Trust Region Policy Optimization
Announcements

Thanks for providing midterm feedback!

1. HW2 will be out this Friday

2. I will have an additional office hour every Monday morning
   (11am - noon)
Recap Policy Gradient

\[ J(\pi_\theta) = \mathbb{E} \left[ \sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \mid s_0 \sim \mu, a \sim \pi_\theta \right] \]
Recap Policy Gradient

\[ J(\pi_\theta) = \mathbb{E} \left[ \sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \mid s_0 \sim \mu, a \sim \pi_\theta \right] \]

The most commonly used formulation:

\[ \nabla_\theta J(\pi_\theta) = \mathbb{E}_{s,a \sim d^\pi_\theta} \left[ \nabla_\theta \ln \pi_\theta(a \mid s) A^\pi_\theta(s, a) \right] \]
Recap Policy Gradient

\[
J(\pi_{\theta}) = \mathbb{E} \left[ \sum_{h=0}^{\infty} \gamma^h r(s_h, a_h) \mid s_0 \sim \mu, a \sim \pi_{\theta} \right]
\]

The most commonly used formulation:

\[
\nabla_{\theta} J(\pi_{\theta}) = \mathbb{E}_{s,a \sim d_{\pi_{\theta}}} \left[ \nabla_{\theta} \ln \pi_{\theta_i}(a \mid s) A_{\pi_{\theta_i}}(s, a) \right]
\]

Algorithm: Stochastic Gradient Ascent
Recap on Conservative Policy Iteration

For $t = 0$ …

1. Greedy Policy Selector:
   $$\pi' \in \arg \max_{\pi \in \Pi} \mathbb{E}_{s \sim d^\pi_t} \left[ A^\pi'(s, \pi(s)) \right]$$

2. Incremental Update:
   $$\pi^{t+1}(\cdot|s) = (1 - \alpha)\pi^t(\cdot|s) + \alpha\pi'(\cdot|s), \forall s$$
Recap on Conservative Policy Iteration

For $t = 0$ …

1. Greedy Policy Selector:
   \[
   \pi' \in \arg\max_{\pi \in \Pi} \mathbb{E}_{s \sim d^t_{\pi}} \left[ A^{\pi'}(s, \pi(s)) \right]
   \]

2. Incremental Update:
   \[
   \pi^{t+1}(\cdot | s) = (1 - \alpha) \pi^t(\cdot | s) + \alpha \pi'(\cdot | s), \forall s
   \]

Q: Why this is incremental? In what sense?

Q: Can we get monotonic policy improvement?
Recap of CPI:

Incremental update (Lemma 12.1 in AJKS)

\[ \| d_{\mu}^{\pi_{t+1}} - d_{\mu}^{\pi_t} \|_1 \leq \frac{2\gamma \alpha}{1 - \gamma} \]
Pros and Cons of CPI:

Pros:
This is fundamental!
The idea of incremental update and the theorem behind it are still being used today…

Cons:
Practical Issue (e.g., memory issue)
e.g., what if my policies are all extremely large neural networks…
Today’s Question

Can we develop some practical version of CPI?
Outlines

1. Quick intro on KL-divergence

2. A Trust-Region Formulation for Policy Optimization

3. Algorithm: Natural Policy Gradient
Interesting videos from the today’s algorithm

Train a robot to “run” forward as fast as possible:

**State:** joint angles, center of mass, velocity, etc

**Action:** torques on joints

**Reward:** distance of moving forward between two steps
Interesting videos from the today’s algorithm

Train a robot to “run” forward as fast as possible:

**State**: joint angles, center of mass, velocity, etc

**Action**: torques on joints

**Reward**: distance of moving forward between two steps
Interesting videos from the today’s algorithm

Train a robot to “run” forward as fast as possible:
  **State**: joint angles, center of mass, velocity, etc
  **Action**: torques on joints
  **Reward**: distance of moving forward between two steps
Interesting videos from the today’s algorithm

Train a robot to “run” forward as fast as possible:

State: joint angles, center of mass, velocity, etc
Action: torques on joints
Reward: distance of moving forward between two steps
Interesting videos from the today’s algorithm

Train a robot to “run” forward as fast as possible:

**State:** joint angles, center of mass, velocity, etc

**Action:** torques on joints

**Reward:** distance of moving forward between two steps

(BTW, This reveals an issue on reward design—we will study it in Learning from Demonstrations)
KL-divergence: measures the distance between two distributions

Given two distributions $P$ & $Q$, where $P \in \Delta(X)$, $Q \in \Delta(X)$, KL Divergence is defined as:

$$KL(P \mid Q) = \mathbb{E}_{x \sim P} \left[ \ln \frac{P(x)}{Q(x)} \right]$$
KL-divergence: measures the distance between two distributions

Given two distributions $P$ & $Q$, where $P \in \Delta(X), Q \in \Delta(X)$,
KL Divergence is defined as:

$$KL(P \mid Q) = \mathbb{E}_{x \sim P} \left[ \ln \frac{P(x)}{Q(x)} \right]$$

Examples:

If $Q = P$, then $KL(P \mid Q) = KL(Q \mid P) = 0$
KL-divergence: measures the distance between two distributions

Given two distributions $P \& Q$, where $P \in \Delta(X)$, $Q \in \Delta(X)$,
KL Divergence is defined as:

$$KL(P \| Q) = \mathbb{E}_{x \sim P} \left[ \ln \frac{P(x)}{Q(x)} \right]$$

Examples:

If $Q = P$, then $KL(P \| Q) = KL(Q \| P) = 0$

If $P = \mathcal{N}(\mu_1, \sigma^2 I)$, $Q = \mathcal{N}(\mu_2, \sigma^2 I)$, then $KL(P \| Q) = \|\mu_1 - \mu_2\|^2 / 2\sigma^2$
KL-divergence: measures the distance between two distributions

Given two distributions $P$ & $Q$, where $P \in \Delta(X)$, $Q \in \Delta(X)$, KL Divergence is defined as:

$$KL(P \| Q) = \mathbb{E}_{x \sim P} \left[ \ln \frac{P(x)}{Q(x)} \right]$$

**Examples:**

If $Q = P$, then $KL(P \| Q) = KL(Q \| P) = 0$

If $P = \mathcal{N}(\mu_1, \sigma^2 I)$, $Q = \mathcal{N}(\mu_2, \sigma^2 I)$, then

$$KL(P \| Q) = \frac{||\mu_1 - \mu_2||_2^2}{\sigma^2}$$

**Fact:**

$$KL(P \| Q) \geq 0$$, and being 0 if and only if $P = Q$
Outlines

1. Quick intro on KL-divergence

2. A Trust-Region Formulation for Policy Optimization

3. Algorithm: Natural Policy Gradient
Policy Parameterization

Recall that we consider parameterized policy $\pi_\theta(\cdot | s) \in \Delta(A)$, $\forall s$

1. Softmax linear Policy
(We will try this in HW2)

Feature vector $\phi(s, a) \in \mathbb{R}^d$, and parameter $\theta \in \mathbb{R}^d$

$$
\pi_\theta(a | s) = \frac{\exp(\theta^T \phi(s, a))}{\sum_{a'} \exp(\theta^T \phi(s, a'))}
$$

2. Neural Policy:

Neural network $f_\theta : S \times A \mapsto \mathbb{R}$

$$
\pi_\theta(a | s) = \frac{\exp(f_\theta(s, a))}{\sum_{a'} \exp(f_\theta(s, a'))}
$$
A trust region formulation for policy update:

At iteration $t$, with $\pi_{\theta_t}$ at hand, we compute $\theta_{t+1}$ as follows:
A trust region formulation for policy update:

At iteration $t$, with $\pi_{\theta_t}$ at hand, we compute $\theta_{t+1}$ as follows:

$$\max_{\pi_{\theta}} \mathbb{E}_{s \sim d_{\mu_{t}}^{\pi_{\theta}}} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} A_{\pi_{\theta_t}}(s, a) \right]$$

s.t., $KL \left( \rho_{\pi_{\theta_t}} \| \rho_{\pi_{\theta}} \right) \leq \delta$
A trust region formulation for policy update:

At iteration $t$, with $\pi_{\theta_t}$ at hand, we compute $\theta_{t+1}$ as follows:

$$\max_{\pi_{\theta}} \mathbb{E}_{s \sim d_{\mu_t}} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} A_{\pi_{\theta_t}}(s, a) \right]$$

s.t., $KL \left( \rho_{\pi_{\theta_t}} \mid \rho_{\pi_{\theta}} \right) \leq \delta$

We want to maximize local advantage against $\pi_{\theta_t}$, but we want the new policy to be close to $\pi_{\theta_t}$ (in the KL sense)
A trust region formulation for policy update:

At iteration $t$, with $\pi_{\theta_t}$ at hand, we compute $\theta_{t+1}$ as follows:

$$\max_{\theta_t} \mathbb{E}_{s \sim d_{\mu_t}} \left[ \mathbb{E}_{a \sim \pi_{\theta_t}(s)} A_{\pi_{\theta_t}}(s, a) \right]$$

s.t., $KL(\rho_{\pi_{\theta_t}} | \rho_{\pi_{\theta}}) \leq \delta$

We want to maximize local advantage against $\pi_{\theta_t}$, but we want the new policy to be close to $\pi_{\theta_t}$ (in the KL sense)

How we can actually do the optimization here? After all, we don’t even know the analytical form of trajectory likelihood…
A trust region formulation for policy update:

At iteration $t$, with $\pi_{\theta_t}$ at hand, we compute $\theta_{t+1}$ as follows:

$$\max_{\pi_{\theta}} \mathbb{E}_{s \sim d_{\mu_t}^\pi} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} A_{\pi_{\theta_t}}(s, a) \right]$$

s.t., $KL \left( \rho_{\pi_{\theta_t}} \mid \rho_{\pi_{\theta}} \right) \leq \delta$

High-level strategy:
1. First-order Taylor expansion on the objective at $\theta_t$
2. Second-order Taylor expansion of the constraint at $\theta_t$
Simplify Objective Function

\[
\max_{\theta} \mathbb{E}_{s \sim d^{\pi_\theta}_t} \left[ \mathbb{E}_{a \sim \pi_\theta(s)} A^{\pi_\theta}(s, a) \right]
\]
Simplify Objective Function

$$\max_\theta \mathbb{E}_{s \sim d_\mu} \left[ \mathbb{E}_{a \sim \pi_\theta(s)} A_{\pi_\theta}(s, a) \right]$$

Since the objective is also non-linear, let's do first order taylor expansion on it:
Simplify Objective Function

$$\max_{\theta} \mathbb{E}_{s \sim d_{\mu}^{\pi_{\theta} t}} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} A^{\pi_{\theta} t}(s, a) \right]$$

Since the objective is also non-linear, let's do first order-taylor expansion on it:

$$\mathbb{E}_{s \sim d_{\mu}^{\pi_{\theta} t}} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} A^{\pi_{\theta} t}(s, a) \right] \approx \mathbb{E}_{s \sim d_{\mu}^{\pi_{\theta} t}} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} A^{\pi_{\theta} t}(s, a) \right] + \mathbb{E}_{s \sim d_{\mu}^{\pi_{\theta} t}} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} \nabla_{\theta} \ln \pi_{\theta}(a | s) A^{\pi_{\theta} t}(s, a) \right] \cdot (\theta - \theta_t)$$

$$\nabla_{\theta} J(\pi_{\theta t})$$
Simplify Objective Function

\[
\max_{\theta} \mathbb{E}_{s \sim d_{\mu}^{\pi_{\theta}}} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} A^{\pi_{\theta}}(s, a) \right]
\]

Since the objective is also non-linear, let's do first order Taylor expansion on it:

\[
\mathbb{E}_{s \sim d_{\mu}^{\pi_{\theta}}} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} A^{\pi_{\theta}}(s, a) \right] \approx \mathbb{E}_{s \sim d_{\mu}^{\pi_{\theta}}} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} A^{\pi_{\theta}}(s, a) \right] + \mathbb{E}_{s \sim d_{\mu}^{\pi_{\theta}}} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} \nabla_{\theta} \ln \pi_{\theta}(a | s) A^{\pi_{\theta}}(s, a) \right] \cdot (\theta - \theta_t)
\]

\[
= \nabla_{\theta} J(\pi_{\theta})^T (\theta - \theta_t)
\]
Simplify Constraint via second-order Taylor Expansion:
Simplify Constraint via second-order Taylor Expansion:

\[ KL(\rho_{\theta_i} | \rho_{\theta}) := \ell(\theta) \]
Simplify Constraint via second-order Taylor Expansion:

\[ KL(\rho_{\theta_t} | \rho_\theta) := \ell(\theta) \]

\[ \ell(\theta) \approx \ell(\theta_t) + \nabla \ell(\theta_t)^T (\theta - \theta_t) + \frac{1}{2} (\theta - \theta_t)^T \nabla^2_\theta \ell(\theta_t) (\theta - \theta_t) \]
Simplify Constraint via second-order Taylor Expansion:

\[ KL(\rho_{\theta_i} | \rho_\theta) := \ell(\theta) \]

\[ \ell(\theta) \approx \ell(\theta_t) + \nabla \ell(\theta_t)^\top (\theta - \theta_t) + \frac{1}{2}(\theta - \theta_t)^\top \nabla^2_\theta \ell(\theta_t)(\theta - \theta_t) \]

\[ \ell(\theta_t) = KL(\rho_{\theta_t} | \rho_{\theta_t}) = 0 \]
Simplify Constraint via second-order Taylor Expansion:

\[ KL(\rho_{\theta_i} \mid \rho_\theta) := \ell(\theta) \]

\[ \ell(\theta) \approx \ell(\theta_t) + \nabla \ell(\theta_t)^\top (\theta - \theta_t) + \frac{1}{2} (\theta - \theta_t)^\top \nabla_\theta^2 \ell(\theta_t)(\theta - \theta_t) \]

\[ \ell(\theta_t) = KL(\rho_{\theta_i} \mid \rho_{\theta_t}) = 0 \]

We will show that \( \nabla_\theta \ell(\theta_t) = 0 \), and \( \nabla^2 \ell(\theta_t) \) has a nice form!
The gradient of the KL-divergence is zero at $\theta_t$

Change from trajectory distribution to state-action distribution:
The gradient of the KL-divergence is zero at $\theta_t$

Change from trajectory distribution to state-action distribution:

$$KL \left( \rho_{\pi_{\theta_t}} \mid \rho_{\pi_0} \right) = \mathbb{E}_{\tau \sim \rho_{\pi_{\theta_t}}} \ln \frac{\rho_{\pi_{\theta_t}}(\tau)}{\rho_{\pi_0}(\tau)} = \mathbb{E}_{\tau \sim \rho_{\pi_{\theta_t}}} \sum_{h=0}^{H-1} \ln \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_{\theta}(a_h \mid s_h)}$$
The gradient of the KL-divergence is zero at $\theta_t$

Change from trajectory distribution to state-action distribution:

$$KL \left( \rho_{\pi_\theta_t} \mid \rho_{\pi_\theta} \right) = \mathbb{E}_{\tau \sim \rho_{\pi_\theta_t}} \ln \frac{\rho_{\pi_\theta_t}(\tau)}{\rho_{\pi_\theta}(\tau)} = \mathbb{E}_{\tau \sim \rho_{\pi_\theta_t}} \sum_{h=0}^{H-1} \ln \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_{\theta}(a_h \mid s_h)}$$

$$= \frac{1}{1 - \gamma} \mathbb{E}_{s_h, a_h \sim d^{\pi_{\theta_t}}} \left[ \ln \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_{\theta}(a_h \mid s_h)} \right] : = \ell(\theta)$$
The gradient of the KL-divergence is zero at $\theta_t$

Change from trajectory distribution to state-action distribution:

$$
KL \left( \rho_{\pi_{\theta_t}} \mid \rho_{\pi_{\theta}} \right) = \mathbb{E}_{\tau \sim \rho_{\pi_{\theta_t}}} \ln \frac{\rho_{\pi_{\theta_t}}(\tau)}{\rho_{\pi_{\theta}}(\tau)} = \mathbb{E}_{\tau \sim \rho_{\pi_{\theta_t}}} \sum_{h=0}^{H-1} \ln \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_{\theta}(a_h \mid s_h)}
$$

$$
= \frac{1}{1 - \gamma} \mathbb{E}_{s_h, a_h \sim d^s_{\mu}} \left[ \ln \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_{\theta}(a_h \mid s_h)} \right] := \mathcal{L}(\theta)
$$

$$
\nabla_\theta \mathcal{L}(\theta) \mid_{\theta = \theta_t} = \mathbb{E}_{s \sim d^s_{\mu}} \sum_a \pi_{\theta_t}(a \mid s) \left( - \nabla_\theta \ln \pi_{\theta}(a_h \mid s_h) \bigg|_{\theta = \theta_t} \right)
$$
The gradient of the KL-divergence is zero at $\theta_t$

Change from trajectory distribution to state-action distribution:

$$KL \left( \rho_{\pi_{\theta_t}} \mid \rho_{\pi_{\theta}} \right) = \mathbb{E}_{\tau \sim \rho_{\pi_{\theta_t}}} \ln \frac{\rho_{\pi_{\theta_t}}(\tau)}{\rho_{\pi_{\theta}}(\tau)} = \mathbb{E}_{\tau \sim \rho_{\pi_{\theta}}} \sum_{h=0}^{H-1} \ln \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_{\theta}(a_h \mid s_h)}$$

$$= \frac{1}{1 - \gamma} \mathbb{E}_{s_h,a_h \sim d_{\mu_{\theta_t}}} \left[ \ln \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_{\theta}(a_h \mid s_h)} \right] =: \mathcal{L}(\theta)$$

$$\nabla_\theta \mathcal{L}(\theta) \mid_{\theta = \theta_t} = \mathbb{E}_{s \sim d_{\mu_{\theta_t}}} \sum_a \pi_{\theta_t}(a \mid s) \left( - \nabla_\theta \ln \pi_{\theta}(a_h \mid s_h) \bigg|_{\theta = \theta_t} \right)$$

$$= - \mathbb{E}_{s \sim d_{\mu_{\theta_t}}} \sum_a \pi_{\theta_t}(a \mid s) \frac{\nabla_\theta \pi_{\theta_t}(a \mid s)}{\pi_{\theta_t}(a \mid s)}$$
The gradient of the KL-divergence is zero at $\theta_t$

Change from trajectory distribution to state-action distribution:

$$\begin{align*}
KL \left( \rho_{\pi_{\theta_t}} \mid \rho_{\pi_{\theta}} \right) &= \mathbb{E}_{\tau \sim \rho_{\pi_{\theta_t}}} \ln \frac{\rho_{\pi_{\theta_t}}(\tau)}{\rho_{\pi_{\theta}}(\tau)} = \mathbb{E}_{\tau \sim \rho_{\pi_{\theta_t}}} \sum_{h=0}^{H-1} \ln \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_{\theta}(a_h \mid s_h)} \\
&= \frac{1}{1 - \gamma} \mathbb{E}_{s_h, a_h \sim d_{\mu}^{\pi_{\theta_t}}} \left[ \ln \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_{\theta}(a_h \mid s_h)} \right] := \ell(\theta)
\end{align*}$$

$$\begin{align*}
\nabla_{\theta} \ell(\theta) \mid_{\theta = \theta_t} &= \mathbb{E}_{s \sim d_{\mu}^{\pi_{\theta_t}}} \sum_a \pi_{\theta_t}(a \mid s) \left( - \nabla_{\theta} \ln \pi_{\theta}(a_h \mid s_h) \bigg|_{\theta = \theta_t} \right) \\
&= - \mathbb{E}_{s \sim d_{\mu}^{\pi_{\theta_t}}} \sum_a \pi_{\theta_t}(a \mid s) \frac{\nabla_{\theta} \pi_{\theta_t}(a \mid s)}{\pi_{\theta_t}(a \mid s)} = 0
\end{align*}$$
Let’s compute the Hessian of the KL-divergence at $\theta_t$

$$\mathbb{E}_{s,a \sim d_{\theta_t}^\pi} \left[ \ln \frac{\pi_{\theta_t}(a_h | s_h)}{\pi_{\theta}(a_h | s_h)} \right] := \ell(\theta)$$
Let’s compute the Hessian of the KL-divergence at $\theta_t$

$$\mathbb{E}_{s, a \sim d_{\mu_t}^{\theta_t}} \left[ \ln \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_{\theta}(a_h \mid s_h)} \right] := \ell(\theta)$$

$$\nabla_\theta^2 \ell(\theta) \big|_{\theta = \theta_t} = \mathbb{E}_{s \sim d_{\mu_t}^{\theta_t}} \sum_a \pi_{\theta_t}(a \mid s) \left( -\nabla_\theta^2 \ln \pi_{\theta}(a \mid s) \big|_{\theta = \theta_t} \right)$$
Let's compute the Hessian of the KL-divergence at $\theta_t$

$$
\mathbb{E}_{s,a \sim d^s_{\mu_0}} \left[ \ln \frac{\pi_{\theta_t}(a_h | s_h)}{\pi_{\theta}(a_h | s_h)} \right] := \ell(\theta)
$$

$$
\nabla^2_\theta \ell(\theta) |_{\theta=\theta_t} = \mathbb{E}_{s,a \sim d_s^{\mu_t}} \sum_a \pi_{\theta_t}(a | s) \left( -\nabla^2_\theta \ln \pi_{\theta}(a | s) |_{\theta=\theta_t} \right)
$$

$$
= -\mathbb{E}_{s,a \sim d_s^{\mu_t}} \sum_a \pi_{\theta_t}(a | s) \left( \frac{\nabla^2_\theta \pi_{\theta_t}(a | s)}{\pi_{\theta_t}(a | s)} - \frac{\nabla_\theta \pi_{\theta_t}(a | s) \nabla_\theta \pi_{\theta_t}(a | s)^T}{\pi_{\theta_t}^2(a | s)} \right)
$$
Let’s compute the Hessian of the KL-divergence at $\theta_t$

$$\mathbb{E}_{s,a \sim d_{\mu}^{\pi_{\theta_t}}} \left[ \ln \frac{\pi_{\theta_t}(a_h | s_h)}{\pi(\theta_t)} \right] := \ell(\theta)$$

$$\nabla_{\theta}^2 \ell(\theta) |_{\theta = \theta_t} = \mathbb{E}_{s,a \sim d_{\mu}^{\pi_{\theta_t}}} \sum_a \pi_{\theta_t}(a | s) \left( -\nabla_{\theta}^2 \ln \pi_{\theta_t}(a | s) |_{\theta = \theta_t} \right)$$

$$= -\mathbb{E}_{s,a \sim d_{\mu}^{\pi_{\theta_t}}} \sum_a \pi_{\theta_t}(a | s) \left( \frac{\nabla_{\theta}^2 \pi_{\theta_t}(a | s)}{\pi_{\theta_t}(a | s)} - \frac{\nabla_{\theta} \pi_{\theta_t}(a | s) \nabla_{\theta} \pi_{\theta_t}(a | s)^T}{\pi_{\theta_t}^2(a | s)} \right)$$

$$= \mathbb{E}_{s,a \sim d_{\mu}^{\pi_{\theta_t}}} \left[ \nabla_{\theta} \ln \pi_{\theta_t}(a | s) \left( \nabla_{\theta} \ln \pi_{\theta_t}(a | s) \right)^T \right] \in \mathbb{R}^{\dim_{\theta} \times \dim_{\theta}}$$
Let’s compute the Hessian of the KL-divergence at $\theta_t$

$$\mathbb{E}_{s,a \sim d_\mu^\pi} \left[ \ln \frac{\pi_{\theta_t}(a_h | s_h)}{\pi(\mu(a | s_h))} \right] := \ell(\theta)$$

$$\nabla_\theta^2 \ell(\theta) \big|_{\theta=\theta_t} = \mathbb{E}_{s,a \sim d_\mu^\pi} \sum_a \pi_{\theta_t}(a | s) \left( -\nabla_\theta^2 \ln \pi_{\theta_t}(a | s) \big|_{\theta=\theta_t} \right)$$

$$= -\mathbb{E}_{s,a \sim d_\mu^\pi} \sum_a \pi_{\theta_t}(a | s) \left( \frac{\nabla_\theta^2 \pi_{\theta_t}(a | s)}{\pi_{\theta_t}(a | s)} - \frac{\nabla_\theta \pi_{\theta_t}(a | s) \nabla_\theta \pi_{\theta_t}(a | s)^T}{\pi_{\theta_t}^2(a | s)} \right)$$

$$= \mathbb{E}_{s,a \sim d_\mu^\pi} \left[ \nabla_\theta \ln \pi_{\theta_t}(a | s) \left( \nabla_\theta \ln \pi_{\theta_t}(a | s) \right)^T \right] \in \mathbb{R}^{\text{dim}_\theta \times \text{dim}_\theta}$$

It’s called fisher Information Matrix!
Summary so far:

We did second-order Taylor expansion on the KL constraint, and we get:

\[
\frac{1}{H}KL(\rho_{\pi_{\theta,t}} | \rho_{\pi_{\theta}}) \approx \frac{1}{2}(\theta - \theta_t)^\top F_{\theta_t}(\theta - \theta_t)
\]

\[
F_{\theta_t} := \mathbb{E}_{s,a \sim d_{\mu}}\left[ \nabla_{\theta} \ln \pi_{\theta_t}(a | s) \left( \nabla_{\theta} \ln \pi_{\theta_t}(a | s) \right)^\top \right] \in \mathbb{R}^{\text{dim}_\mu \times \text{dim}_\theta}
\]
Summary so far:

We did second-order Taylor expansion on the KL constraint, and we get:

\[
\frac{1}{H} KL \left( \rho_{\pi_{\theta_t}} \mid \rho_{\pi_{\theta}} \right) \approx \frac{1}{2} (\theta - \theta_t)^\top F_{\theta_t}(\theta - \theta_t)
\]

\[
F_{\theta_t} := \mathbb{E}_{s, a \sim d_{\mu}^{\pi_{\theta_t}}} \left[ \nabla_{\theta} \ln \pi_{\theta_t}(a \mid s) \left( \nabla_{\theta} \ln \pi_{\theta_t}(a \mid s) \right)^\top \right] \in \mathbb{R}^{\text{dim}_\theta \times \text{dim}_\theta}
\]

This leads to the following much simplified constrained optimization:
Summary so far:

We did second-order Taylor expansion on the KL constraint, and we get:

\[
\frac{1}{H} KL \left( \rho_{\pi_{\theta}} | \rho_{\pi_{0}} \right) \approx \frac{1}{2} (\theta - \theta_{t})^{\top} F_{\theta_{t}} (\theta - \theta_{t})
\]

\[
F_{\theta_{t}} := \mathbb{E}_{s, a \sim d_{\mu_{\theta_{t}}}} \left[ \nabla_{\theta} \ln \pi_{\theta_{t}} (a | s) \left( \nabla_{\theta} \ln \pi_{\theta_{t}} (a | s) \right)^{\top} \right] \in \mathbb{R}^{d_{\theta_{t}} \times d_{\theta_{t}}}
\]

This leads to the following much simplified constrained optimization:

\[
\max_{\theta} \nabla_{\theta} J(\pi_{\theta_{t}})^{\top} (\theta - \theta_{t})
\]

\[
s.t. \quad (\theta - \theta_{t})^{\top} F_{\theta_{t}} (\theta - \theta_{t}) \leq \delta
\]
Outlines

1. Quick intro on KL-divergence
2. A Trust-Region Formulation for Policy Optimization
3. Algorithm: Natural Policy Gradient
Put everything together, we get:

At iteration $t$, we update to $\theta_{t+1}$ via:

$$\max_{\theta} \nabla_{\theta} J(\pi_{\theta})^T(\theta - \theta_t)$$

s.t. $(\theta - \theta_t)^T F_{\theta_t}(\theta - \theta_t) \leq \delta$
Put everything together, we get:

At iteration $t$, we update to $\theta_{t+1}$ via:

$$\max_{\theta} \nabla_{\theta} J(\pi_{\theta_t})^T(\theta - \theta_t)$$

s.t. $(\theta - \theta_t)^T F_{\theta_t}(\theta - \theta_t) \leq \delta$

Linear objective and quadratic convex constraint, we can solve it optimally!
Put everything together, we get:

At iteration $t$, we update to $\theta_{t+1}$ via:

$$\max_{\theta} \nabla_{\theta} J(\pi_{\theta_t})^T(\theta - \theta_t)$$

s.t. $(\theta - \theta_t)^T F_{\theta_t}(\theta - \theta_t) \leq \delta$

Linear objective and quadratic convex constraint, we can solve it optimally!

Indeed this gives us:

$$\theta_{t+1} = \theta_t + \eta F_{\theta_t}^{-1} \nabla_{\theta} J(\pi_{\theta_t})$$
Put everything together, we get:

At iteration $t$, we update to $\theta_{t+1}$ via:

$$\max_{\theta} \nabla_{\theta} J(\pi_{\theta_t})^T(\theta - \theta_t)$$

subject to:

$$(\theta - \theta_t)^T F_{\theta_t}(\theta - \theta_t) \leq \delta$$

Linear objective and quadratic convex constraint, we can solve it optimally!

Indeed this gives us:

$$\theta_{t+1} = \theta_t + \eta F_{\theta_t}^{-1} \nabla_{\theta} J(\pi_{\theta_t})$$

Where $\eta = \sqrt{\frac{\delta}{\nabla_{\theta} J(\pi_{\theta_t})^T F_{\theta_t}^{-1} \nabla_{\theta} J(\pi_{\theta_t})}}$
Algorithm: Natural Policy Gradient

Initialize $\theta_0$

For $t = 0, \ldots$
Algorithm: Natural Policy Gradient

Initialize $\theta_0$

For $t = 0, \ldots$

\[ \text{Estimate PG } \nabla_\theta J(\pi_{\theta_t}) \]
Algorithm: Natural Policy Gradient

Initialize $\theta_0$

For $t = 0, \ldots$

Estimate PG $\nabla_\theta J(\pi_{\theta_t})$

Estimate Fisher info-matrix $F_{\theta_t} := \mathbb{E}_{s,a \sim d_{\mu_t}} \nabla_\theta \ln \pi_{\theta_t}(a | s)(\nabla_\theta \ln \pi_{\theta_t}(a | s))^\top$
Algorithm: Natural Policy Gradient

Initialize $\theta_0$

For $t = 0, \ldots$

Estimate PG $\nabla_{\theta} J(\pi_{\theta_t})$

Estimate Fisher info-matrix $F_{\theta_t} := \mathbb{E}_{s,a \sim d_{\mu_t}} \nabla_{\theta} \ln \pi_{\theta_t}(a \mid s)(\nabla_{\theta} \ln \pi_{\theta_t}(a \mid s))^\top$

Natural Gradient Ascent: $\theta_{t+1} = \theta_t + \eta F_{\theta_t}^{-1} \nabla_{\theta} J(\pi_{\theta_t})$
Algorithm: Natural Policy Gradient

Initialize $\theta_0$

For $t = 0, \ldots$

Estimate PG $\nabla_\theta J(\pi_{\theta_t})$

Estimate Fisher info-matrix $F_{\theta_t} := \mathbb{E}_{s,a \sim d_{\mu_t}} \nabla_\theta \ln \pi_{\theta_t}(a \mid s)(\nabla_\theta \ln \pi_{\theta_t}(a \mid s))^\top$

**Natural Gradient Ascent:** $\theta_{t+1} = \theta_t + \eta F_{\theta_t}^{-1} \nabla_\theta J(\pi_{\theta_t})$

Where $\eta = \sqrt{\frac{\delta}{\nabla_\theta J(\pi_{\theta_t})^\top F_{\theta_t}^{-1} \nabla_\theta J(\pi_{\theta_t})}}$
Algorithm: Natural Policy Gradient

Initialize $\theta_0$

For $t = 0, \ldots$

Estimate PG $\nabla_\theta J(\pi_{\theta_t})$

Estimate Fisher info-matrix $F_{\theta_t} := \mathbb{E}_{s,a \sim d_{\mu_{\theta_t}}} \nabla_\theta \ln \pi_{\theta_t}(a \mid s)(\nabla_\theta \ln \pi_{\theta_t}(a \mid s))^\top$

**Natural Gradient Ascent:** $\theta_{t+1} = \theta_t + \eta F_{\theta_t}^{-1} \nabla_\theta J(\pi_{\theta_t})$

Where $\eta = \sqrt{\frac{\delta}{\nabla_\theta J(\pi_{\theta_t})^\top F_{\theta_t}^{-1} \nabla_\theta J(\pi_{\theta_t})}}$

(We will implement it in HW2 on Cartpole)
Summary for today:

Trust Region Policy Optimization and NPG

At iteration $t$:

$$\max_{\pi_\theta} \mathbb{E}_{s \sim d_{\pi_\theta}} \left[ \mathbb{E}_{a \sim \pi_\theta(s)} A_{\pi_\theta}(s, a) \right]$$

$$\text{s.t., } KL\left( \rho_{\pi_\theta} \| \rho_{\pi_\theta} \right) \leq \delta$$

Intuition: maximize local adv subject to being incremental (in KL);
Summary for today:

Trust Region Policy Optimization and NPG

At iteration $t$:

$$\max_{\pi_{\theta}} \mathbb{E}_{s \sim d_{\mu_{\pi_{\theta}, t}}} \left[ \mathbb{E}_{a \sim \pi_{\theta}(s)} A_{\pi_{\theta}}(s, a) \right]$$

First-order Taylor expansion at $\theta_t$

s.t., $KL \left( \rho_{\pi_{\theta_t}} \| \rho_{\pi_{\theta}} \right) \leq \delta$

second-order Taylor expansion at $\theta_t$

Intuition: maximize local adv subject to being incremental (in KL);
Summary for today:

Trust Region Policy Optimization and NPG

At iteration $t$:

\[
\max_{\pi_\theta} \mathbb{E}_{s \sim d_\pi_{\theta_t}} \left[ \mathbb{E}_{a \sim \pi_\theta(s)} A_{\pi_\theta}(s, a) \right] \quad \text{First-order Taylor expansion at } \theta_t
\]

s.t., $KL (\rho_{\pi_{\theta_t}} | \rho_{\pi_{\theta}}) \leq \delta$ \quad second-order Taylor expansion at $\theta_t$

Intuition: maximize local adv subject to being incremental (in KL);

\[
\max_{\theta} \nabla_\theta J(\pi_{\theta_t})^\top (\theta - \theta_t) \quad \text{s.t. } (\theta - \theta_t)^\top F_{\theta_t}(\theta - \theta_t) \leq \delta
\]
Summary for today:

Trust Region Policy Optimization and NPG

At iteration $t$:

$$\max_{\pi_{\theta}} \mathbb{E}_{s \sim d_{\mu_t}} \left[ \mathbb{E}_{a \sim \pi_\theta(s)} A_{\pi_{\theta_t}}(s, a) \right]$$

subject to

$$\text{KL} \left( \rho_{\pi_{\theta_t}} \middle| \rho_{\pi_{\theta}} \right) \leq \delta$$

Intuition: maximize local adv subject to being incremental (in KL);

First-order Taylor expansion at $\theta_t$

$$\max_{\theta} \nabla_{\theta} J(\pi_{\theta_t})^\top(\theta - \theta_t)$$

subject to

$$(\theta - \theta_t)^\top F_{\theta_t}(\theta - \theta_t) \leq \delta$$

(Exercise: work out the arg max)
Summary for today:

Trust Region Policy Optimization and NPG

At iteration $t$:

$$\max_{\pi_\theta} \mathbb{E}_{s \sim d_\mu^{\pi_\theta t}} \left[ \mathbb{E}_{a \sim \pi_\theta(s)} A^{\pi_\theta}(s, a) \right]$$

First-order Taylor expansion at $\theta_t$

s.t., $KL \left( \rho_{\pi_\theta t} \mid \rho_{\pi_\theta} \right) \leq \delta$

second-order Taylor expansion at $\theta_t$

Intuition: maximize local adv subject to being incremental (in KL);

$$\theta_{t+1} = \theta_t + \eta F^{-1}_{\theta_t} \nabla \theta J(\pi_{\theta_t})$$

NPG

$$\max_{\theta} \nabla \theta J(\pi_{\theta_t})^\top (\theta - \theta_t)$$

s.t. $(\theta - \theta_t)^\top F_{\theta_t}(\theta - \theta_t) \leq \delta$

(Exercise: work out the arg max)