid the estimate bias of $Q(s,a;w)$
- This numerical estimate of $\frac{\partial J(\pi_\theta)}{\partial \theta}$ enables **Policy Gradient Ascent**

Policy Gradient Theorem

Theorem

$$\frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \frac{\partial \pi(s,a;\theta)}{\partial \theta} Q^\pi(s,a) \cdot da \cdot ds$$

- Note: $\rho^\pi(s)$ depends on θ, but we don't have a $\frac{\partial \rho^\pi(s)}{\partial \theta}$ term in $\frac{\partial J(\pi_\theta)}{\partial \theta}$
- So we can simply sample simulation paths, and at each time step, we calculate $\frac{\partial \log \pi(s,a;\theta)}{\partial \theta} Q^\pi(s,a)$ (probabilities implicit in the paths)
- We will estimate $Q^\pi(s,a)$ with a func approx $Q(s,a;w)$
- We will later show how to avoid the estimate bias of $Q(s,a;w)$
- This numerical estimate of $\frac{\partial J(\pi_\theta)}{\partial \theta}$ enables **Policy Gradient Ascent**
- We will now go through the PGT proof slowly and rigorously

Policy Gradient Theorem

Theorem

$$\frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \frac{\partial \pi(s, a; \theta)}{\partial \theta} Q^\pi(s, a) \cdot da \cdot ds$$

- Note: $\rho^\pi(s)$ depends on θ, but we don't have a $\frac{\partial \rho^\pi(s)}{\partial \theta}$ term in $\frac{\partial J(\pi_\theta)}{\partial \theta}$
- So we can simply sample simulation paths, and at each time step, we calculate $\frac{\partial \log \pi(s,a;\theta)}{\partial \theta} Q^\pi(s, a)$ (probabilities implicit in the paths)
- We will estimate $Q^\pi(s, a)$ with a func approx $Q(s, a; w)$
- We will later show how to avoid the estimate bias of $Q(s, a; w)$
- This numerical estimate of $\frac{\partial J(\pi_\theta)}{\partial \theta}$ enables **Policy Gradient Ascent**
- We will now go through the PGT proof slowly and rigorously
- Providing commentary and intuition before each step in the proof

Proof of Policy Gradient Theorem

Proof of Policy Gradient Theorem

We begin the proof by noting that:

$$J(\pi_\theta) = \int_{\mathcal{S}} p_0(s_0) \cdot V^\pi(s_0) \cdot ds_0 = \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \pi(s_0, a_0; \theta) \cdot Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0$$

Proof of Policy Gradient Theorem

We begin the proof by noting that:

$$J(\pi_\theta) = \int_{\mathcal{S}} p_0(s_0) \cdot V^\pi(s_0) \cdot ds_0 = \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \pi(s_0, a_0; \theta) \cdot Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0$$

Spilt $\frac{\partial J(\pi_\theta)}{\partial \theta}$ by partial of $\pi(s_0, a_0; \theta)$ and partial of $Q^\pi(s_0, a_0)$

Proof of Policy Gradient Theorem

We begin the proof by noting that:

$$J(\pi_\theta) = \int_{\mathcal{S}} p_0(s_0) \cdot V^\pi(s_0) \cdot ds_0 = \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \pi(s_0, a_0; \theta) \cdot Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0$$

Spilt $\frac{\partial J(\pi_\theta)}{\partial \theta}$ by partial of $\pi(s_0, a_0; \theta)$ and partial of $Q^\pi(s_0, a_0)$

$$\begin{aligned}\frac{\partial J(\pi_\theta)}{\partial \theta} &= \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0 \\ &+ \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \pi(s_0, a_0; \theta) \frac{\partial Q^\pi(s_0, a_0)}{\partial \theta} \cdot da_0 \cdot ds_0\end{aligned}$$

Proof of Policy Gradient Theorem

Proof of Policy Gradient Theorem

Now expand $Q^\pi(s_0, a_0)$ to $\mathcal{R}_{s_0}^{a_0} + \int_{\mathcal{S}} \gamma \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot V^\pi(s_1) \cdot ds_1$ (Bellman)

Proof of Policy Gradient Theorem

Now expand $Q^\pi(s_0, a_0)$ to $\mathcal{R}_{s_0}^{a_0} + \int_\mathcal{S} \gamma \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot V^\pi(s_1) \cdot ds_1$ (Bellman)

$$= \int_\mathcal{S} p_0(s_0) \int_\mathcal{A} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0$$
$$+ \int_\mathcal{S} p_0(s_0) \int_\mathcal{A} \pi(s_0, a_0; \theta) \cdot \frac{\partial}{\partial \theta}(\mathcal{R}_{s_0}^{a_0} + \int_\mathcal{S} \gamma \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot V^\pi(s_1) \cdot ds_1) \cdot da_0 \cdot ds_0$$

Proof of Policy Gradient Theorem

Now expand $Q^\pi(s_0, a_0)$ to $\mathcal{R}_{s_0}^{a_0} + \int_\mathcal{S} \gamma \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot V^\pi(s_1) \cdot ds_1$ (Bellman)

$$= \int_\mathcal{S} p_0(s_0) \int_\mathcal{A} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0$$

$$+ \int_\mathcal{S} p_0(s_0) \int_\mathcal{A} \pi(s_0, a_0; \theta) \cdot \frac{\partial}{\partial \theta} (\mathcal{R}_{s_0}^{a_0} + \int_\mathcal{S} \gamma \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot V^\pi(s_1) \cdot ds_1) \cdot da_0 \cdot ds_0$$

Note: $\frac{\partial \mathcal{R}_{s_0}^a}{\partial \theta} = 0$, so remove that term

Proof of Policy Gradient Theorem

Now expand $Q^\pi(s_0, a_0)$ to $\mathcal{R}_{s_0}^{a_0} + \int_\mathcal{S} \gamma \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot V^\pi(s_1) \cdot ds_1$ (Bellman)

$$= \int_\mathcal{S} p_0(s_0) \int_\mathcal{A} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0$$
$$+ \int_\mathcal{S} p_0(s_0) \int_\mathcal{A} \pi(s_0, a_0; \theta) \cdot \frac{\partial}{\partial \theta} (\mathcal{R}_{s_0}^{a_0} + \int_\mathcal{S} \gamma \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot V^\pi(s_1) \cdot ds_1) \cdot da_0 \cdot ds_0$$

Note: $\frac{\partial \mathcal{R}_{s_0}^a}{\partial \theta} = 0$, so remove that term

$$= \int_\mathcal{S} p_0(s_0) \int_\mathcal{A} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a) \cdot da_0 \cdot ds_0$$
$$+ \int_\mathcal{S} p_0(s_0) \int_\mathcal{A} \pi(s_0, a_0; \theta) \cdot \frac{\partial}{\partial \theta} (\int_\mathcal{S} \gamma \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot V^\pi(s_1) \cdot ds_1) \cdot da_0 \cdot ds_0$$

Proof of Policy Gradient Theorem

Proof of Policy Gradient Theorem

Now take the $\frac{\partial}{\partial \theta}$ inside $\int_{\mathcal{S}}$ to apply only on $V^\pi(s_1)$

Proof of Policy Gradient Theorem

Now take the $\frac{\partial}{\partial \theta}$ inside $\int_{\mathcal{S}}$ to apply only on $V^\pi(s_1)$

$$= \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a) \cdot da_0 \cdot ds_0$$

$$+ \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \pi(s_0, a_0; \theta) \int_{\mathcal{S}} \gamma \cdot \mathcal{P}_{s_0, s_1}^{a_0} \frac{\partial V^\pi(s_1)}{\partial \theta} ds_1 \cdot da_0 \cdot ds_0$$

Proof of Policy Gradient Theorem

Now take the $\frac{\partial}{\partial \theta}$ inside $\int_{\mathcal{S}}$ to apply only on $V^\pi(s_1)$

$$= \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a) \cdot da_0 \cdot ds_0$$
$$+ \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \pi(s_0, a_0; \theta) \int_{\mathcal{S}} \gamma \cdot \mathcal{P}_{s_0, s_1}^{a_0} \frac{\partial V^\pi(s_1)}{\partial \theta} ds_1 \cdot da_0 \cdot ds_0$$

Now bring the outside $\int_{\mathcal{S}}$ and $\int_{\mathcal{A}}$ inside the inner $\int_{\mathcal{S}}$

Proof of Policy Gradient Theorem

Now take the $\frac{\partial}{\partial\theta}$ inside $\int_\mathcal{S}$ to apply only on $V^\pi(s_1)$

$$= \int_\mathcal{S} p_0(s_0) \int_\mathcal{A} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a) \cdot da_0 \cdot ds_0$$
$$+ \int_\mathcal{S} p_0(s_0) \int_\mathcal{A} \pi(s_0, a_0; \theta) \int_\mathcal{S} \gamma \cdot \mathcal{P}_{s_0,s_1}^{a_0} \frac{\partial V^\pi(s_1)}{\partial \theta} ds_1 \cdot da_0 \cdot ds_0$$

Now bring the outside $\int_\mathcal{S}$ and $\int_\mathcal{A}$ inside the inner $\int_\mathcal{S}$

$$= \int_\mathcal{S} p_0(s_0) \int_\mathcal{A} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0$$
$$+ \int_\mathcal{S} (\int_\mathcal{S} \gamma \cdot p_0(s_0) \int_\mathcal{A} \pi(s_0, a_0; \theta) \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot da_0 \cdot ds_0) \frac{\partial V^\pi(s_1)}{\partial \theta} \cdot ds_1$$

Policy Gradient Theorem

Policy Gradient Theorem

Note that $\int_{\mathcal{A}} \pi(s_0, a_0; \theta) \cdot \mathcal{P}_{s_0, s_1}^{a_0} \cdot da_0 = p(s_0 \rightarrow s_1, 1, \pi)$

Policy Gradient Theorem

Note that $\int_{\mathcal{A}} \pi(s_0, a_0; \theta) \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot da_0 = p(s_0 \rightarrow s_1, 1, \pi)$

$$= \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0$$
$$+ \int_{\mathcal{S}} (\int_{\mathcal{S}} \gamma \cdot p_0(s_0) \cdot p(s_0 \rightarrow s_1, 1, \pi) \cdot ds_0) \cdot \frac{\partial V^\pi(s_1)}{\partial \theta} \cdot ds_1$$

Policy Gradient Theorem

Note that $\int_{\mathcal{A}} \pi(s_0, a_0; \theta) \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot da_0 = p(s_0 \rightarrow s_1, 1, \pi)$

$$= \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0$$
$$+ \int_{\mathcal{S}} (\int_{\mathcal{S}} \gamma \cdot p_0(s_0) \cdot p(s_0 \rightarrow s_1, 1, \pi) \cdot ds_0) \cdot \frac{\partial V^\pi(s_1)}{\partial \theta} \cdot ds_1$$

Now expand $V^\pi(s_1)$ to $\int_{\mathcal{A}} \pi(s_1, a_1; \theta) \cdot Q^\pi(s_1, a_1) \cdot da_1$

Policy Gradient Theorem

Note that $\int_{\mathcal{A}} \pi(s_0, a_0; \theta) \cdot \mathcal{P}_{s_0,s_1}^{a_0} \cdot da_0 = p(s_0 \rightarrow s_1, 1, \pi)$

$$= \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0$$
$$+ \int_{\mathcal{S}} (\int_{\mathcal{S}} \gamma \cdot p_0(s_0) \cdot p(s_0 \rightarrow s_1, 1, \pi) \cdot ds_0) \cdot \frac{\partial V^\pi(s_1)}{\partial \theta} \cdot ds_1$$

Now expand $V^\pi(s_1)$ to $\int_{\mathcal{A}} \pi(s_1, a_1; \theta) \cdot Q^\pi(s_1, a_1) \cdot da_1$

$$= \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a_0) \cdot da \cdot ds_0$$
$$+ \int_{\mathcal{S}} (\int_{\mathcal{S}} \gamma \cdot p_0(s_0) p(s_0 \rightarrow s_1, 1, \pi) ds_0) \frac{\partial}{\partial \theta} (\int_{\mathcal{A}} \pi(s_1, a_1; \theta) \cdot Q^\pi(s_1, a_1) da_1) ds_1$$

Proof of Policy Gradient Theorem

Proof of Policy Gradient Theorem

We are now back to where we started calculating partial of $\int_{\mathcal{A}} \pi \cdot Q^{\pi} \cdot da$.

Proof of Policy Gradient Theorem

We are now back to where we started calculating partial of $\int_{\mathcal{A}} \pi \cdot Q^\pi \cdot da$. Follow the same process of splitting $\pi \cdot Q^\pi$, then Bellman-expanding Q^π (to calculate its partial), and iterate.

Proof of Policy Gradient Theorem

We are now back to where we started calculating partial of $\int_{\mathcal{A}} \pi \cdot Q^\pi \cdot da$. Follow the same process of splitting $\pi \cdot Q^\pi$, then Bellman-expanding Q^π (to calculate its partial), and iterate.

$$= \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^\pi(s_0, a_0) \cdot da_0 \cdot ds_0$$
$$+ \int_{\mathcal{S}} \int_{\mathcal{S}} \gamma \cdot p_0(s_0) p(s_0 \rightarrow s_1, 1, \pi) ds_0 (\int_{\mathcal{A}} \frac{\partial \pi(s_1, a_1; \theta)}{\partial \theta} Q^\pi(s_1, a_1) da_1 + \ldots) ds_1$$

Proof of Policy Gradient Theorem

We are now back to where we started calculating partial of $\int_{\mathcal{A}} \pi \cdot Q^{\pi} \cdot da$. Follow the same process of splitting $\pi \cdot Q^{\pi}$, then Bellman-expanding Q^{π} (to calculate its partial), and iterate.

$$= \int_{\mathcal{S}} p_0(s_0) \int_{\mathcal{A}} \frac{\partial \pi(s_0, a_0; \theta)}{\partial \theta} Q^{\pi}(s_0, a_0) \cdot da_0 \cdot ds_0$$
$$+ \int_{\mathcal{S}} \int_{\mathcal{S}} \gamma \cdot p_0(s_0) p(s_0 \rightarrow s_1, 1, \pi) ds_0 (\int_{\mathcal{A}} \frac{\partial \pi(s_1, a_1; \theta)}{\partial \theta} Q^{\pi}(s_1, a_1) da_1 + \ldots) ds_1$$

This iterative process leads us to:

$$= \sum_{t=0}^{\infty} \int_{\mathcal{S}} \int_{\mathcal{S}} \gamma^t \cdot p_0(s_0) \cdot p(s_0 \rightarrow s_t, t, \pi) \cdot ds_0 \int_{\mathcal{A}} \frac{\partial \pi(s_t, a_t; \theta)}{\partial \theta} Q^{\pi}(s_t, a_t) \cdot da_t \cdot ds_t$$

Proof of Policy Gradient Theorem

Proof of Policy Gradient Theorem

Bring $\sum_{t=0}^{\infty}$ inside the two $\int_{\mathcal{S}}$, and note that $\int_{\mathcal{A}} \frac{\partial \pi(s_t, a_t; \theta)}{\partial \theta} Q^{\pi}(s_t, a_t) \cdot da_t$ is independent of t.

Proof of Policy Gradient Theorem

Bring $\sum_{t=0}^{\infty}$ inside the two $\int_{\mathcal{S}}$, and note that $\int_{\mathcal{A}} \frac{\partial \pi(s_t,a_t;\theta)}{\partial \theta} Q^\pi(s_t,a_t) \cdot da_t$ is independent of t.

$$= \int_{\mathcal{S}} \int_{\mathcal{S}} \sum_{t=0}^{\infty} \gamma^t \cdot p_0(s_0) \cdot p(s_0 \rightarrow s, t, \pi) \cdot ds_0 \int_{\mathcal{A}} \frac{\partial \pi(s,a;\theta)}{\partial \theta} Q^\pi(s,a) \cdot da \cdot ds$$

Proof of Policy Gradient Theorem

Bring $\sum_{t=0}^{\infty}$ inside the two $\int_{\mathcal{S}}$, and note that $\int_{\mathcal{A}} \frac{\partial \pi(s_t,a_t;\theta)}{\partial \theta} Q^\pi(s_t, a_t) \cdot da_t$ is independent of t.

$$= \int_{\mathcal{S}} \int_{\mathcal{S}} \sum_{t=0}^{\infty} \gamma^t \cdot p_0(s_0) \cdot p(s_0 \rightarrow s, t, \pi) \cdot ds_0 \int_{\mathcal{A}} \frac{\partial \pi(s, a; \theta)}{\partial \theta} Q^\pi(s, a) \cdot da \cdot ds$$

Reminder that $\int_{\mathcal{S}} \sum_{t=0}^{\infty} \gamma^t \cdot p_0(s_0) \cdot p(s_0 \rightarrow s, t, \pi) \cdot ds_0 \stackrel{\text{def}}{=} \rho^\pi(s)$. So,

Proof of Policy Gradient Theorem

Bring $\sum_{t=0}^{\infty}$ inside the two $\int_{\mathcal{S}}$, and note that $\int_{\mathcal{A}} \frac{\partial \pi(s_t,a_t;\theta)}{\partial \theta} Q^{\pi}(s_t,a_t) \cdot da_t$ is independent of t.

$$= \int_{\mathcal{S}} \int_{\mathcal{S}} \sum_{t=0}^{\infty} \gamma^t \cdot p_0(s_0) \cdot p(s_0 \rightarrow s, t, \pi) \cdot ds_0 \int_{\mathcal{A}} \frac{\partial \pi(s,a;\theta)}{\partial \theta} Q^{\pi}(s,a) \cdot da \cdot ds$$

Reminder that $\int_{\mathcal{S}} \sum_{t=0}^{\infty} \gamma^t \cdot p_0(s_0) \cdot p(s_0 \rightarrow s, t, \pi) \cdot ds_0 \stackrel{\text{def}}{=} \rho^{\pi}(s)$. So,

$$\frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^{\pi}(s) \int_{\mathcal{A}} \frac{\partial \pi(s,a;\theta)}{\partial \theta} Q^{\pi}(s,a) \cdot da \cdot ds$$

$\mathbb{Q}.\mathbb{E}.\mathbb{D}.$

But we don't know the (true) $Q^\pi(s, a)$

But we don't know the (true) $Q^\pi(s, a)$

- Yes, and as usual, we will estimate it with a func approx $Q(s, a; w)$

But we don't know the (true) $Q^\pi(s, a)$

- Yes, and as usual, we will estimate it with a func approx $Q(s, a; w)$
- We refer to $Q(s, a; w)$ as the Critic func approx (with params w)

But we don't know the (true) $Q^\pi(s, a)$

- Yes, and as usual, we will estimate it with a func approx $Q(s, a; w)$
- We refer to $Q(s, a; w)$ as the Critic func approx (with params w)
- We refer to $\pi(s, a; \theta)$ as the Actor func approx (with params θ)

But we don't know the (true) $Q^\pi(s, a)$

- Yes, and as usual, we will estimate it with a func approx $Q(s, a; w)$
- We refer to $Q(s, a; w)$ as the Critic func approx (with params w)
- We refer to $\pi(s, a; \theta)$ as the Actor func approx (with params θ)
- But $Q(s, a; w)$ is a biased estimate of $Q^\pi(s, a)$, which is problematic

But we don't know the (true) $Q^\pi(s, a)$

- Yes, and as usual, we will estimate it with a func approx $Q(s, a; w)$
- We refer to $Q(s, a; w)$ as the Critic func approx (with params w)
- We refer to $\pi(s, a; \theta)$ as the Actor func approx (with params θ)
- But $Q(s, a; w)$ is a biased estimate of $Q^\pi(s, a)$, which is problematic
- To overcome bias $\Rightarrow$ *Compatible Function Approximation Theorem*

Compatible Function Approximation Theorem

Compatible Function Approximation Theorem

Theorem

If the following two conditions are satisfied:

Compatible Function Approximation Theorem

Theorem

If the following two conditions are satisfied:

1. *Critic gradient is* compatible *with the Actor score function*

$$\frac{\partial Q(s,a;w)}{\partial w} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta}$$

Compatible Function Approximation Theorem

Theorem

If the following two conditions are satisfied:

1. *Critic gradient is* compatible *with the Actor score function*

$$\frac{\partial Q(s,a;w)}{\partial w} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta}$$

2. *Critic parameters w minimize the following mean-squared error:*

$$\epsilon = \int_{\mathcal{S}} \rho^{\pi}(s) \int_{\mathcal{A}} \pi(s,a;\theta)(Q^{\pi}(s,a) - Q(s,a;w))^2 \cdot da \cdot ds$$

Compatible Function Approximation Theorem

Theorem

If the following two conditions are satisfied:

1. *Critic gradient is* compatible *with the Actor score function*

$$\frac{\partial Q(s,a;w)}{\partial w} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta}$$

2. *Critic parameters w minimize the following mean-squared error:*

$$\epsilon = \int_{\mathcal{S}} \rho^{\pi}(s) \int_{\mathcal{A}} \pi(s,a;\theta)(Q^{\pi}(s,a) - Q(s,a;w))^2 \cdot da \cdot ds$$

Then the Policy Gradient using critic $Q(s,a;w)$ is exact:

$$\frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^{\pi}(s) \int_{\mathcal{A}} \frac{\partial \pi(s,a;\theta)}{\partial \theta} Q(s,a;w) \cdot da \cdot ds$$

Proof of Compatible Function Approximation Theorem

Proof of Compatible Function Approximation Theorem

For w that minimizes

$$\epsilon = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s,a;\theta) \cdot (Q^\pi(s,a) - Q(s,a;w))^2 \cdot da \cdot ds,$$

Proof of Compatible Function Approximation Theorem

For w that minimizes

$$\epsilon = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot (Q^\pi(s, a) - Q(s, a; w))^2 \cdot da \cdot ds,$$

$$\int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot (Q^\pi(s, a) - Q(s, a; w)) \cdot \frac{\partial Q(s, a; w)}{\partial w} \cdot da \cdot ds = 0$$

Proof of Compatible Function Approximation Theorem

For w that minimizes

$$\epsilon = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot (Q^\pi(s, a) - Q(s, a; w))^2 \cdot da \cdot ds,$$

$$\int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot (Q^\pi(s, a) - Q(s, a; w)) \cdot \frac{\partial Q(s, a; w)}{\partial w} \cdot da \cdot ds = 0$$

But since $\frac{\partial Q(s,a;w)}{\partial w} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta}$, we have:

Proof of Compatible Function Approximation Theorem

For w that minimizes

$$\epsilon = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s,a;\theta) \cdot (Q^\pi(s,a) - Q(s,a;w))^2 \cdot da \cdot ds,$$

$$\int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s,a;\theta) \cdot (Q^\pi(s,a) - Q(s,a;w)) \cdot \frac{\partial Q(s,a;w)}{\partial w} \cdot da \cdot ds = 0$$

But since $\frac{\partial Q(s,a;w)}{\partial w} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta}$, we have:

$$\int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s,a;\theta) \cdot (Q^\pi(s,a) - Q(s,a;w)) \cdot \frac{\partial \log \pi(s,a;\theta)}{\partial \theta} \cdot da \cdot ds = 0$$

Proof of Compatible Function Approximation Theorem

For w that minimizes

$$\epsilon = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot (Q^\pi(s, a) - Q(s, a; w))^2 \cdot da \cdot ds,$$

$$\int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot (Q^\pi(s, a) - Q(s, a; w)) \cdot \frac{\partial Q(s, a; w)}{\partial w} \cdot da \cdot ds = 0$$

But since $\frac{\partial Q(s,a;w)}{\partial w} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta}$, we have:

$$\int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot (Q^\pi(s, a) - Q(s, a; w)) \cdot \frac{\partial \log \pi(s, a; \theta)}{\partial \theta} \cdot da \cdot ds = 0$$

$$\text{Therefore, } \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot Q^\pi(s, a) \cdot \frac{\partial \log \pi(s, a; \theta)}{\partial \theta} \cdot da \cdot ds$$
$$= \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot Q(s, a; w) \cdot \frac{\partial \log \pi(s, a; \theta)}{\partial \theta} \cdot da \cdot ds$$

Proof of Compatible Function Approximation Theorem

Proof of Compatible Function Approximation Theorem

But $$\frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot Q^\pi(s, a) \cdot \frac{\partial \log \pi(s, a; \theta)}{\partial \theta} \cdot da \cdot ds$$

Proof of Compatible Function Approximation Theorem

$$\text{But } \frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s,a;\theta) \cdot Q^\pi(s,a) \cdot \frac{\partial \log \pi(s,a;\theta)}{\partial \theta} \cdot da \cdot ds$$

$$\begin{aligned}\text{So, } \frac{\partial J(\pi_\theta)}{\partial \theta} &= \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s,a;\theta) \cdot Q(s,a;w) \cdot \frac{\partial \log \pi(s,a;\theta)}{\partial \theta} \cdot da \cdot ds \\ &= \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \frac{\partial \pi(s,a;\theta)}{\partial \theta} \cdot Q(s,a;w) \cdot da \cdot ds\end{aligned}$$

$\mathbb{Q}.\mathbb{E}.\mathbb{D}.$

Proof of Compatible Function Approximation Theorem

$$\text{But } \frac{\partial J(\pi_\theta)}{\partial \theta} = \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot Q^\pi(s, a) \cdot \frac{\partial \log \pi(s, a; \theta)}{\partial \theta} \cdot da \cdot ds$$

$$\begin{aligned}
\text{So, } \frac{\partial J(\pi_\theta)}{\partial \theta} &= \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s, a; \theta) \cdot Q(s, a; w) \cdot \frac{\partial \log \pi(s, a; \theta)}{\partial \theta} \cdot da \cdot ds \\
&= \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \frac{\partial \pi(s, a; \theta)}{\partial \theta} \cdot Q(s, a; w) \cdot da \cdot ds
\end{aligned}$$

$\mathbb{Q}.\mathbb{E}.\mathbb{D}.$

This means with conditions (1) and (2) of Compatible Function Approximation Theorem, we can use the critic func approx $Q(s, a; w)$ **and still have the exact Policy Gradient.**

So what does the algorithm look like?

So what does the algorithm look like?

- Generate a sufficient set of simulation paths $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$

So what does the algorithm look like?

- Generate a sufficient set of simulation paths $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$
- s_0 is sampled from the distribution $p_0(\cdot)$

So what does the algorithm look like?

- Generate a sufficient set of simulation paths $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$
- s_0 is sampled from the distribution $p_0(\cdot)$
- a_t is sampled from $\pi(s_t, \cdot; \theta)$

So what does the algorithm look like?

- Generate a sufficient set of simulation paths $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$
- s_0 is sampled from the distribution $p_0(\cdot)$
- a_t is sampled from $\pi(s_t, \cdot; \theta)$
- s_{t+1} sampled from transition probs and r_{t+1} from reward func

So what does the algorithm look like?

- Generate a sufficient set of simulation paths $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$
- s_0 is sampled from the distribution $p_0(\cdot)$
- a_t is sampled from $\pi(s_t, \cdot; \theta)$
- s_{t+1} sampled from transition probs and r_{t+1} from reward func
- Sum $\gamma^t \cdot \frac{\partial \log \pi(s_t, a_t; \theta)}{\partial \theta} \cdot Q(s_t, a_t; w)$ over t and over paths

So what does the algorithm look like?

- Generate a sufficient set of simulation paths $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$
- s_0 is sampled from the distribution $p_0(\cdot)$
- a_t is sampled from $\pi(s_t, \cdot; \theta)$
- s_{t+1} sampled from transition probs and r_{t+1} from reward func
- Sum $\gamma^t \cdot \frac{\partial \log \pi(s_t, a_t; \theta)}{\partial \theta} \cdot Q(s_t, a_t; w)$ over t and over paths
- This gives an unbiased estimate of $\frac{\partial J(\pi_\theta)}{\partial \theta}$

So what does the algorithm look like?

- Generate a sufficient set of simulation paths $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$
- s_0 is sampled from the distribution $p_0(\cdot)$
- a_t is sampled from $\pi(s_t, \cdot; \theta)$
- s_{t+1} sampled from transition probs and r_{t+1} from reward func
- Sum $\gamma^t \cdot \frac{\partial \log \pi(s_t, a_t; \theta)}{\partial \theta} \cdot Q(s_t, a_t; w)$ over t and over paths
- This gives an unbiased estimate of $\frac{\partial J(\pi_\theta)}{\partial \theta}$
- To reduce variance, use advantage function estimate $A(s, a; w, v) = Q(s, a; w) - V(s; v)$ (instead of $Q(s, a; w)$)

So what does the algorithm look like?

- Generate a sufficient set of simulation paths $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$
- s_0 is sampled from the distribution $p_0(\cdot)$
- a_t is sampled from $\pi(s_t, \cdot; \theta)$
- s_{t+1} sampled from transition probs and r_{t+1} from reward func
- Sum $\gamma^t \cdot \frac{\partial \log \pi(s_t, a_t; \theta)}{\partial \theta} \cdot Q(s_t, a_t; w)$ over t and over paths
- This gives an unbiased estimate of $\frac{\partial J(\pi_\theta)}{\partial \theta}$
- To reduce variance, use advantage function estimate $A(s, a; w, v) = Q(s, a; w) - V(s; v)$ (instead of $Q(s, a; w)$)

So what does the algorithm look like?

- Generate a sufficient set of simulation paths $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$
- s_0 is sampled from the distribution $p_0(\cdot)$
- a_t is sampled from $\pi(s_t, \cdot; \theta)$
- s_{t+1} sampled from transition probs and r_{t+1} from reward func
- Sum $\gamma^t \cdot \frac{\partial \log \pi(s_t, a_t; \theta)}{\partial \theta} \cdot Q(s_t, a_t; w)$ over t and over paths
- This gives an unbiased estimate of $\frac{\partial J(\pi_\theta)}{\partial \theta}$
- To reduce variance, use advantage function estimate $A(s, a; w, v) = Q(s, a; w) - V(s; v)$ (instead of $Q(s, a; w)$)

$$\text{Note: } \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \frac{\partial \pi(s, a; \theta)}{\partial \theta} \cdot V(s; v) \cdot da \cdot ds$$
$$= \int_{\mathcal{S}} \rho^\pi(s) \cdot V(s; v) \frac{\partial}{\partial \theta} \left(\int_{\mathcal{A}} \pi(s, a; \theta) \cdot da \right) \cdot ds = 0$$

How to enable Compatible Function Approximation

How to enable Compatible Function Approximation

A simple way to enable Compatible Function Approximation

How to enable Compatible Function Approximation

A simple way to enable Compatible Function Approximation $\frac{\partial Q(s,a;w)}{\partial w_i} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i}, \forall i$ is to set $Q(s,a;w)$ to be linear in its features.

How to enable Compatible Function Approximation

A simple way to enable Compatible Function Approximation $\frac{\partial Q(s,a;w)}{\partial w_i} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i}, \forall i$ is to set $Q(s,a;w)$ to be linear in its features.

$$Q(s,a;w) = \sum_{i=1}^{n} \phi_i(s,a) \cdot w_i = \sum_{i=1}^{n} \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i} \cdot w_i$$

How to enable Compatible Function Approximation

A simple way to enable Compatible Function Approximation $\frac{\partial Q(s,a;w)}{\partial w_i} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i}, \forall i$ is to set $Q(s,a;w)$ to be linear in its features.

$$Q(s,a;w) = \sum_{i=1}^{n} \phi_i(s,a) \cdot w_i = \sum_{i=1}^{n} \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i} \cdot w_i$$

We note below that a compatible $Q(s,a;w)$ serves as an approximation of the advantage function.

How to enable Compatible Function Approximation

A simple way to enable Compatible Function Approximation $\frac{\partial Q(s,a;w)}{\partial w_i} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i}, \forall i$ is to set $Q(s,a;w)$ to be linear in its features.

$$Q(s,a;w) = \sum_{i=1}^{n} \phi_i(s,a) \cdot w_i = \sum_{i=1}^{n} \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i} \cdot w_i$$

We note below that a compatible $Q(s,a;w)$ serves as an approximation of the advantage function.

$$\int_{\mathcal{A}} \pi(s,a;\theta) \cdot Q(s,a;w) \cdot da = \int_{\mathcal{A}} \pi(s,a;\theta) \cdot (\sum_{i=1}^{n} \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i} \cdot w_i) \cdot da$$

How to enable Compatible Function Approximation

A simple way to enable Compatible Function Approximation $\frac{\partial Q(s,a;w)}{\partial w_i} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i}, \forall i$ is to set $Q(s,a;w)$ to be linear in its features.

$$Q(s,a;w) = \sum_{i=1}^{n} \phi_i(s,a) \cdot w_i = \sum_{i=1}^{n} \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i} \cdot w_i$$

We note below that a compatible $Q(s,a;w)$ serves as an approximation of the advantage function.

$$\int_{\mathcal{A}} \pi(s,a;\theta) \cdot Q(s,a;w) \cdot da = \int_{\mathcal{A}} \pi(s,a;\theta) \cdot (\sum_{i=1}^{n} \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i} \cdot w_i) \cdot da$$

$$\int_{\mathcal{A}} \cdot (\sum_{i=1}^{n} \frac{\partial \pi(s,a;\theta)}{\partial \theta_i} \cdot w_i) \cdot da = \sum_{i=1}^{n} (\int_{\mathcal{A}} \frac{\partial \pi(s,a;\theta)}{\partial \theta_i} \cdot da) \cdot w_i$$

How to enable Compatible Function Approximation

A simple way to enable Compatible Function Approximation $\frac{\partial Q(s,a;w)}{\partial w_i} = \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i}, \forall i$ is to set $Q(s,a;w)$ to be linear in its features.

$$Q(s,a;w) = \sum_{i=1}^{n} \phi_i(s,a) \cdot w_i = \sum_{i=1}^{n} \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i} \cdot w_i$$

We note below that a compatible $Q(s,a;w)$ serves as an approximation of the advantage function.

$$\int_{\mathcal{A}} \pi(s,a;\theta) \cdot Q(s,a;w) \cdot da = \int_{\mathcal{A}} \pi(s,a;\theta) \cdot (\sum_{i=1}^{n} \frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i} \cdot w_i) \cdot da$$

$$\int_{\mathcal{A}} \cdot (\sum_{i=1}^{n} \frac{\partial \pi(s,a;\theta)}{\partial \theta_i} \cdot w_i) \cdot da = \sum_{i=1}^{n} (\int_{\mathcal{A}} \frac{\partial \pi(s,a;\theta)}{\partial \theta_i} \cdot da) \cdot w_i$$

$$= \sum_{i=1}^{n} \frac{\partial}{\partial \theta_i} (\int_{\mathcal{A}} \pi(s,a;\theta) \cdot da) \cdot w_i = \sum_{i=1}^{n} \frac{\partial 1}{\partial \theta_i} \cdot w_i = 0$$

Fisher Information Matrix

Fisher Information Matrix

Denoting $[\frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i}], i = 1, \ldots, n$ as the score column vector $SC(s, a; \theta)$ and denoting $\frac{\partial J(\pi_\theta)}{\partial \theta}$ as $\nabla_\theta J(\pi_\theta)$, assuming compatible linear-approx critic:

Fisher Information Matrix

Denoting $[\frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i}], i = 1, \ldots, n$ as the score column vector $SC(s,a;\theta)$ and denoting $\frac{\partial J(\pi_\theta)}{\partial \theta}$ as $\nabla_\theta J(\pi_\theta)$, assuming compatible linear-approx critic:

$$\begin{aligned}
\nabla_\theta J(\pi_\theta) &= \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s,a;\theta) \cdot (SC(s,a;\theta) \cdot SC(s,a;\theta)^T \cdot w) \cdot da \cdot ds \\
&= E_{s \sim \rho^\pi, a \sim \pi}[SC(s,a;\theta) \cdot SC(s,a;\theta)^T] \cdot w \\
&= FIM_{\rho^\pi, \pi}(\theta) \cdot w
\end{aligned}$$

Fisher Information Matrix

Denoting $[\frac{\partial \log \pi(s,a;\theta)}{\partial \theta_i}], i = 1, \ldots, n$ as the score column vector $SC(s,a;\theta)$ and denoting $\frac{\partial J(\pi_\theta)}{\partial \theta}$ as $\nabla_\theta J(\pi_\theta)$, assuming compatible linear-approx critic:

$$\begin{aligned}
\nabla_\theta J(\pi_\theta) &= \int_{\mathcal{S}} \rho^\pi(s) \int_{\mathcal{A}} \pi(s,a;\theta) \cdot (SC(s,a;\theta) \cdot SC(s,a;\theta)^T \cdot w) \cdot da \cdot ds \\
&= E_{s \sim \rho^\pi, a \sim \pi}[SC(s,a;\theta) \cdot SC(s,a;\theta)^T] \cdot w \\
&= FIM_{\rho^\pi,\pi}(\theta) \cdot w
\end{aligned}$$

where $FIM_{\rho_\pi,\pi}(\theta)$ is the Fisher Information Matrix w.r.t. $s \sim \rho^\pi, a \sim \pi$.

Natural Policy Gradient

Natural Policy Gradient

- Recall the idea of Natural Gradient from Numerical Optimization

Natural Policy Gradient

- Recall the idea of Natural Gradient from Numerical Optimization
- Natural gradient $\nabla_\theta^{nat} J(\pi_\theta)$ is the direction of optimal θ movement

Natural Policy Gradient

- Recall the idea of Natural Gradient from Numerical Optimization
- Natural gradient $\nabla_\theta^{nat} J(\pi_\theta)$ is the direction of optimal θ movement
- In terms of the KL-divergence metric (versus plain Euclidean norm)

Natural Policy Gradient

- Recall the idea of Natural Gradient from Numerical Optimization
- Natural gradient $\nabla_\theta^{nat} J(\pi_\theta)$ is the direction of optimal θ movement
- In terms of the KL-divergence metric (versus plain Euclidean norm)
- Natural gradient yields better convergence (we won't cover proof)

Natural Policy Gradient

- Recall the idea of Natural Gradient from Numerical Optimization
- Natural gradient $\nabla_\theta^{nat} J(\pi_\theta)$ is the direction of optimal θ movement
- In terms of the KL-divergence metric (versus plain Euclidean norm)
- Natural gradient yields better convergence (we won't cover proof)

Natural Policy Gradient

- Recall the idea of Natural Gradient from Numerical Optimization
- Natural gradient $\nabla_\theta^{nat} J(\pi_\theta)$ is the direction of optimal θ movement
- In terms of the KL-divergence metric (versus plain Euclidean norm)
- Natural gradient yields better convergence (we won't cover proof)

Formally defined as: $\nabla_\theta J(\pi_\theta) = FIM_{\rho_\pi,\pi}(\theta) \cdot \nabla_\theta^{nat} J(\pi_\theta)$

Natural Policy Gradient

- Recall the idea of Natural Gradient from Numerical Optimization
- Natural gradient $\nabla_\theta^{nat} J(\pi_\theta)$ is the direction of optimal θ movement
- In terms of the KL-divergence metric (versus plain Euclidean norm)
- Natural gradient yields better convergence (we won't cover proof)

$$\text{Formally defined as: } \nabla_\theta J(\pi_\theta) = FIM_{\rho_\pi,\pi}(\theta) \cdot \nabla_\theta^{nat} J(\pi_\theta)$$

$$\text{Therefore, } \nabla_\theta^{nat} J(\pi_\theta) = w$$

Natural Policy Gradient

- Recall the idea of Natural Gradient from Numerical Optimization
- Natural gradient $\nabla_\theta^{nat} J(\pi_\theta)$ is the direction of optimal θ movement
- In terms of the KL-divergence metric (versus plain Euclidean norm)
- Natural gradient yields better convergence (we won't cover proof)

$$\text{Formally defined as: } \nabla_\theta J(\pi_\theta) = FIM_{\rho_\pi,\pi}(\theta) \cdot \nabla_\theta^{nat} J(\pi_\theta)$$

$$\text{Therefore, } \nabla_\theta^{nat} J(\pi_\theta) = w$$

This compact result is great for our algorithm:

Natural Policy Gradient

- Recall the idea of Natural Gradient from Numerical Optimization
- Natural gradient $\nabla_\theta^{nat} J(\pi_\theta)$ is the direction of optimal θ movement
- In terms of the KL-divergence metric (versus plain Euclidean norm)
- Natural gradient yields better convergence (we won't cover proof)

$$\text{Formally defined as: } \nabla_\theta J(\pi_\theta) = FIM_{\rho_\pi,\pi}(\theta) \cdot \nabla_\theta^{nat} J(\pi_\theta)$$

$$\text{Therefore, } \nabla_\theta^{nat} J(\pi_\theta) = w$$

This compact result is great for our algorithm:

- Update Critic params w with the critic loss gradient (at step t) as:

$$\gamma^t \cdot (SC(s_t, a_t, \theta) \cdot w - r_t - \gamma \cdot SC(s_{t+1}, a_{t+1}, \theta) \cdot w) \cdot SC(s_t, a_t, \theta)$$

Natural Policy Gradient

- Recall the idea of Natural Gradient from Numerical Optimization
- Natural gradient $\nabla_\theta^{nat} J(\pi_\theta)$ is the direction of optimal θ movement
- In terms of the KL-divergence metric (versus plain Euclidean norm)
- Natural gradient yields better convergence (we won't cover proof)

$$\text{Formally defined as: } \nabla_\theta J(\pi_\theta) = FIM_{\rho_\pi,\pi}(\theta) \cdot \nabla_\theta^{nat} J(\pi_\theta)$$

$$\text{Therefore, } \nabla_\theta^{nat} J(\pi_\theta) = w$$

This compact result is great for our algorithm:

- Update Critic params w with the critic loss gradient (at step t) as:

$$\gamma^t \cdot (SC(s_t, a_t, \theta) \cdot w - r_t - \gamma \cdot SC(s_{t+1}, a_{t+1}, \theta) \cdot w) \cdot SC(s_t, a_t, \theta)$$

- Update Actor params θ in the direction equal to value of w