Approximate Policy Iteration
And Performance Difference Lemma

Recap: Supervised Learning and Data Generation Process

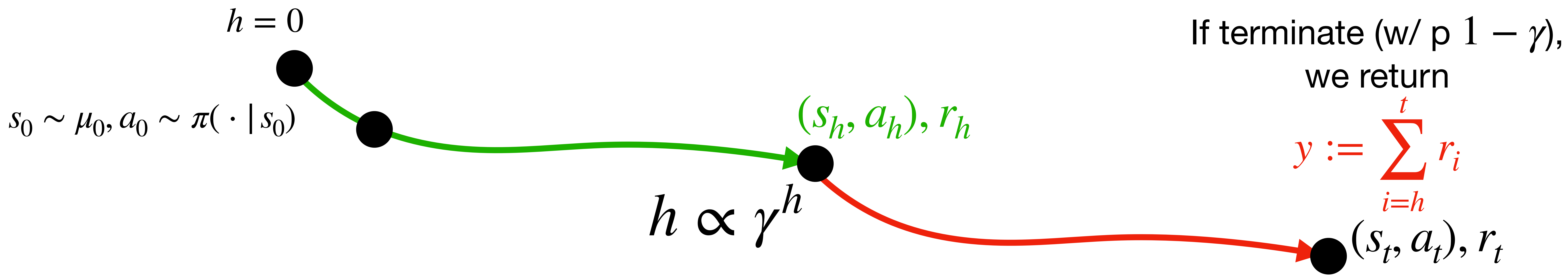
Recap: Supervised Learning and Data Generation Process

1. Supervised Learning works (in both theory and practice) if there is no train-test mismatch

Recap: Supervised Learning and Data Generation Process

1. Supervised Learning works (in both theory and practice) if there is no train-test mismatch

2. A **data generation process**: given π, we roll-in & roll-out to get (s, a, y),
where $(s, a) \sim d^{\pi}_{\mu_0}$, $\mathbb{E}[y] = Q^{\pi}(s, a)$



Plans for Today

1. Algorithm: Approximate Policy Iteration

2. When does API could make monotonic improvement?

3. Performance Difference Lemma (Another important lemma)

Estimating the function $Q^{\pi}(s, a)$ using Least Square Regression

Given π, repeat N times of the roll-in & roll-out process,
we get a training dataset of N samples:

$$\mathscr{D}^{\pi} = \left\{s^i, a^i, y^i\right\}_{i=1}^{N}$$

Estimating the function $Q^\pi(s, a)$ using Least Square Regression

Given π, repeat N times of the roll-in & roll-out process,
we get a training dataset of N samples:

$$\mathcal{D}^\pi = \left\{s^i, a^i, y^i\right\}_{i=1}^N$$

Least square regression:

$$\widehat{Q}^\pi \in \arg\min_{Q \in \mathcal{Q}} \sum_{i=1}^N \left(Q(s^i, a^i) - y^i\right)^2$$

Estimating the function $Q^\pi(s, a)$ using Least Square Regression

Given π, repeat N times of the roll-in & roll-out process,
we get a training dataset of N samples:

$$\mathscr{D}^\pi = \left\{s^i, a^i, y^i\right\}_{i=1}^N$$

Least square regression:

$$\widehat{Q}^\pi \in \arg\min_{Q \in \mathscr{Q}} \sum_{i=1}^N \left(Q(s^i, a^i) - y^i\right)^2$$

Assume successful supervise learning, we have:

$$\mathbb{E}_{s,a \sim d_\mu^\pi} \left(\widehat{Q}^\pi(s, a) - Q^\pi(s, a) \right)^2 \leq \delta,$$

where δ being some small number (e.g., $1/\sqrt{N}$)

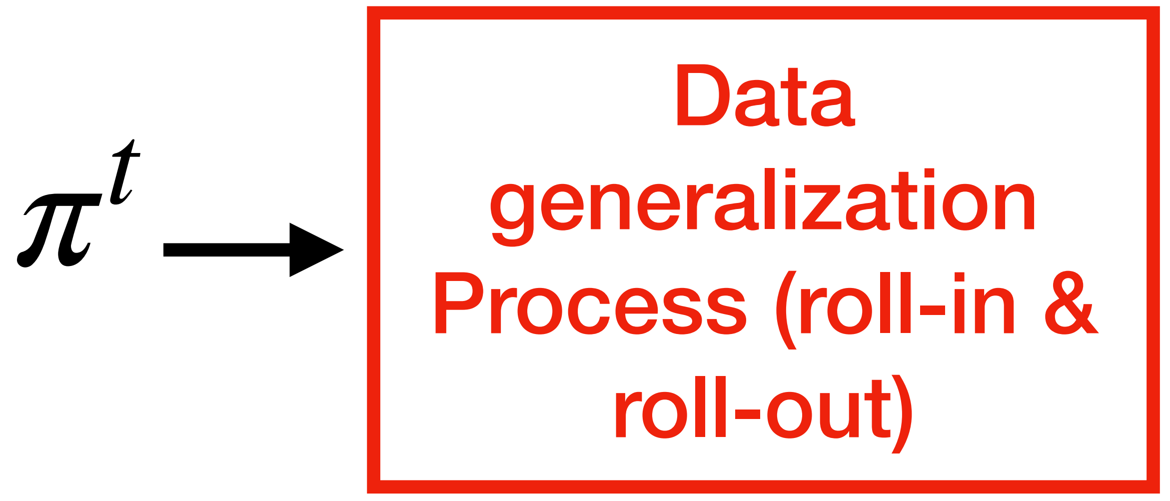
Put things together: Algorithm of Approximate Policy Iteration

Put things together: Algorithm of Approximate Policy Iteration

Initialize $\widehat{Q}^0 \in \mathcal{Q}$, set $\pi^0(s) = \arg\max_a \widehat{Q}^0(s, a)$

Put things together: Algorithm of Approximate Policy Iteration

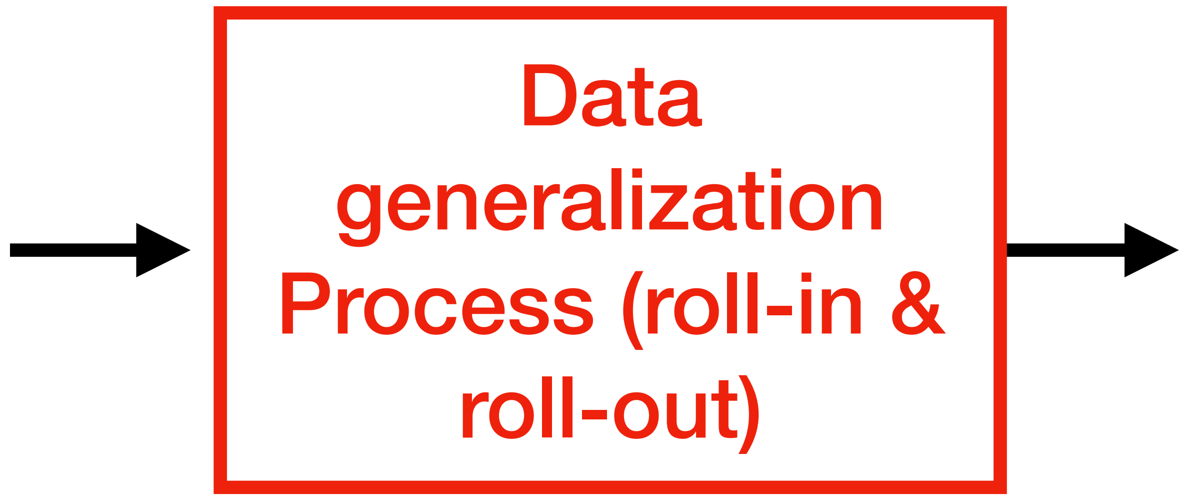
Initialize $\widehat{Q}^0 \in \mathcal{Q}$, set $\pi^0(s) = \arg\max_a \widehat{Q}^0(s, a)$



Put things together: Algorithm of Approximate Policy Iteration

Initialize $\widehat{Q}^0 \in \mathcal{Q}$, set $\pi^0(s) = \arg\max_a \widehat{Q}^0(s,a)$

π^t



$$\mathcal{D}^{\pi^t} = \{s^i, a^i, y^i\}_{i=1}^N$$

$$\mathbb{E}[y_i] = Q^{\pi^t}(s_i, a_i)$$

$$s^i, a^i \sim d^\pi_{\mu_0}$$

Put things together: Algorithm of Approximate Policy Iteration

Initialize $\widehat{Q}^0 \in \mathcal{Q}$, set $\pi^0(s) = \arg\max_a \widehat{Q}^0(s, a)$

π^t → **Data generalization Process (roll-in & roll-out)** →

$$\mathcal{D}^{\pi^t} = \{s^i, a^i, y^i\}_{i=1}^N$$

$$\mathbb{E}[y_i] = Q^{\pi^t}(s_i, a_i)$$

$$s^i, a^i \sim d^{\pi}_{\mu_0}$$

→ **Least Square Regression oracle**

Put things together: Algorithm of Approximate Policy Iteration

Initialize $\widehat{Q}^0 \in \mathcal{Q}$, set $\pi^0(s) = \arg\max_a \widehat{Q}^0(s,a)$

π^t → Data generalization Process (roll-in & roll-out) →

$$\mathcal{D}^{\pi^t} = \{s^i, a^i, y^i\}_{i=1}^N$$

$$\mathbb{E}[y_i] = Q^{\pi^t}(s_i, a_i)$$

$$s^i, a^i \sim d^{\pi}_{\mu_0}$$

→ Least Square Regression oracle ↓

$$\widehat{Q}^t \in \arg\min_{Q \in \mathcal{Q}} \sum_{i=1}^N (Q(s^i, a^i) - y^i)^2$$

Put things together: Algorithm of Approximate Policy Iteration

Initialize $\widehat{Q}^0 \in \mathcal{Q}$, set $\pi^0(s) = \arg\max_a \widehat{Q}^0(s, a)$

π^t → **Data generalization Process (roll-in & roll-out)** →

$$\mathcal{D}^{\pi^t} = \{s^i, a^i, y^i\}_{i=1}^N$$

$$\mathbb{E}[y_i] = Q^{\pi^t}(s_i, a_i)$$

$$s^i, a^i \sim d^{\pi}_{\mu_0}$$

→ **Least Square Regression oracle** ↓

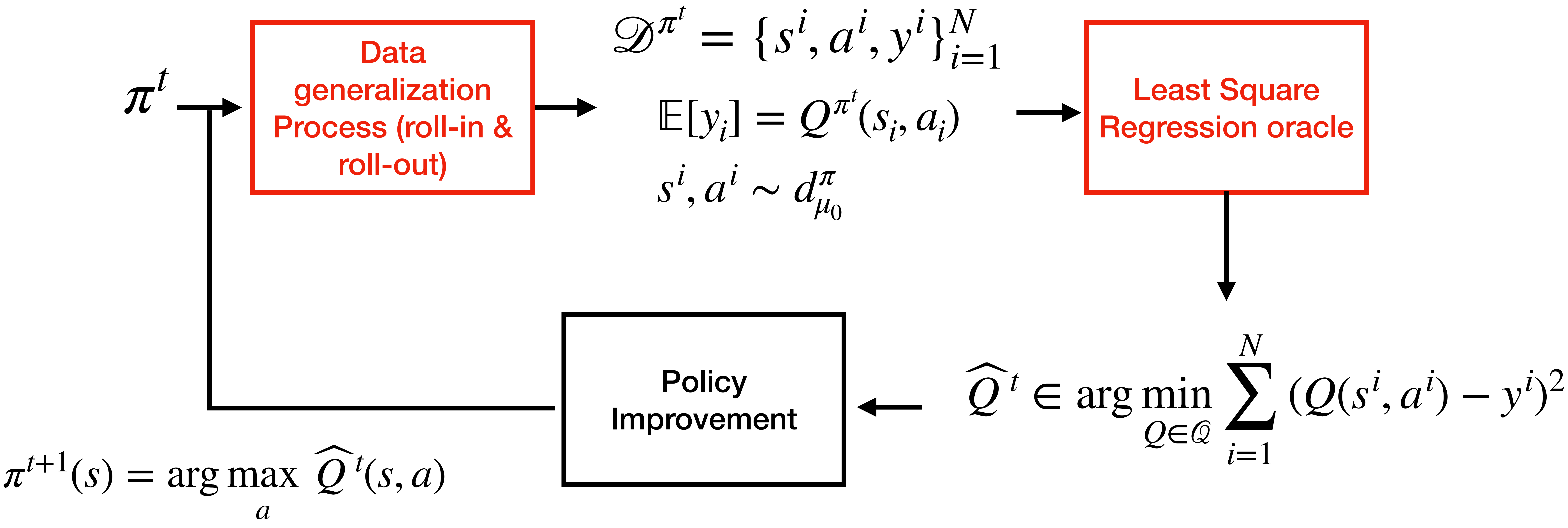
$$\widehat{Q}^t \in \arg\min_{Q \in \mathcal{Q}} \sum_{i=1}^N (Q(s^i, a^i) - y^i)^2$$

← **Policy Improvement**

$$\pi^{t+1}(s) = \arg\max_a \widehat{Q}^t(s, a)$$

Put things together: Algorithm of Approximate Policy Iteration

Initialize $\widehat{Q}^0 \in \mathcal{Q}$, set $\pi^0(s) = \arg\max_a \widehat{Q}^0(s,a)$



Put things together: Algorithm of Approximate Policy Iteration

Initialize $\widehat{Q}^0 \in \mathcal{Q}$, set $\pi^0(s) = \arg\min_a \widehat{Q}^0(s, a)$

For $t = 0, \ldots,$

Repeat N roll-in & roll-out w/ π^t; get N training points $\{s^i, a^i, y^i\}_{i=1}^N$

Least Square Minimization: $\widehat{Q}^t \in \arg\min_{Q \in \mathcal{Q}} \sum_{i=1}^{N} \left(Q(s^i, a^i) - y^i\right)^2$

Policy Improvement $\pi^{t+1}(s) = \arg\max_a \widehat{Q}^t(s, a)$

Plans for Today

✓ 1. Algorithm: Approximate Policy Iteration

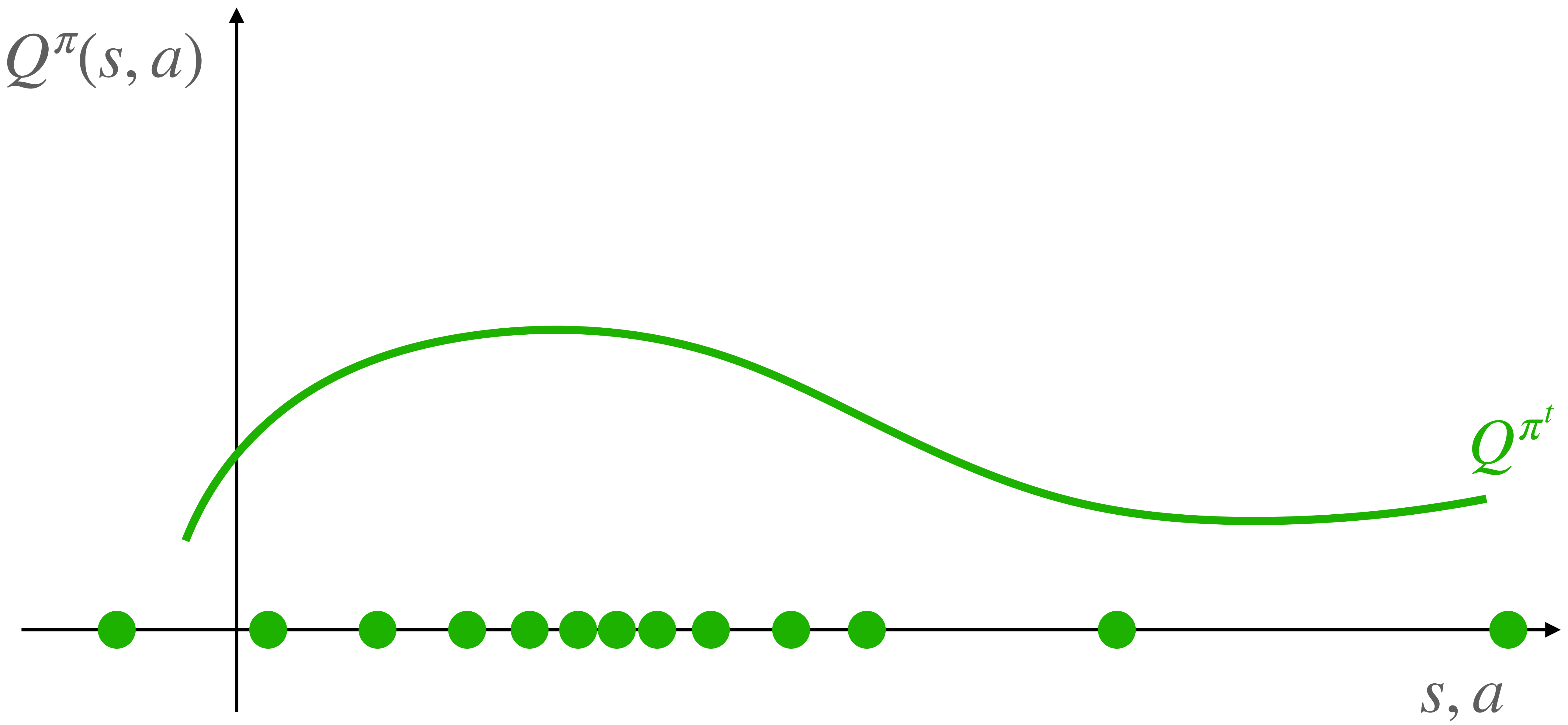
2. When does API could make monotonic improvement?

3. Performance Difference Lemma (Another important lemma)

The Oscillation of API from Abrupt Distribution Change

Recall that Policy Iteration w/ known (P, r) makes monotonic improvement;

But API cannot guarantee to make monotonic improvement:



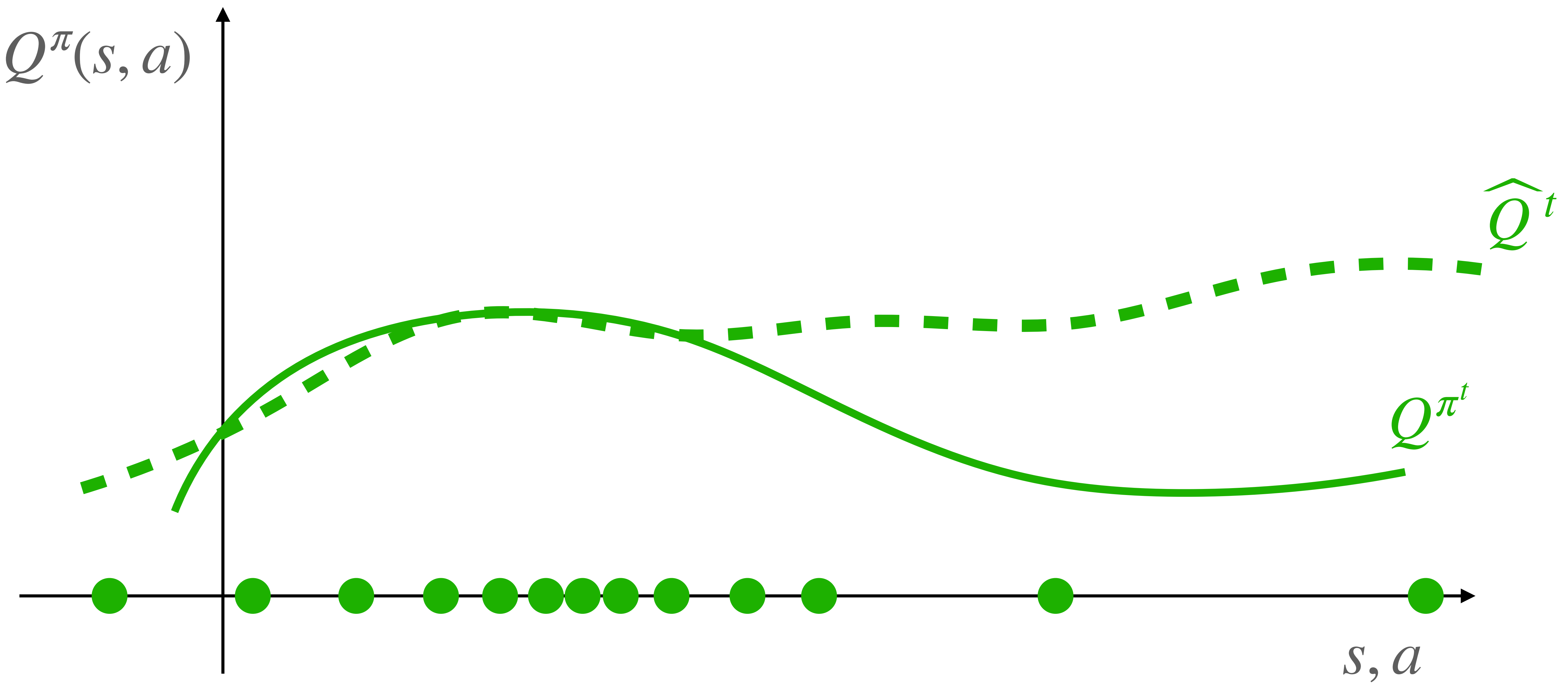
Green dots: (s, a) from π^t

Red dots: (s, a) from π^{t+1}

The Oscillation of API from Abrupt Distribution Change

Recall that Policy Iteration w/ known (P, r) makes monotonic improvement;

But API cannot guarantee to make monotonic improvement:



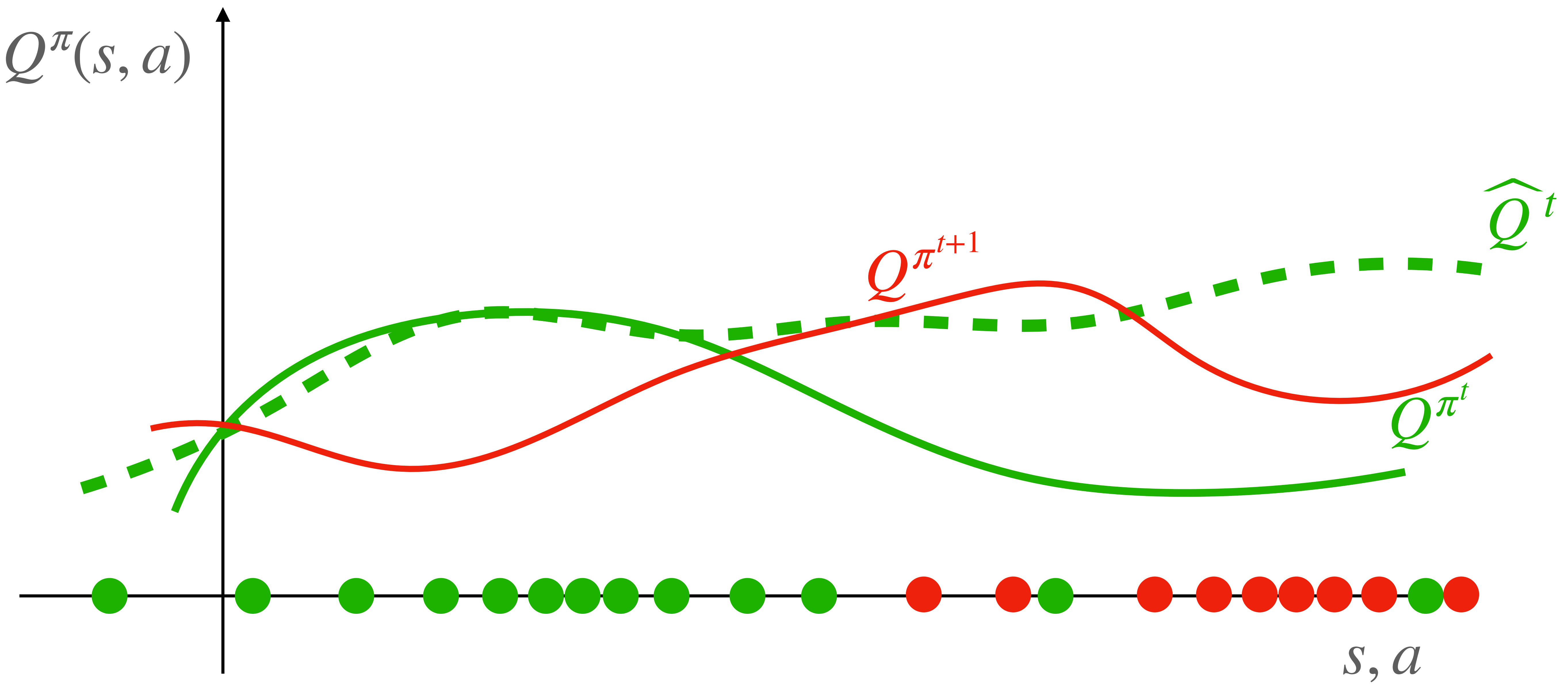
Green dots: (s, a) from π^t

Red dots: (s, a) from π^{t+1}

The Oscillation of API from Abrupt Distribution Change

Recall that Policy Iteration w/ known (P, r) makes monotonic improvement;

But API cannot guarantee to make monotonic improvement:



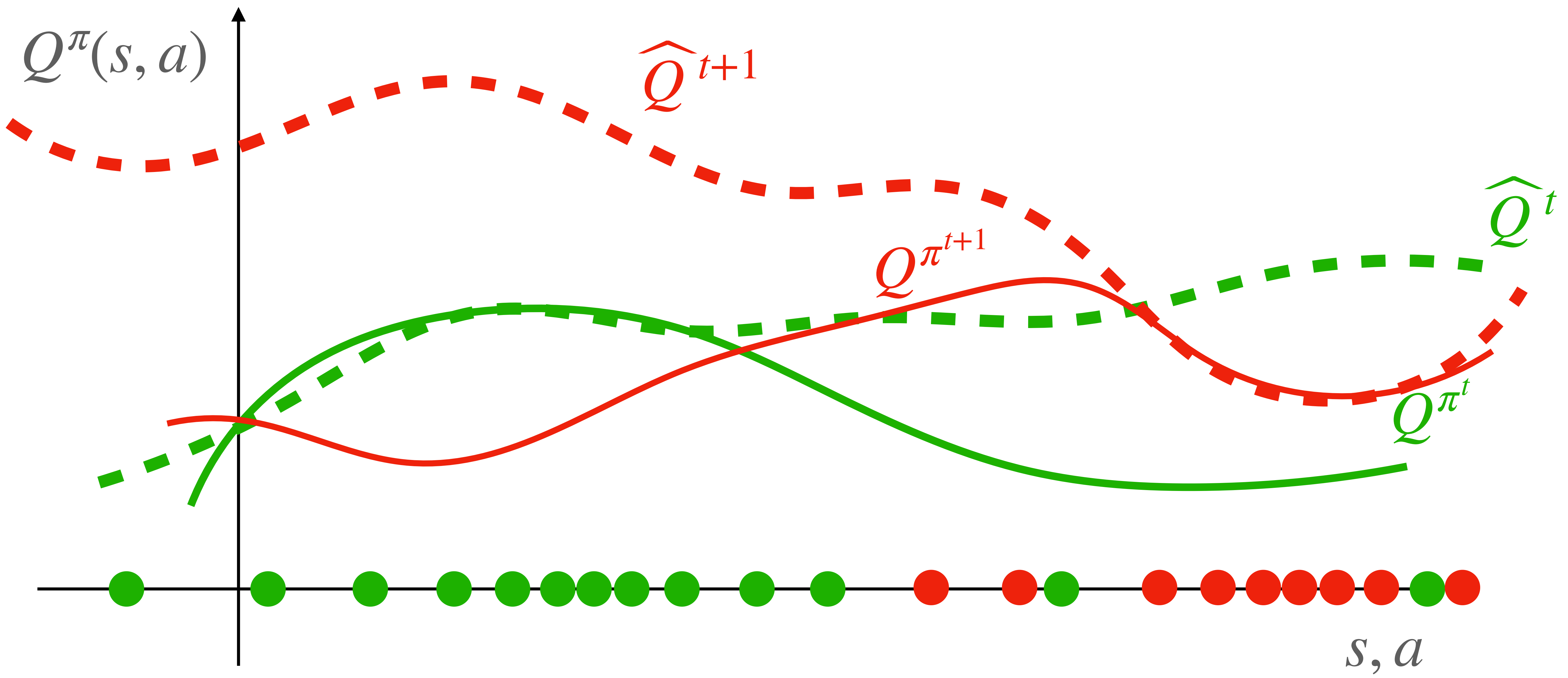
Green dots: (s, a) from π^t

Red dots: (s, a) from π^{t+1}

The Oscillation of API from Abrupt Distribution Change

Recall that Policy Iteration w/ known (P, r) makes monotonic improvement;

But API cannot guarantee to make monotonic improvement:



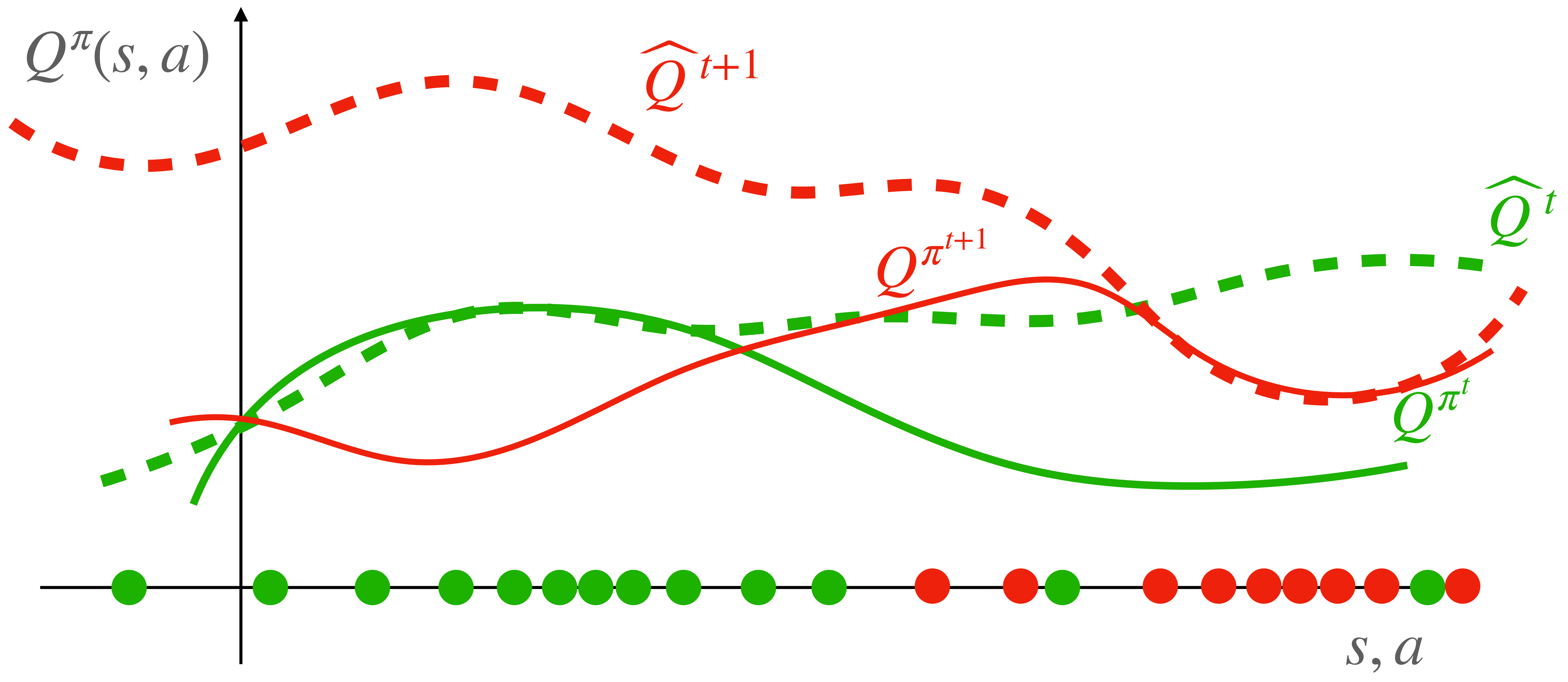
Green dots: (s, a) from π^t

Red dots: (s, a) from π^{t+1}

The Oscillation of API from Abrupt Distribution Change

Recall that Policy Iteration w/ known (P, r) makes monotonic improvement;

But API cannot guarantee to make monotonic improvement:



Green dots: (s, a) from π^t

Red dots: (s, a) from π^{t+1}

Oscillation between two updates:
No monotonic improvement

Key Issue: Abrupt Policy Change, i.e., $d_{\mu_0}^{\pi^{t+1}}$ and $d_{\mu_0}^{\pi^t}$ could be widely different

Key Issue: Abrupt Policy Change, i.e., $d_{\mu_0}^{\pi^{t+1}}$ and $d_{\mu_0}^{\pi^t}$ could be widely different

Our estimator $\widehat{Q}^t$ is only good under $d_{\mu_0}^{\pi^t}$,

i.e. $\mathbb{E}_{s\sim d_{\mu_0}^{\pi^t}}(\widehat{Q}^t(s,a) - Q^{\pi^t}(s,a))^2$ small,

Key Issue: Abrupt Policy Change, i.e., $d_{\mu_0}^{\pi^{t+1}}$ and $d_{\mu_0}^{\pi^t}$ could be widely different

Our estimator $\widehat{Q}^t$ is only good under $d_{\mu_0}^{\pi^t}$,

i.e. $\mathbb{E}_{s\sim d_{\mu_0}^{\pi^t}}(\widehat{Q}^t(s,a) - Q^{\pi^t}(s,a))^2$ small,

but $\mathbb{E}_{s\sim d_{\mu_0}^{\pi^{t+1}}}(\widehat{Q}^t(s,a) - Q^{\pi^t}(s,a))^2$ **might be arbitrarily big**

Key Issue: Abrupt Policy Change, i.e., $d_{\mu_0}^{\pi^{t+1}}$ and $d_{\mu_0}^{\pi^t}$ could be widely different

Our estimator $\widehat{Q}^t$ is only good under $d_{\mu_0}^{\pi^t}$,

i.e. $\mathbb{E}_{s\sim d_{\mu_0}^{\pi^t}}(\widehat{Q}^t(s,a) - Q^{\pi^t}(s,a))^2$ small,

but $\mathbb{E}_{s\sim d_{\mu_0}^{\pi^{t+1}}}(\widehat{Q}^t(s,a) - Q^{\pi^t}(s,a))^2$ **might be arbitrarily big**

To make API to make monotonic improvement, we need a strong coverage assumption:

A strong Concentrability Coefficient: $\max_{\pi} \max_{s} \frac{d_{\mu_0}^{\pi}(s)}{\mu_0(s)} \leq C < \infty$

Key Issue: Abrupt Policy Change, i.e., $d_{\mu_0}^{\pi^{t+1}}$ and $d_{\mu_0}^{\pi^t}$ could be widely different

Our estimator $\widehat{Q}^t$ is only good under $d_{\mu_0}^{\pi^t}$,

i.e. $\mathbb{E}_{s\sim d_{\mu_0}^{\pi^t}}(\widehat{Q}^t(s,a) - Q^{\pi^t}(s,a))^2$ small,

but $\mathbb{E}_{s\sim d_{\mu_0}^{\pi^{t+1}}}(\widehat{Q}^t(s,a) - Q^{\pi^t}(s,a))^2$ **might be arbitrarily big**

To make API to make monotonic improvement, we need a strong coverage assumption:

A strong Concentrability Coefficient: $\max_{\pi} \max_{s} \frac{d_{\mu_0}^{\pi}(s)}{\mu_0(s)} \leq C < \infty$

If $C < \infty$, i.e., μ covers **all** $d_{\mu_0}^{\pi}$,

then we can expect $\widehat{Q}^t$ can approximate Q^{π^t} almost everywhere

Outline for Today

1. API could fail to make improvement?

2. When does API could make steady improvement?
(Next a few lectures, we will talk about **incremental** algorithms that **forces** π^{t+1} **to be close to** π^t

3. Performance Difference Lemma (Another important lemma)

Motivation (or the key question) behind the Performance Difference Lemma (PDL)

Let's recall simulation lemma, given two MDPs, $\widehat{P}, P$, and a policy π,

$$\left| \widehat{V}^{\pi}(s_0) - V^{\pi}(s_0) \right| \leq \frac{\gamma}{(1-\gamma)^2} \mathbb{E}_{s,a \sim d^{\pi}_{s_0}} \left\| \widehat{P}(s,a) - P(s,a) \right\|_1$$

i.e., we can upper bound value difference by model disagreement (average over real traces)

Motivation (or the key question) behind the Performance Difference Lemma (PDL)

Let's recall simulation lemma, given two MDPs, $\widehat{P}$, P, and a policy π,

$$\left| \widehat{V}^{\pi}(s_0) - V^{\pi}(s_0) \right| \leq \frac{\gamma}{(1-\gamma)^2} \mathbb{E}_{s,a \sim d^{\pi}_{s_0}} \left\| \widehat{P}(s,a) - P(s,a) \right\|_1$$

i.e., we can upper bound value difference by model disagreement (average over real traces)

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

Motivation (or the key question) behind the Performance Difference Lemma (PDL)

Let's recall simulation lemma, given two MDPs, $\widehat{P}, P$, and a policy π,

$$\left| \widehat{V}^{\pi}(s_0) - V^{\pi}(s_0) \right| \leq \frac{\gamma}{(1-\gamma)^2} \mathbb{E}_{s,a \sim d^{\pi}_{s_0}} \left\| \widehat{P}(s,a) - P(s,a) \right\|_1$$

i.e., we can upper bound value difference by model disagreement (average over real traces)

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

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

Setting and Notation

Discounted infinite horizon MDP:

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

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

Setting and Notation

Discounted infinite horizon MDP:

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

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

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

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

Setting and Notation

Discounted infinite horizon MDP:

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

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

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

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

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

Setting and Notation

Discounted infinite horizon MDP:

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

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

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

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

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

Recall PI:

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

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

PDL:

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

PDL:

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

Performance Difference Lemma (PDL):

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

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

PDL Explanation

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

PDL Proof

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

Proof of Sketch (see reading material for detailed steps)

PDL Proof

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

Proof of Sketch (see reading material for detailed steps)

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

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

PDL Proof

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

Proof of Sketch (see reading material for detailed steps)

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

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

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

PDL Proof

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

Proof of Sketch (see reading material for detailed steps)

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

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

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

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

PDL Proof

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

Proof of Sketch (see reading material for detailed steps)

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

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

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

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

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

Summary of PDL:

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

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

Summary of PDL:

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

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

(Use the fact that $Q^{\pi}(s,a) \in [0,1/(1-\gamma)]$)

Summary of PDL:

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

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

(Use the fact that $Q^{\pi}(s,a) \in [0,1/(1-\gamma)]$)

$$\left|V^{\pi}(s_0) - V^{\pi'}(s_0)\right| \leq \frac{1}{(1-\gamma)^2}\mathbb{E}_{s\sim d^{\pi}_{s_0}}\left[\left\| \pi(\cdot \mid s) - \pi'(\cdot \mid s) \right\|_1\right]$$

Summary of PDL:

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

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

(Use the fact that $Q^{\pi}(s,a) \in [0,1/(1-\gamma)]$)

$$\left|V^{\pi}(s_0) - V^{\pi'}(s_0)\right| \leq \frac{1}{(1-\gamma)^2}\mathbb{E}_{s\sim d^{\pi}_{s_0}}\left[\left\| \pi(\cdot\,|\,s) - \pi'(\cdot\,|\,s) \right\|_1\right]$$

Policy disagreement (ℓ_1) averaged over one policy's traces

An Application of PDL in Policy Iteration

Recall that $\pi^{t+1}(s) = \arg\max_a A^{\pi^t}(s, a)$

Show monotonic improvement using PDL:

An Application of PDL in Policy Iteration

Recall that $\pi^{t+1}(s) = \arg\max_a A^{\pi^t}(s, a)$

Show monotonic improvement using PDL:

$$V^{\pi^{t+1}}(s_0) - V^{\pi^t}(s_0) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi^{t+1}}}\mathbb{E}_{a\sim\pi^{t+1}(\cdot|s)}A^{\pi^t}(s, a)$$

An Application of PDL in Policy Iteration

Recall that $\pi^{t+1}(s) = \arg\max_a A^{\pi^t}(s, a)$

Show monotonic improvement using PDL:

$$V^{\pi^{t+1}}(s_0) - V^{\pi^t}(s_0) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi^{t+1}}}\mathbb{E}_{a\sim\pi^{t+1}(\cdot|s)}A^{\pi^t}(s, a)$$

$$= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi^{t+1}}}A^{\pi^t}(s, \pi^{t+1}(s))$$

An Application of PDL in Policy Iteration

Recall that $\pi^{t+1}(s) = \arg\max_a A^{\pi^t}(s, a)$

Show monotonic improvement using PDL:

$$V^{\pi^{t+1}}(s_0) - V^{\pi^t}(s_0) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi^{t+1}}}\mathbb{E}_{a\sim\pi^{t+1}(\cdot|s)}A^{\pi^t}(s, a)$$

$$= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi^{t+1}}}A^{\pi^t}(s, \pi^{t+1}(s))$$

$$\geq \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi^{t+1}}}A^{\pi^t}(s, \pi^t(s))$$

An Application of PDL in Policy Iteration

Recall that $\pi^{t+1}(s) = \arg\max_a A^{\pi^t}(s, a)$

Show monotonic improvement using PDL:

$$V^{\pi^{t+1}}(s_0) - V^{\pi^t}(s_0) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi^{t+1}}}\mathbb{E}_{a\sim\pi^{t+1}(\cdot|s)}A^{\pi^t}(s, a)$$

$$= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi^{t+1}}}A^{\pi^t}(s, \pi^{t+1}(s))$$

$$\geq \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{s_0}^{\pi^{t+1}}}A^{\pi^t}(s, \pi^t(s)) \quad = 0$$

Summary for the recent 3 lectures:

Three fundamental ingredients in RL and MDPs:

Two Algorithms:

Summary for the recent 3 lectures:

Three fundamental ingredients in RL and MDPs:

1. Simulation Lemma (concerns the perf difference of π under two MDPs)

Two Algorithms:

Summary for the recent 3 lectures:

Three fundamental ingredients in RL and MDPs:

1. Simulation Lemma (concerns the perf difference of π under two MDPs)
2. PDL (concert the perf diff of π & π' under one MDP)

Two Algorithms:

Summary for the recent 3 lectures:

Three fundamental ingredients in RL and MDPs:

1. Simulation Lemma (concerns the perf difference of π under two MDPs)
2. PDL (concert the perf diff of π & π' under one MDP)
3. How to draw samples from $d^{\pi}_{\mu_0}$, and how to get unbiased estimate of $Q^{\pi}(s,a)$

Two Algorithms:

Summary for the recent 3 lectures:

Three fundamental ingredients in RL and MDPs:

1. Simulation Lemma (concerns the perf difference of π under two MDPs)
2. PDL (concert the perf diff of π & π' under one MDP)
3. How to draw samples from $d^{\pi}_{\mu_0}$, and how to get unbiased estimate of $Q^{\pi}(s, a)$

Two Algorithms:

1. Model-based RL w/ Generative model: fit $\widehat{P}$ (by counting) and run Policy-Iter on $(\widehat{P}, r)$

Summary for the recent 3 lectures:

Three fundamental ingredients in RL and MDPs:

1. Simulation Lemma (concerns the perf difference of π under two MDPs)
2. PDL (concert the perf diff of π & π' under one MDP)
3. How to draw samples from $d^{\pi}_{\mu_0}$, and how to get unbiased estimate of $Q^{\pi}(s, a)$

Two Algorithms:

1. Model-based RL w/ Generative model: fit $\widehat{P}$ (by counting) and run Policy-Iter on $(\widehat{P}, r)$
2. Approximate Policy Iteration (Alg that uses a Regression oracle)

Next Week:

We will talk about Incremental Policy Optimization
(Recall the failure case of API; we will force incremental update on policies)