Deep Reinforcement Learning

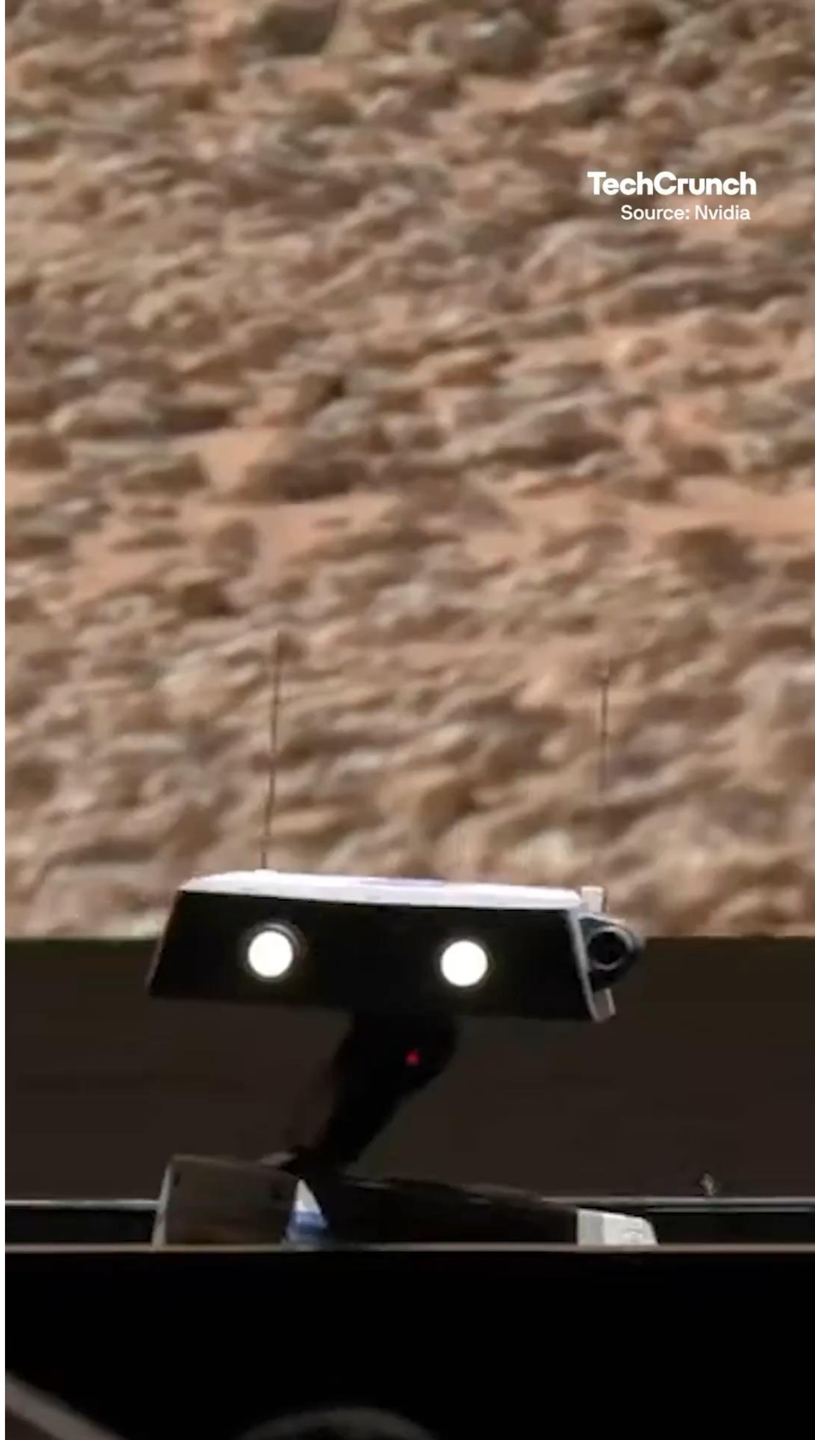
Lecture 6: Advanced RL Algorithms

Huazhe Xu

Tsinghua University

AI This Week


GTC
GPU
TECHNOLOGY
CONFERENCE
2024


TechCrunch
Source: Nvidia

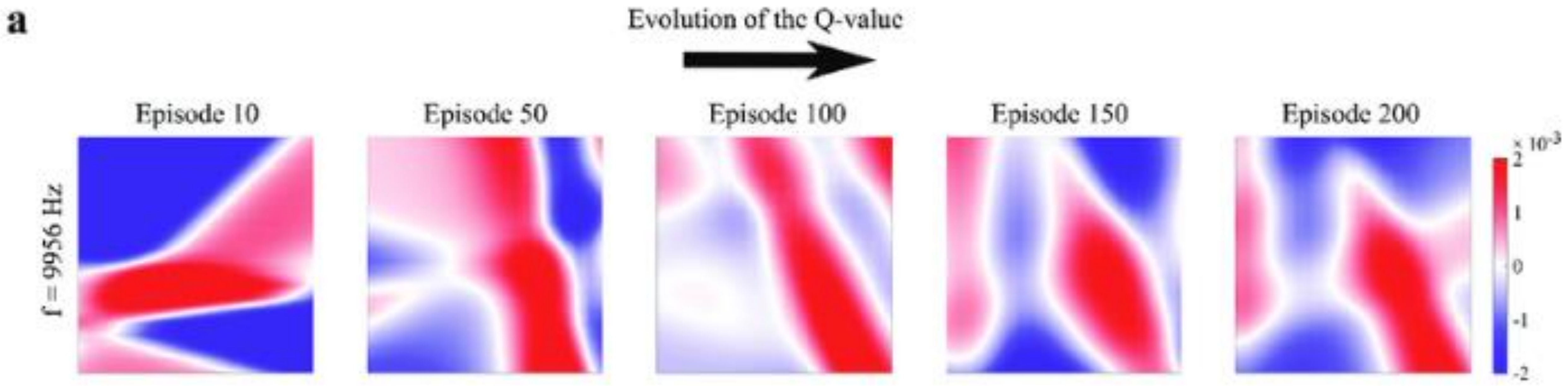
In Lec6

1. Twin Delayed DDPG
2. Proximal Policy Optimization

In Lec6

1 Twin Delayed DDPG

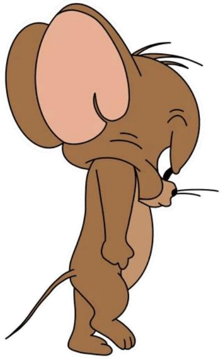
2 Proximal Policy Optimization



An example of Q value.

DDPG can be improved

- It suffers similar problems as in DQN
  - Overestimation
- The critics might be unstable.

- The critic weirdly prefers some action but not their neighbor actions. Strange landscape.

Twin Delayed DDPG (TD3)

- Solution 1: Clipped Double Q Learning
  - Toward addressing overestimation
  - We have two Qs. And we calculate min of them as my Q.

$$y(r, s', d) = r + \gamma(1 - d) \min_{i=1,2} Q_{\phi_{i,\text{targ}}}(s', a_{\text{TD3}}(s'))$$

  - Why don't we use original double Q?

Fujimoto et al. ICML 2018

Twin Delayed DDPG (TD3)

- Solution 2: Delayed Policy Updates
  - Toward addressing unstable critics
  - Lower the frequency of actor updates.
  - For every N updates in critics, we update policy once.

Fujimoto et al. ICML 2018

Twin Delayed DDPG (TD3)

- Solution 3: Target Policy Smoothing
  - Toward addressing strange landscape
  - Add noise to the actions to smooth the value

$$a_{\mathrm{TD3}}(s') = \mathbf{clip}\left(\mu_{\theta,\mathrm{targ}}(s') + \mathbf{clip}(\epsilon, -c, c), a_{\mathrm{low}}, a_{\mathrm{high}}\right)$$

Fujimoto et al. ICML 2018

In Lec6

1 Twin Delayed DDPG

2 Proximal Policy Optimization

What does a good policy-based method look like?

- Its objective is the same or similar to policy gradient.
- The difference between the old policy and the updated policy should be small enough.
- It can use history to perform multiple updates.

Proximal Policy Optimization (PPO)

$$\underset{\theta}{\textbf{maximize}} \quad \hat{\mathbb{E}}_t \left[\boxed{}\boxed{} \right] - \boxed{}$$

- We guessed. But PPO is actually an simplified version of trust-region policy optimization (TRPO).
- You might have a series of questions.
    - Why importance sampling?
    - Why advantage?
    - Why KL divergence?
- TRPO is math heavy. We will learn it soon.

Proximal Policy Optimization (PPO), which perform comparably or better than state-of-the-art approaches while being much simpler to implement and tune.

-- OpenAI

If you do not know KL Divergence.

- Kullback-Leibler Divergence is a measure of how one probability distribution *P* is different from a second, reference probability distribution *Q*. （Wikipedia）

$$D_{\mathrm{KL}}(P \| Q) = \sum_{x \in \mathcal{X}} P(x) \log\left(\frac{P(x)}{Q(x)}\right)$$

Is PPO on-policy or off-policy?

- It uses its history. Maybe it is off-policy?
- But it only uses very recent history.
- PPO is usually regarded as on policy.
- Or we may call it "on-policy-ish”

Adaptive KL Penalty

$$\underset{\theta}{\operatorname{maximize}} \quad \hat{\mathbb{E}}_t\left[\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)} \hat{A}_t\right] - \beta \hat{\mathbb{E}}_t\left[\mathrm{KL}\left[\pi_{\theta_{\text{old}}}(\cdot \mid s_t), \pi_\theta(\cdot \mid s_t)\right]\right]$$

- Parameter is hard to choose.

Compute $d = \hat{\mathbb{E}}_t\left[\mathrm{KL}\left[\pi_{\theta_{\text{old}}}(\cdot \mid s_t), \pi_\theta(\cdot \mid s_t)\right]\right]$

- If $d < d_{\text{targ}}/1.5, \beta \leftarrow \beta/2$

- If $d > d_{\text{targ}} \times 1.5, \beta \leftarrow \beta \times 2$

PPO with Clipped Objective

$$\underset{\theta}{\operatorname{maximize}} \quad \hat{\mathbb{E}}_t\left[\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)}\hat{A}_t\right] \quad r_t(\theta) = \frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)}$$

- Since we only have a soft constraints on the KL divergence.
- Fluctuation happens when the ratio is too large.

$$L^{CLIP}(\theta) = \hat{\mathbb{E}}_t\Big[\min\Big(r_t(\theta)\hat{A}_t, \operatorname{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\Big)\Big]$$

PPO in Practice

$$L_t^{CLIP+VF+S}(\theta) = \hat{\mathbb{E}}_t\left[L_t^{CLIP}(\theta) - c_1 L_t^{VF}(\theta) + c_2 S[\pi_\theta](s_t)\right]$$

a squared-error loss for "critic"

$$\left(V_\theta(s_t) - V_t^{\text{targ}}\right)^2$$

entropy bonus to ensure sufficient exploration
encourage "diversity"

Performance of PPO

algorithm	avg. normalized score
No clipping or penalty	-0.39
Clipping, $\epsilon = 0.1$	0.76
Clipping, $\epsilon = 0.2$	**0.82**
Clipping, $\epsilon = 0.3$	0.70
Adaptive KL $d_{\text{targ}} = 0.003$	0.68
Adaptive KL $d_{\text{targ}} = 0.01$	0.74
Adaptive KL $d_{\text{targ}} = 0.03$	0.71
Fixed KL, $\beta = 0.3$	0.62
Fixed KL, $\beta = 1.$	0.71
Fixed KL, $\beta = 3.$	0.72
Fixed KL, $\beta = 10.$	0.69

Choose PPO when you want to do a real project.

Proximal policy optimization algorithms [PDF] arxiv.org

J Schulman, F Wolski, P Dhariwal, A Radford… - arXiv preprint arXiv …, 2017 - arxiv.org

… We propose a new family of policy gradient methods for reinforcement learning, which alternate between sampling data through interaction with the environment, and optimizing a "…

☆ 保存 ፵ 引用 被引用次数：10939 相关文章 所有 9 个版本 »

Proximal policy optimization algorithms [PDF] arxiv.org

J Schulman, F Wolski, P Dhariwal, A Radford… - arXiv preprint arXiv …, 2017 - arxiv.org

… It shows how several objectives vary as we interpolate along the **policy** update direction, obtained by **proximal policy optimization** (the algorithm we will introduce shortly) on a …

☆ Save ፵ Cite Cited by 24700 Related articles All 11 versions »

Buckle up!

What does policy gradient do?

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

$$L^{PG}(\theta) = \hat{\mathbb{E}}_t \Big[\log \pi_\theta(a_t \mid s_t) \hat{A}_t \Big]$$

- We are actually evaluating the advantage and then improve policy based on it.

Policy Gradient is a “soft” version of Policy Iteration.

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

$$J(\theta') - J(\theta) = J(\theta') - E_{\mathbf{s}_0 \sim p(\mathbf{s}_0)}[V^{\pi_\theta}(\mathbf{s}_0)]$$

$$= J(\theta') - E_{\tau \sim p_{\theta'}(\tau)}[V^{\pi_\theta}(\mathbf{s}_0)]$$

$$= J(\theta') - E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty} \gamma^t V^{\pi_\theta}(\mathbf{s}_t) - \sum_{t=1}^{\infty} \gamma^t V^{\pi_\theta}(\mathbf{s}_t)\right]$$

$$= J(\theta') + E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty} \gamma^t (\gamma V^{\pi_\theta}(\mathbf{s}_{t+1}) - V^{\pi_\theta}(\mathbf{s}_t))\right]$$

$$= E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty} \gamma^t r(\mathbf{s}_t, \mathbf{a}_t)\right] + E_{\tau \sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty} \gamma^t (\gamma V^{\pi_\theta}(\mathbf{s}_{t+1}) - V^{\pi_\theta}(\mathbf{s}_t))\right]$$

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

This answers our question about ”Why advantage?”

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

Now we try to answer "Why importance sampling?"

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

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

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

Almost there! But the state distribution is still annoying.

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

We approximate by ignoring the difference.

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

This whole thing

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

- Now, smart as you are, you might notice what is going to appear here.

$$p_\theta(\mathbf{s}_t) \text{ is close to } p_{\theta'}(\mathbf{s}_t) \text{ when } \pi_\theta \text{ is close to } \pi_{\theta'}$$

Is this true?

$$p_\theta(\mathbf{s}_t) \text{ is close to } p_{\theta'}(\mathbf{s}_t) \text{ when } \pi_\theta \text{ is close to } \pi_{\theta'}$$

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 $\pi_{\theta'}(\mathbf{a}_t \neq \pi_\theta(\mathbf{s}_t) \mid \mathbf{s}_t) \leq \epsilon$

$$p_{\theta'}(\mathbf{s}_t) = (1-\epsilon)^t p_\theta(\mathbf{s}_t) + \left(1 - (1-\epsilon)^t\right) p_{\text{mistake}}(\mathbf{s}_t)$$

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

Useful tool $(1-\epsilon)^t \geq 1 - \epsilon t \text{ for } \epsilon \in [0,1]$

This also applies in a more general case.

General case: assume π_θ is an arbitrary distribution

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

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

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 $p(x = y) = 1 - \epsilon$

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

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

If the two distribution distance are bounded, we will have …

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

$$\begin{aligned}E_{p_{\theta'}(\mathbf{s}_t)}[f(\mathbf{s}_t)] = \sum_{\mathbf{s}_t} p_{\theta'}(\mathbf{s}_t) f(\mathbf{s}_t) &\geq \sum_{\mathbf{s}_t} p_\theta(\mathbf{s}_t) f(\mathbf{s}_t) - |p_\theta(\mathbf{s}_t) - p_{\theta'}(\mathbf{s}_t)| \max_{\mathbf{s}_t} f(\mathbf{s}_t) \\ &\geq E_{p_\theta(\mathbf{s}_t)}[f(\mathbf{s}_t)] - 2\epsilon t \max_{\mathbf{s}_t} f(\mathbf{s}_t)\end{aligned}$$

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

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

Lower bound!

A more convenient bound (and answers the question about KL divergence!)

$$|\pi_{\theta'}(\mathbf{a}_t \mid \mathbf{s}_t) - \pi_\theta(\mathbf{a}_t \mid \mathbf{s}_t)| \leq \sqrt{\frac{1}{2} D_{\mathrm{KL}}(\pi_{\theta'}(\mathbf{a}_t \mid \mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t \mid \mathbf{s}_t))}$$

Now we have something like this

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

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

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

How to optimize this other than PPO?

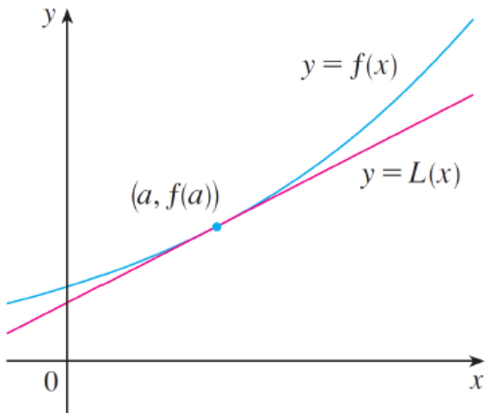
$$\theta' \leftarrow \arg\max_{\theta'} \sum_t E_{\mathbf{s}_t \sim p_\theta(\mathbf{s}_t)} \left[E_{\mathbf{a}_t \sim \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)} \left[\frac{\pi_{\theta'}(\mathbf{a}_t \mid \mathbf{s}_t)}{\pi_\theta(\mathbf{a}_t \mid \mathbf{s}_t)} \gamma^t A^{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t)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

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

Linearization

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

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



Linearization

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

How do we deal with the constraints?

- We still have KL Divergence.
- First guess, KL of policies is almost the same as constraining the parameters θ .
- Is this true?

$$D_{\mathrm{KL}}(\pi_{\theta'}(\mathbf{a}_t \mid \mathbf{s}_t) \| \pi_\theta(\mathbf{a}_t \mid \mathbf{s}_t)) \leq \epsilon$$

$$\|\theta - \theta'\|^2 \leq \epsilon$$

Our guess: Parameter constraints are equal to KL constraints

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

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

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

- Is this true?

Same idea: Taylor Expansion

- But to the KL Divergence

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

- We call **F** Fisher-information matrix.

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

- You can estimate it from samples!
- What happens if the fisher-information matrix is identity?

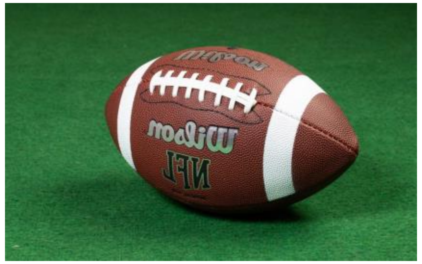
A closer look at this quadratic term

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

- Now we have some sort of sensitivity estimation in parameter space.

$$\theta' = \theta + \alpha \mathbf{F}^{-1} \nabla_\theta J(\theta) \qquad \alpha = \sqrt{\frac{2\epsilon}{\nabla_\theta J(\theta)^T \mathbf{F} \nabla_\theta J(\theta)}}$$

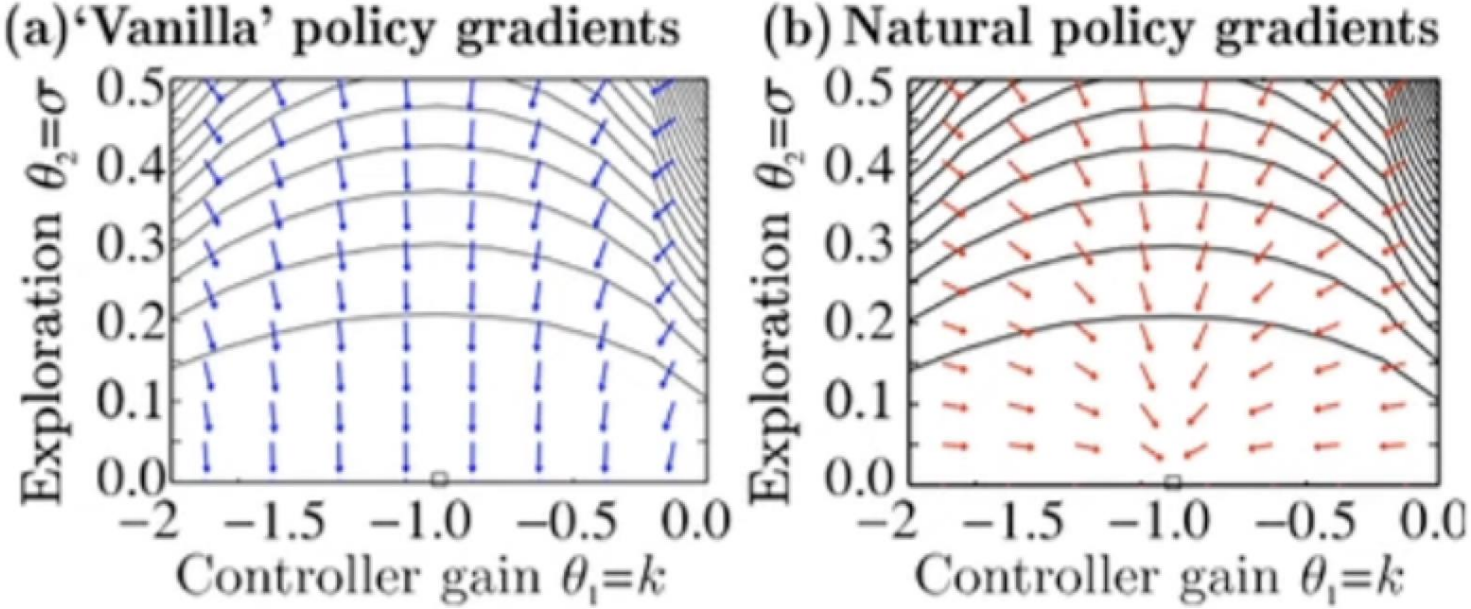
Natural gradient



Trying to make an example:

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

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



Gradient of sigma can be large in vanilla PG.

Peters & Schaal, 2008

We are almost done there!

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

- Natural Policy Gradient
  - Good choice to stabilize
  - Peters, Schaal. Reinforcement learning of motor skills with policy gradients.
- Trust Region Policy Gradient
  - Provide a practical implementation for this
  - Conjugate gradient
  - Schulman et al. Trust region policy optimization

Recap

- Value-based (Value Iteration, TD/MC, Q-learning, Explore/Exploit, DQN)
- Policy-based (PG, off-policy PG, baseline)
- Actor-Critic (Value as the baseline, async methods, off-policy methods)
- After actor-critic, the line is no longer so clear.
    - Q learning can deal with continuous action: DDPG
    - DDPG needs to be stabilized: TD3
    - Soft actor critic in recitation ☺
    - Policy gradient with a soft constraints (and a critic + entropy term): PPO
    - Policy gradient with a hard constraints: NPG -> TRPO
        - Monotonic improvement
        - Slight off-policiness is fine
        - Contrain the parameters with sensitivity weight

Now you are an RL expert ☺

- But with limited hands-on experience
- But only in the model-free world
- But only in the online world
- But only in the vector input world
- But only with a single agent
- But only with a typical world/env, what if some of your world is broken?
    - Partially observable
    - Reward very sparse
    - Action space is too large
    - the simulator is different from the real world

you

Also you

Next: Special Topics

- Model-based RL
- Visual RL and Generalization
- RL for Robotics
- LLM in RL/Robotics
- … TBD