Advanced Policy Gradients

CS 294-112: Deep Reinforcement Learning

Sergey Levine

Class Notes

1. Homework 2 due today (11:59 pm)!
   - Don’t be late!
2. Final project proposal due in one week!
   - See submission instructions in project proposal assignment

Today's Lecture

1. Why does policy gradient work?
2. Policy gradient is a type of policy iteration
3. Policy gradient as a constrained optimization
4. From constrained optimization to natural gradient
5. Natural gradients and trust regions

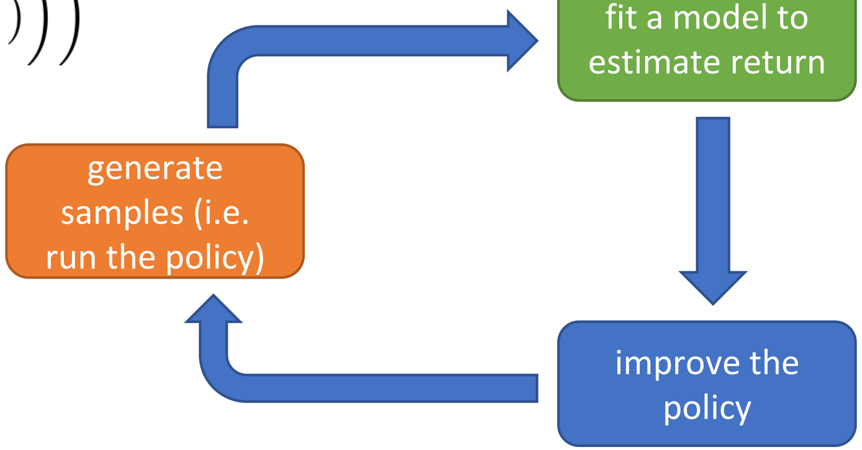
- Goals:
  - Understand the policy iteration view of policy gradient
  - Understand how to analyze policy gradient improvement
  - Understand what natural gradient does and how to use it

Recap: policy gradients

REINFORCE algorithm:

1. sample $\{\tau^i\}$ from $\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$ (run the policy)
2. $\nabla_\theta J(\theta) \approx \sum_i \left(\sum_{t=1}^T \nabla_\theta \log \pi_\theta(\mathbf{a}_t^i|\mathbf{s}_t^i) \left(\sum_{t'=t}^T r(\mathbf{s}_{t'}^i, \mathbf{a}_{t'}^i)\right)\right)$
3. $\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta)$

$$\hat{Q}^\pi(\mathbf{x}_t, \mathbf{u}_t) = \sum_{t'=t}^{T} r(\mathbf{x}_{t'}, \mathbf{u}_{t'})$$



$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta)$$

$$\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(\mathbf{a}_{i,t}|\mathbf{s}_{i,t}) \hat{Q}_{i,t}^\pi$$

"reward to go"

can also use function approximation here

Why does policy gradient work?

$$\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(\mathbf{a}_{i,t}|\mathbf{s}_{i,t}) \hat{A}^\pi_{i,t}$$

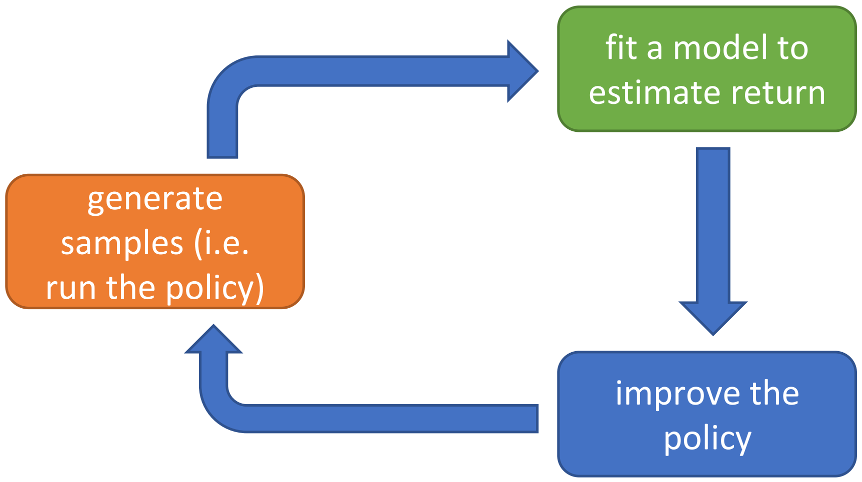
1. Estimate $\hat{A}^\pi(\mathbf{s}_t, \mathbf{a}_t)$ for current policy π
2. Use $\hat{A}^\pi(\mathbf{s}_t, \mathbf{a}_t)$ to get *improved* policy π'

look familiar?

policy iteration algorithm:

1. evaluate $A^\pi(\mathbf{s}, \mathbf{a})$
2. set $\pi \leftarrow \pi'$

$\hat{A}^\pi(\mathbf{x}_t, \mathbf{u}_t)$



$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta)$

Policy gradient as **policy iteration**

$$J(\theta) = E_{\tau \sim p_\theta(\tau)}\left[\sum_t \gamma^t r(\mathbf{s}_t, \mathbf{a}_t)\right]$$

$$\text{claim:} J(\theta') - J(\theta) = E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_t \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t)\right]$$

$$\begin{aligned}
J(\theta') - J(\theta) &= J(\theta') - E_{\mathbf{s}_0 \sim p(\mathbf{s}_1)}\left[V^{\pi_\theta}(\mathbf{s}_0)\right] \\
&= J(\theta') - E_{\tau \sim p_{\theta'}(\tau)}\left[V^{\pi_\theta}(\mathbf{s}_0)\right] \\
&= J(\theta') - E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty} \gamma^t V^{\pi_\theta}(\mathbf{s}_t) - \sum_{t=1}^{\infty} \gamma^t V^{\pi_\theta}(\mathbf{s}_t)\right] \\
&= J(\theta') + E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty} \gamma^t (\gamma V^{\pi_\theta}(\mathbf{s}_{t+1}) - V^{\pi_\theta}(\mathbf{s}_t))\right] \\
&= E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_{t=1}^{\infty} \gamma^t r(\mathbf{s}_t, \mathbf{a}_t)\right] + E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty} \gamma^t (\gamma V^{\pi_\theta}(\mathbf{s}_{t+1}) - V^{\pi_\theta}(\mathbf{s}_t))\right] \\
&= E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty} \gamma^t (r(\mathbf{s}_t, \mathbf{a}_t) + \gamma V^{\pi_\theta}(\mathbf{s}_{t+1}) - V^{\pi_\theta}(\mathbf{s}_t))\right] \\
&= E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t)\right]
\end{aligned}$$

Policy gradient as **policy iteration**

$$J(\theta') - J(\theta) = E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_t \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t)\right]$$

expectation under $\pi_{\theta'}$

advantage under π_θ

importance sampling

$$E_{x \sim p(x)}[f(x)] = \int p(x) f(x) dx = \int \frac{q(x)}{q(x)} p(x) f(x) dx = \int q(x) \frac{p(x)}{q(x)} f(x) dx = E_{x \sim q(x)}\left[\frac{p(x)}{q(x)} f(x)\right]$$

$$E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_t \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t)\right] = \sum_t E_{\mathbf{s}_t \sim p_{\theta'}(\mathbf{s}_t)}\left[E_{\mathbf{a}_t \sim \pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}\left[\gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t)\right]\right]$$

$$= \sum_t E_{\mathbf{s}_t \sim p_{\theta'}(\mathbf{s}_t)}\left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)}\left[\frac{\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t)\right]\right]$$

is it OK to use $p_\theta(\mathbf{s}_t)$ instead?

Ignoring distribution mismatch?

$$\sum_t E_{\mathbf{s}_t \sim p_{\theta'}(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right] \overset{?}{\approx} \underbrace{\sum_t E_{\mathbf{s}_t \sim p_\theta(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right]}_{\bar{A}(\theta')}$$

why do we want this to be true?

$$J(\theta') - J(\theta) \approx \bar{A}(\theta') \quad \Rightarrow \quad \theta' \leftarrow \arg\max_{\theta'} \bar{A}(\theta)$$

2. Use $\hat{A}^\pi(\mathbf{s}_t, \mathbf{a}_t)$ to get *improved* policy π'

is it true? and when?

Claim: $p_\theta(\mathbf{s}_t)$ is *close* to $p_{\theta'}(\mathbf{s}_t)$ when π_θ is *close* to $\pi_{\theta'}$

Bounding the distribution change

Claim: $p_\theta(\mathbf{s}_t)$ is *close* to $p_{\theta'}(\mathbf{s}_t)$ when π_θ is *close* to $\pi_{\theta'}$

Simple case: assume π_θ is a *deterministic* policy $\mathbf{a}_t = \pi_\theta(\mathbf{s}_t)$

$\pi_{\theta'}$ is *close* to π_θ if $\pi_{\theta'}(\mathbf{a}_t \neq \pi_\theta(\mathbf{s}_t)|\mathbf{s}_t) \leq \epsilon$

$$p_{\theta'}(\mathbf{s}_t) = \underbrace{(1-\epsilon)^t}_{\text{probability we made no mistakes}} p_\theta(\mathbf{s}_t) + (1-(1-\epsilon)^t)\underbrace{p_{\text{mistake}}(\mathbf{s}_t)}_{\text{some \textit{other} distribution}}$$

seem familiar?

$$|p_{\theta'}(\mathbf{s}_t) - p_\theta(\mathbf{s}_t)| = (1-(1-\epsilon)^t)|p_{\text{mistake}}(\mathbf{s}_t) - p_\theta(\mathbf{s}_t)| \leq 2(1-(1-\epsilon)^t)$$

useful identity: $(1-\epsilon)^t \geq 1 - \epsilon t$ for $\epsilon \in [0,1]$

$$\leq 2\epsilon t$$

not a great bound, but a bound!

Bounding the distribution change

Claim: $p_\theta(\mathbf{s}_t)$ is *close* to $p_{\theta'}(\mathbf{s}_t)$ when π_θ is *close* to $\pi_{\theta'}$

General case: assume π_θ is an arbitrary distribution

$\pi_{\theta'}$ is *close* to π_θ if $|\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) - \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)| \leq \epsilon$ for all $\mathbf{s}_t$

Useful lemma: if $|p_X(x) - p_Y(x)| = \epsilon$, exists $p(x, y)$ such that $p(x) = p_X(x)$ and $p(y) = p_Y(y)$ and $p(x = y) = \epsilon$

$\Rightarrow p_X(x)$ "agrees" with $p_Y(y)$ with probability ϵ

$\Rightarrow \pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)$ takes a different action than $\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$ with probability at most ϵ

$$
\begin{aligned}
|p_{\theta'}(\mathbf{s}_t) - p_\theta(\mathbf{s}_t)| = (1 - (1-\epsilon)^t)|p_{\text{mistake}}(\mathbf{s}_t) - p_\theta(\mathbf{s}_t)| &\leq 2(1 - (1-\epsilon)^t) \\
&\leq 2\epsilon t
\end{aligned}
$$

Proof based on: Schulman, Levine, Moritz, Jordan, Abbeel. "Trust Region Policy Optimization."

Bounding the objective value

$\pi_{\theta'}$ is *close* to π_θ if $|\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) - \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)| \leq \epsilon$ for all $\mathbf{s}_t$

$$|p_{\theta'}(\mathbf{s}_t) - p_\theta(\mathbf{s}_t)| \leq 2\epsilon t$$

$$E_{p_{\theta'}(\mathbf{s}_t)}[f(\mathbf{s}_t)] = \sum_{\mathbf{s}_t} p_{\theta'}(\mathbf{s}_t) f(\mathbf{s}_t) \geq \sum_{\mathbf{s}_t} p_\theta(\mathbf{s}_t) f(\mathbf{s}_t) - |p_{\theta'}(\mathbf{s}_t) - p_\theta(\mathbf{s}_t)| \max_{\mathbf{s}_t} f(\mathbf{s}_t)$$

$$\geq E_{p_\theta(\mathbf{s}_t)}[f(\mathbf{s}_t)] - 2\epsilon t \max_{\mathbf{s}_t} f(\mathbf{s}_t)$$

$$\sum_t E_{\mathbf{s}_t \sim p_{\theta'}(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right] \geq$$

$$\sum_t E_{\mathbf{s}_t \sim p_\theta(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right] - \sum_t 2\epsilon t C$$

$O(Tr_{\max})$ or $O\left(\frac{r_{\max}}{1-\gamma}\right)$

maximizing this maximizes a bound on the thing we want!

A more convenient bound

Claim: $p_\theta(\mathbf{s}_t)$ is *close* to $p_{\theta'}(\mathbf{s}_t)$ when π_θ is *close* to $\pi_{\theta'}$

$\pi_{\theta'}$ is *close* to π_θ if $|\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) - \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)| \leq \epsilon$ for all $\mathbf{s}_t$

$$|p_{\theta'}(\mathbf{s}_t) - p_\theta(\mathbf{s}_t)| \leq 2\epsilon t$$

a more convenient bound: $|\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) - \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)| \leq \sqrt{\frac{1}{2} D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t|\mathbf{s}_t))}$

$\Rightarrow$ $D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t|\mathbf{s}_t))$ bounds state marginal difference

$$D_{\text{KL}}(p_1(x) \| p_2(x)) = E_{x \sim p_1(x)} \left[\log \frac{p_1(x)}{p_2(x)} \right]$$

KL divergence has some very convenient properties that make it much easier to approximate!

How do we optimize the objective?

$$\theta' \leftarrow \arg\max_{\theta'} \sum_t E_{\mathbf{s}_t \sim p_\theta(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right]$$

$$\text{such that } D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) \leq \epsilon$$

for small enough ϵ, this is guaranteed to improve $J(\theta') - J(\theta)$

How do we enforce the constraint?

$$\theta' \leftarrow \arg\max_{\theta'} \sum_t E_{\mathbf{s}_t \sim p_\theta(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right]$$

such that $D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) \leq \epsilon$

$$\mathcal{L}(\theta', \lambda) = \sum_t E_{\mathbf{s}_t \sim p_\theta(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right] - \lambda(D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) - \epsilon)$$

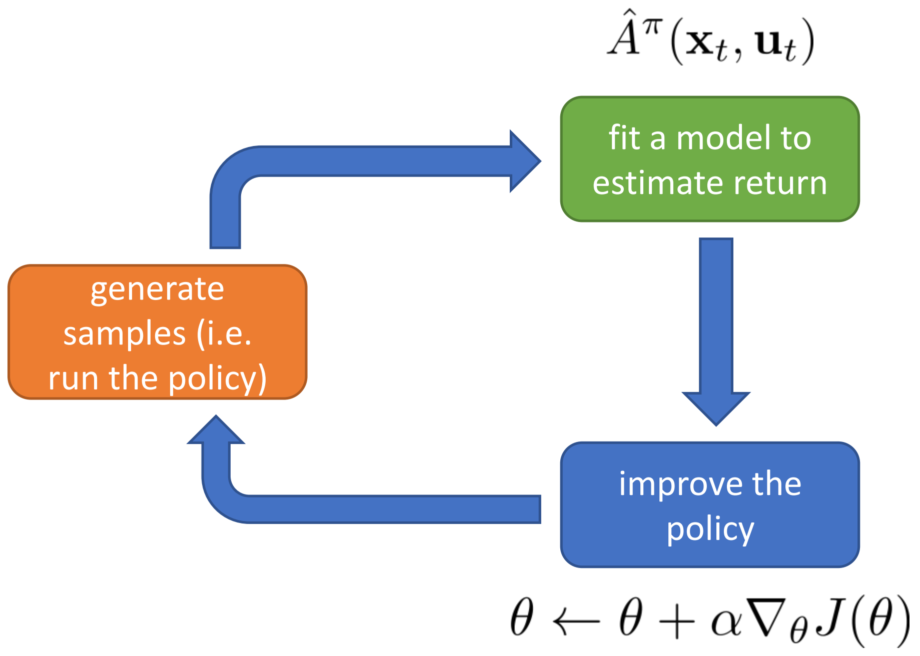
1. Maximize $\mathcal{L}(\theta', \lambda)$ with respect to θ' ← **can do this incompletely (for a few grad steps)**
2. $\lambda \leftarrow \lambda + \alpha(D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) - \epsilon)$

Intuition: raise λ if constraint violated too much, else lower it

an instance of *dual gradient descent* (more on this later!)

Review

- Policy gradient = policy iteration
  - Evaluate advantage of old policy
  - Maximize advantage w.r.t. new policy
- Correct thing to do is optimize expected advantage under new policy state distribution
- Doing this under old policy state distribution optimizes a bound, *if* the policies are close enough
- Results in *constrained* optimization problem



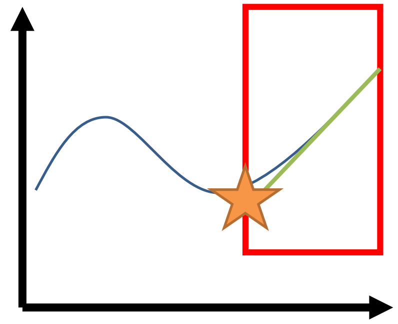
Break

How do we optimize the objective?

$$\theta' \leftarrow \arg\max_{\theta'} \underbrace{\sum_t E_{\mathbf{s}_t \sim p_\theta(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right]}_{\bar{A}(\theta')}$$

$$\text{such that } D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) \leq \epsilon$$

for small enough ϵ, this is guaranteed to improve $J(\theta') - J(\theta)$

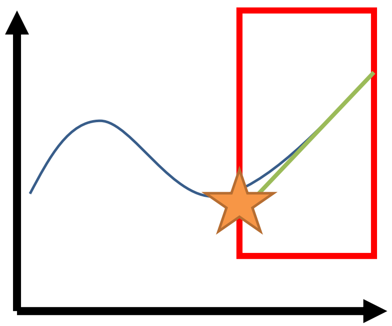


$$\theta' \leftarrow \arg\max_{\theta'} \nabla_\theta \bar{A}(\theta)^T (\theta' - \theta)$$

$$\text{such that } D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) \leq \epsilon$$

Use first order Taylor approximation for objective (a.k.a., linearization)

How do we optimize the objective?



$$\theta' \leftarrow \arg\max_{\theta'} \sum_t E_{\mathbf{s}_t \sim p_\theta(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right]$$

$$\text{such that } D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) \leq \epsilon$$

$$\theta' \leftarrow \arg\max_{\theta'} \nabla_\theta \bar{A}(\theta)^T (\theta' - \theta)$$

$$\text{such that } D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) \leq \epsilon$$

$$\nabla_{\theta'} \bar{A}(\theta') = \sum_t E_{\mathbf{s}_t \sim p_\theta(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t \nabla_{\theta'} \log \pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right]$$

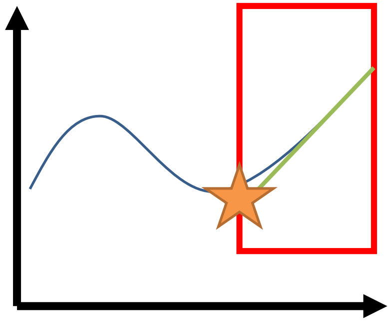
(see policy gradient lecture for derivation)

$$\nabla_\theta \bar{A}(\theta) = \sum_t E_{\mathbf{s}_t \sim p_\theta(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \gamma^t \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t) A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right]$$

$$\nabla_\theta \bar{A}(\theta) = \sum_t E_{\mathbf{s}_t \sim p_\theta(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\gamma^t \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t) A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) \right] \right] = \nabla_\theta J(\theta)$$

exactly the normal policy gradient!

Can we just use the gradient then?

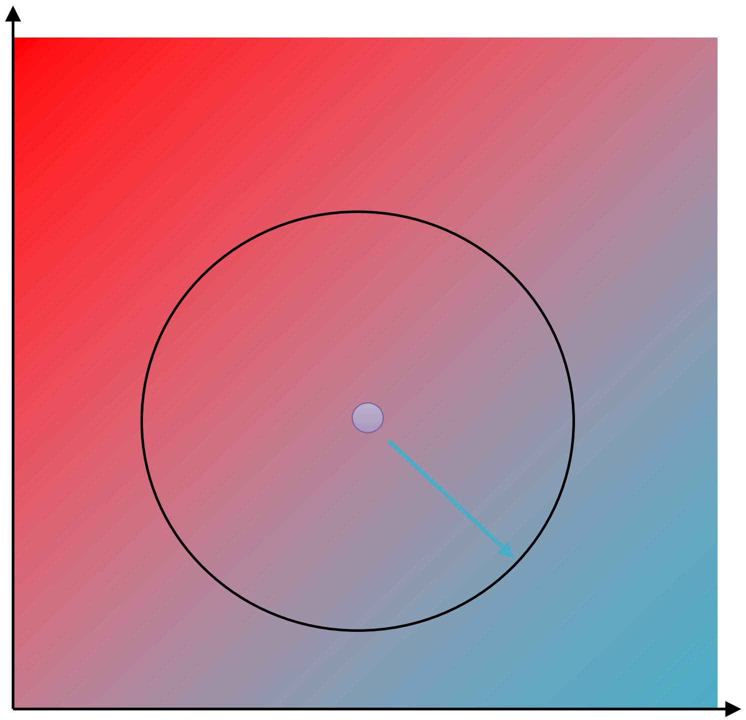


$$\theta' \leftarrow \arg\max_{\theta'} \nabla_\theta J(\theta)^T (\theta' - \theta)$$

$$\text{such that } D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) \leq \epsilon$$

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta)$$

$$\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$$



some parameters change probabilities a lot more than others!

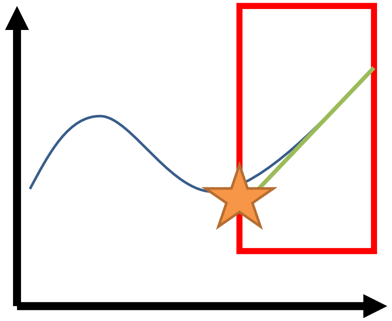
Claim: gradient ascent does this:

$$\theta' \leftarrow \arg\max_{\theta'} \nabla_\theta J(\theta)^T (\theta' - \theta)$$

$$\text{such that } \|\theta - \theta'\|^2 \leq \epsilon$$

$$\theta' = \theta + \frac{\epsilon}{\|\nabla_\theta J(\theta)\|^2} \nabla_\theta J(\theta)$$

Can we just use the gradient then?



$$\theta' \leftarrow \arg\max_{\theta'} \nabla_\theta J(\theta)^T(\theta' - \theta)$$

$$\text{such that } D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)\|\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) \leq \epsilon$$

not the same!

$$\theta' \leftarrow \arg\max_{\theta'} \nabla_\theta J(\theta)^T(\theta' - \theta)$$

$$\text{such that } \|\theta - \theta'\|^2 \leq \epsilon$$

second order Taylor expansion

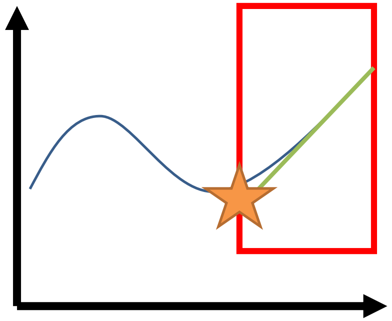
$$D_{\text{KL}}(\pi_{\theta'}\|\pi_\theta) \approx \frac{1}{2}(\theta' - \theta)^T\mathbf{F}(\theta' - \theta)$$

Fisher-information matrix

$$\mathbf{F} = E_{\pi_\theta}[\nabla_\theta \log \pi_\theta(\mathbf{a}|\mathbf{s})\nabla_\theta \log \pi_\theta(\mathbf{a}|\mathbf{s})^T]$$

can estimate with samples

Can we just use the gradient then?



$$\theta' \leftarrow \arg\max_{\theta'} \nabla_\theta J(\theta)^T(\theta' - \theta)$$

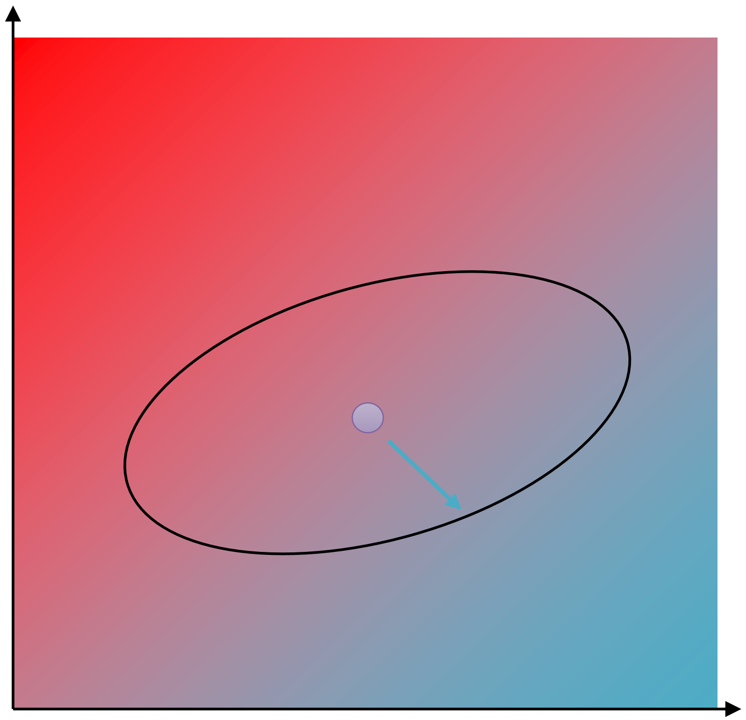
$$\text{such that } D_{\text{KL}}(\pi_{\theta'}(\mathbf{a}_t|\mathbf{s}_t)\|\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) \leq \epsilon$$

$$D_{\text{KL}}(\pi_{\theta'}\|\pi_\theta) \approx \frac{1}{2}(\theta' - \theta)^T\mathbf{F}(\theta' - \theta)$$

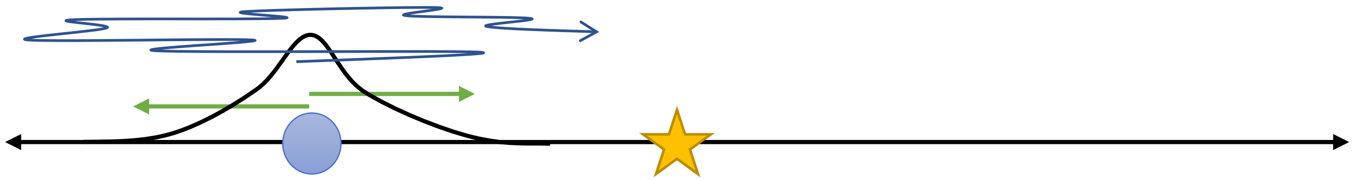
$$\theta' = \theta + \alpha\mathbf{F}^{-1}\nabla_\theta J(\theta)$$

natural gradient

$$\alpha = \sqrt{\frac{2\epsilon}{\nabla_\theta J(\theta)^T\mathbf{F}\nabla_\theta J(\theta)}}$$

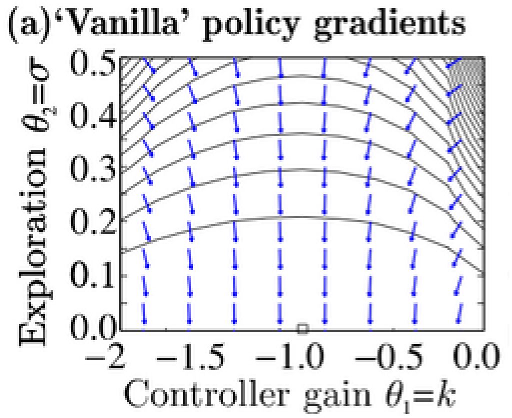


Is this even a problem in practice?

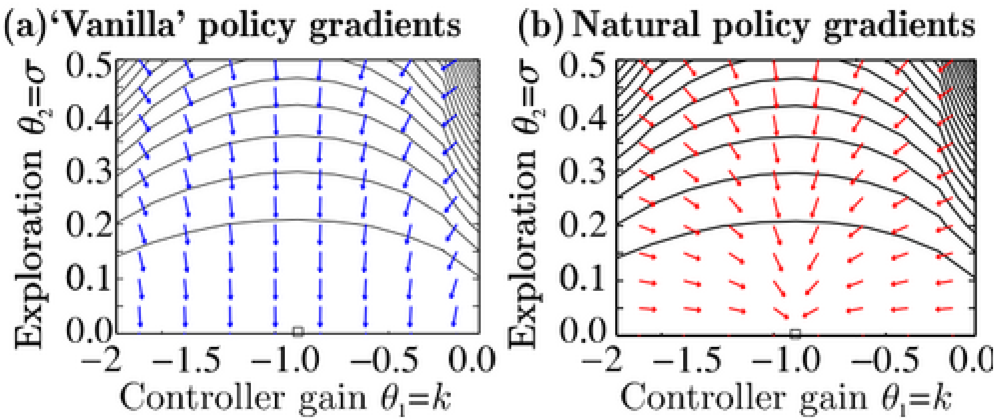


$$r(\mathbf{s}_t, \mathbf{a}_t) = -\mathbf{s}_t^2 - \mathbf{a}_t^2$$

$$\log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t) = -\frac{1}{2\sigma^2}(k\mathbf{s}_t - \mathbf{a}_t)^2 + \text{const} \qquad \theta = (k, \sigma)$$

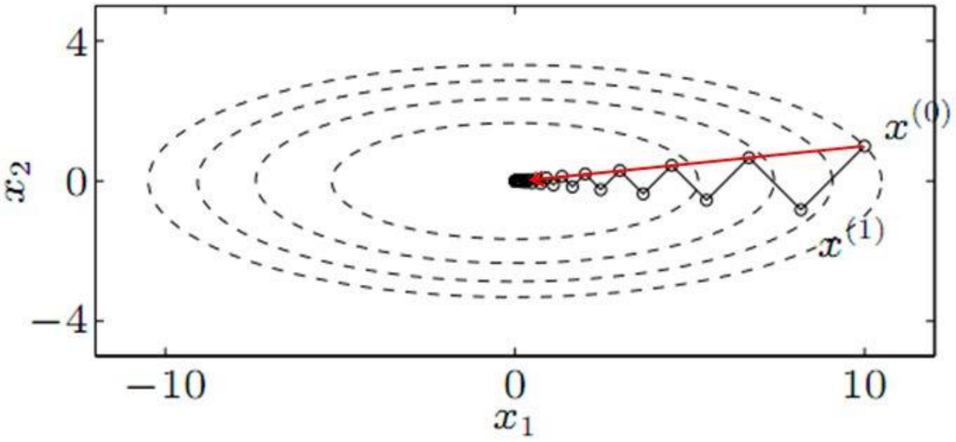


(image from Peters & Schaal 2008)



(figure from Peters & Schaal 2008)

Essentially the same problem as this:



Practical methods and notes

- Natural policy gradient

  $$\theta' = \theta + \alpha \mathbf{F}^{-1} \nabla_\theta J(\theta)$$

  - Generally a good choice to stabilize policy gradient training
  - See this paper for details:
    - Peters, Schaal. Reinforcement learning of motor skills with policy gradients.
  - Practical implementation: requires efficient Fisher-vector products, a bit non-trivial to do without computing the full matrix
    - See: Schulman et al. Trust region policy optimization
- Trust region policy optimization

  $$\alpha = \sqrt{\frac{2\epsilon}{\nabla_\theta J(\theta)^T \mathbf{F} \nabla_\theta J(\theta)}}$$

- Just use the IS objective directly
  - Use regularization to stay close to old policy
  - See: Proximal policy optimization

Review

- First order approximation to objective = gradient ascent
- Regular gradient ascent has the wrong constraint
- Taylor expansion of KL-divergence = natural gradient
- Practical algorithms
  - Natural policy gradient
  - Trust region policy optimization