ECEN 743: Reinforcement Learning

Trust Region Methods

Dileep Kalathil
Assistant Professor
Department of Electrical and Computer Engineering
Texas A&M University

References

- [AJKS, Section 3]

Provable Guarantees for Policy Optimization Algorithms

- We discussed policy gradient and natural policy gradient algorithms

Provable Guarantees for Policy Optimization Algorithms

- We discussed policy gradient and natural policy gradient algorithms
- They are incremental algorithms, making small incremental update to the policy parameter

Provable Guarantees for Policy Optimization Algorithms

- We discussed policy gradient and natural policy gradient algorithms
- They are incremental algorithms, making small incremental update to the policy parameter
- The intuition (and hope) is that each update will give a better policy than the previous one

Provable Guarantees for Policy Optimization Algorithms

- We discussed policy gradient and natural policy gradient algorithms
- They are incremental algorithms, making small incremental update to the policy parameter
- The intuition (and hope) is that each update will give a better policy than the previous one
- PG and NPG do not explicitly give this monotone improvement guarantee

Provable Guarantees for Policy Optimization Algorithms

- We discussed policy gradient and natural policy gradient algorithms
- They are incremental algorithms, making small incremental update to the policy parameter
- The intuition (and hope) is that each update will give a better policy than the previous one
- PG and NPG do not explicitly give this monotone improvement guarantee
- We know that classical policy iteration algorithm gives monotone improvement guarantee

Provable Guarantees for Policy Optimization Algorithms

- We discussed policy gradient and natural policy gradient algorithms
- They are incremental algorithms, making small incremental update to the policy parameter
- The intuition (and hope) is that each update will give a better policy than the previous one
- PG and NPG do not explicitly give this monotone improvement guarantee
- We know that classical policy iteration algorithm gives monotone improvement guarantee
- However, due to the unknown MDP and large state space, greedy improvement of $Q_\pi(s, a)$ for each state-action pair is infeasible

Provable Guarantees for Policy Optimization Algorithms

- We discussed policy gradient and natural policy gradient algorithms
- They are incremental algorithms, making small incremental update to the policy parameter
- The intuition (and hope) is that each update will give a better policy than the previous one
- PG and NPG do not explicitly give this monotone improvement guarantee
- We know that classical policy iteration algorithm gives monotone improvement guarantee
- However, due to the unknown MDP and large state space, greedy improvement of $Q_\pi(s, a)$ for each state-action pair is infeasible
- How do we ensure provable improvement guarantees for policy optimization algorithms?

Performance Difference Lemma

- We will first prove an important result which is very useful in analyzing a number of RL algorithms

Performance Difference Lemma

- We will first prove an important result which is very useful in analyzing a number of RL algorithms

Lemma (Performance Difference Lemma)

Let $V_{\pi,\mu} = \mathbb{E}_{s\sim\mu}[V_\pi(s)]$. *For any two policies,* π *and* $\bar{\pi}$,

$$V_{\pi,\mu} - V_{\bar{\pi},\mu} = \frac{1}{(1-\gamma)}\mathbb{E}_{s\sim\rho_{\pi,\mu}(\cdot)}\mathbb{E}_{a\sim\pi(s,\cdot)}\left[A_{\bar{\pi}}(s,a)\right]$$

Performance Difference Lemma: Proof

Proof:

Performance Difference Lemma: Proof

Proof:

Performance Difference Lemma: Proof

Proof:

$$V_\pi(s) - V_{\bar{\pi}}(s) = \mathbb{E}_{\tau \sim P_{\text{traj},\pi}}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right] - V_{\bar{\pi}}(s)$$

Performance Difference Lemma: Proof

Proof:

$$
\begin{aligned}
V_\pi(s) - V_{\bar{\pi}}(s) &= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right] - V_{\bar{\pi}}(s) \\
&= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(r(s_t, a_t) + V_{\bar{\pi}}(s_t) - V_{\bar{\pi}}(s_t) \right) \right] - V_{\bar{\pi}}(s)
\end{aligned}
$$

Performance Difference Lemma: Proof

Proof:

$$
\begin{aligned}
V_\pi(s) - V_{\bar{\pi}}(s) &= \mathbb{E}_{\tau \sim P_{\text{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right] - V_{\bar{\pi}}(s) \\
&= \mathbb{E}_{\tau \sim P_{\text{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(r(s_t, a_t) + V_{\bar{\pi}}(s_t) - V_{\bar{\pi}}(s_t) \right) \right] - V_{\bar{\pi}}(s) \\
&= \mathbb{E}_{\tau \sim P_{\text{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(r(s_t, a_t) + \gamma V_{\bar{\pi}}(s_{t+1}) - V_{\bar{\pi}}(s_t) \right) \right]
\end{aligned}
$$

Performance Difference Lemma: Proof

Proof:

$$
\begin{aligned}
V_\pi(s) - V_{\bar{\pi}}(s) &= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right] - V_{\bar{\pi}}(s) \\
&= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(r(s_t, a_t) + V_{\bar{\pi}}(s_t) - V_{\bar{\pi}}(s_t) \right) \right] - V_{\bar{\pi}}(s) \\
&= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(r(s_t, a_t) + \gamma V_{\bar{\pi}}(s_{t+1}) - V_{\bar{\pi}}(s_t) \right) \right] \\
&= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(r(s_t, a_t) + \gamma \mathbb{E}[V_{\bar{\pi}}(s_{t+1}) | s_t, a_t] - V_{\bar{\pi}}(s_t) \right) \right]
\end{aligned}
$$

Performance Difference Lemma: Proof

Proof:

$$
\begin{aligned}
V_\pi(s) - V_{\bar\pi}(s) &= \mathbb{E}_{\tau\sim P_{\mathrm{traj},\pi}}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)\right] - V_{\bar\pi}(s) \\
&= \mathbb{E}_{\tau\sim P_{\mathrm{traj},\pi}}\left[\sum_{t=0}^{\infty}\gamma^t \left(r(s_t,a_t) + V_{\bar\pi}(s_t) - V_{\bar\pi}(s_t)\right)\right] - V_{\bar\pi}(s) \\
&= \mathbb{E}_{\tau\sim P_{\mathrm{traj},\pi}}\left[\sum_{t=0}^{\infty}\gamma^t \left(r(s_t,a_t) + \gamma V_{\bar\pi}(s_{t+1}) - V_{\bar\pi}(s_t)\right)\right] \\
&= \mathbb{E}_{\tau\sim P_{\mathrm{traj},\pi}}\left[\sum_{t=0}^{\infty}\gamma^t \left(r(s_t,a_t) + \gamma \mathbb{E}[V_{\bar\pi}(s_{t+1})|s_t,a_t] - V_{\bar\pi}(s_t)\right)\right] \\
&= \mathbb{E}_{\tau\sim P_{\mathrm{traj},\pi}}\left[\sum_{t=0}^{\infty}\gamma^t \left(Q_{\bar\pi}(s_t,a_t) - V_{\bar\pi}(s_t)\right)\right]
\end{aligned}
$$

Performance Difference Lemma: Proof

Proof:

$$
\begin{aligned}
V_\pi(s) - V_{\bar{\pi}}(s) &= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right] - V_{\bar{\pi}}(s) \\
&= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(r(s_t, a_t) + V_{\bar{\pi}}(s_t) - V_{\bar{\pi}}(s_t) \right) \right] - V_{\bar{\pi}}(s) \\
&= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(r(s_t, a_t) + \gamma V_{\bar{\pi}}(s_{t+1}) - V_{\bar{\pi}}(s_t) \right) \right] \\
&= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(r(s_t, a_t) + \gamma \mathbb{E}[V_{\bar{\pi}}(s_{t+1}) | s_t, a_t] - V_{\bar{\pi}}(s_t) \right) \right] \\
&= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(Q_{\bar{\pi}}(s_t, a_t) - V_{\bar{\pi}}(s_t) \right) \right] \\
&= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(A_{\bar{\pi}}(s_t, a_t) \right) \right]
\end{aligned}
$$

Performance Difference Lemma: Proof

Proof: So, we have,

Performance Difference Lemma: Proof

Proof: So, we have,

$$V_\pi(s) - V_{\bar{\pi}}(s) = \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(A_{\bar{\pi}}(s_t, a_t) \right) \right]$$

Performance Difference Lemma: Proof

Proof: So, we have,

$$V_\pi(s) - V_{\bar\pi}(s) = \mathbb{E}_{\tau \sim P_{\text{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(A_{\bar\pi}(s_t, a_t) \right) \right]$$

$$= \sum_{s'} \sum_{a'} \sum_{t=0}^{\infty} \mathbb{P}(s_t = s', a_t = a' | \pi, s_0 = s) \ \gamma^t \ A_{\bar\pi}(s', a')$$

Performance Difference Lemma: Proof

Proof: So, we have,

$$
\begin{aligned}
V_{\pi}(s) - V_{\bar{\pi}}(s) &= \mathbb{E}_{\tau \sim P_{\text{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(A_{\bar{\pi}}(s_t, a_t) \right) \right] \\
&= \sum_{s'} \sum_{a'} \sum_{t=0}^{\infty} \mathbb{P}(s_t = s', a_t = a' | \pi, s_0 = s) \ \gamma^t \ A_{\bar{\pi}}(s', a') \\
&= \sum_{s'} \sum_{a'} A_{\bar{\pi}}(s', a') (\sum_{t=0}^{\infty} \mathbb{P}(s_t = s', a_t = a' | \pi, s_0 = s) \gamma^t)
\end{aligned}
$$

Performance Difference Lemma: Proof

Proof: So, we have,

$$
\begin{aligned}
V_{\pi}(s) - V_{\bar{\pi}}(s) &= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^{t} \left(A_{\bar{\pi}}(s_t, a_t) \right) \right] \\
&= \sum_{s'} \sum_{a'} \sum_{t=0}^{\infty} \mathbb{P}(s_t = s', a_t = a' | \pi, s_0 = s) \ \gamma^{t} \ A_{\bar{\pi}}(s', a') \\
&= \sum_{s'} \sum_{a'} A_{\bar{\pi}}(s', a') (\sum_{t=0}^{\infty} \mathbb{P}(s_t = s', a_t = a' | \pi, s_0 = s) \gamma^{t}) \\
&= \frac{1}{(1-\gamma)} \sum_{s'} \sum_{a'} A_{\bar{\pi}}(s', a') \rho_{\pi, \delta(s)}(s', a')
\end{aligned}
$$

Performance Difference Lemma: Proof

Proof: So, we have,

$$
\begin{aligned}
V_\pi(s) - V_{\bar{\pi}}(s) &= \mathbb{E}_{\tau\sim P_{\mathrm{traj},\pi}}\left[\sum_{t=0}^{\infty}\gamma^t\left(A_{\bar{\pi}}(s_t,a_t)\right)\right] \\
&= \sum_{s'}\sum_{a'}\sum_{t=0}^{\infty}\mathbb{P}(s_t = s', a_t = a'|\pi, s_0 = s)\ \gamma^t\ A_{\bar{\pi}}(s',a') \\
&= \sum_{s'}\sum_{a'}A_{\bar{\pi}}(s',a')(\sum_{t=0}^{\infty}\mathbb{P}(s_t = s', a_t = a'|\pi, s_0 = s)\gamma^t) \\
&= \frac{1}{(1-\gamma)}\sum_{s'}\sum_{a'}A_{\bar{\pi}}(s',a')\rho_{\pi,\delta(s)}(s',a') \\
&= \frac{1}{(1-\gamma)}\mathbb{E}_{(s',a')\sim\rho_{\pi,\delta(s)}}[A_{\bar{\pi}}(s',a')]
\end{aligned}
$$

Performance Difference Lemma: Proof

Proof: So, we have,

$$
\begin{aligned}
V_\pi(s) - V_{\bar{\pi}}(s) &= \mathbb{E}_{\tau \sim P_{\mathrm{traj},\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \left(A_{\bar{\pi}}(s_t, a_t) \right) \right] \\
&= \sum_{s'} \sum_{a'} \sum_{t=0}^{\infty} \mathbb{P}(s_t = s', a_t = a' | \pi, s_0 = s)\ \gamma^t\ A_{\bar{\pi}}(s', a') \\
&= \sum_{s'} \sum_{a'} A_{\bar{\pi}}(s', a') (\sum_{t=0}^{\infty} \mathbb{P}(s_t = s', a_t = a' | \pi, s_0 = s) \gamma^t) \\
&= \frac{1}{(1-\gamma)} \sum_{s'} \sum_{a'} A_{\bar{\pi}}(s', a') \rho_{\pi, \delta(s)}(s', a') \\
&= \frac{1}{(1-\gamma)} \mathbb{E}_{(s', a') \sim \rho_{\pi, \delta(s)}} [A_{\bar{\pi}}(s', a')]
\end{aligned}
$$

Now, averaging over the initial state s with a given initial state distribution μ, we get

$$
\begin{aligned}
V_{\pi,\mu} - V_{\bar{\pi},\mu} &= \frac{1}{(1-\gamma)} \mathbb{E}_{(s', a') \sim \rho_{\pi, \mu}} [A_{\bar{\pi}}(s', a')] \\
&= \frac{1}{(1-\gamma)} \mathbb{E}_{s \sim \rho_{\pi, \mu}} \mathbb{E}_{a' \sim \pi(s, \cdot)} [A_{\bar{\pi}}(s, a)]
\end{aligned}
$$

Why PDL is Useful?

- PDL: $V_{\pi,\mu} - V_{\bar{\pi},\mu} = \frac{1}{(1-\gamma)}\mathbb{E}_{s\sim\rho_{\pi,\mu}}\mathbb{E}_{a\sim\pi(s,\cdot)}\left[A_{\bar{\pi}}(s,a)\right]$

Why PDL is Useful?

- PDL: $V_{\pi,\mu} - V_{\bar{\pi},\mu} = \frac{1}{(1-\gamma)} \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\bar{\pi}}(s,a) \right]$
- Let π_k be the current policy. Then, for any policy π,

$$V_{\pi,\mu} = V_{\pi_k,\mu} + \frac{1}{(1-\gamma)} \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

Why PDL is Useful?

- PDL: $V_{\pi,\mu} - V_{\bar{\pi},\mu} = \frac{1}{(1-\gamma)}\mathbb{E}_{s\sim\rho_{\pi,\mu}}\mathbb{E}_{a\sim\pi(s,\cdot)}\left[A_{\bar{\pi}}(s,a)\right]$
- Let π_k be the current policy. Then, for any policy π,

$$V_{\pi,\mu} = V_{\pi_k,\mu} + \frac{1}{(1-\gamma)}\mathbb{E}_{s\sim\rho_{\pi,\mu}}\mathbb{E}_{a\sim\pi(s,\cdot)}\left[A_{\pi_k}(s,a)\right]$$

- We can then get π_{k+1} as

$$\pi_{k+1} = \arg\max_{\pi}\ V_{\pi,\mu} = \arg\max_{\pi}\ \mathbb{E}_{s\sim\rho_{\pi,\mu}}\mathbb{E}_{a\sim\pi(s,\cdot)}\left[A_{\pi_k}(s,a)\right]$$

Why PDL is Useful?

- PDL: $V_{\pi,\mu} - V_{\bar{\pi},\mu} = \frac{1}{(1-\gamma)} \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\bar{\pi}}(s,a) \right]$
- Let π_k be the current policy. Then, for any policy π,

$$V_{\pi,\mu} = V_{\pi_k,\mu} + \frac{1}{(1-\gamma)} \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- We can then get π_{k+1} as

$$\pi_{k+1} = \arg\max_{\pi} \; V_{\pi,\mu} = \arg\max_{\pi} \; \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- So, we can get the next policy without actually evaluating it (which is expensive)!

Why PDL is Useful?

- PDL: $V_{\pi,\mu} - V_{\bar{\pi},\mu} = \frac{1}{(1-\gamma)} \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\bar{\pi}}(s,a) \right]$
- Let π_k be the current policy. Then, for any policy π,

$$V_{\pi,\mu} = V_{\pi_k,\mu} + \frac{1}{(1-\gamma)} \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- We can then get π_{k+1} as

$$\pi_{k+1} = \arg\max_{\pi} \; V_{\pi,\mu} = \arg\max_{\pi} \; \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- So, we can get the next policy without actually evaluating it (which is expensive)!
- Caveat:

Why PDL is Useful?

- PDL: $V_{\pi,\mu} - V_{\bar{\pi},\mu} = \frac{1}{(1-\gamma)} \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} [A_{\bar{\pi}}(s,a)]$
- Let π_k be the current policy. Then, for any policy π,

$$V_{\pi,\mu} = V_{\pi_k,\mu} + \frac{1}{(1-\gamma)} \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} [A_{\pi_k}(s,a)]$$

- We can then get π_{k+1} as

$$\pi_{k+1} = \arg\max_{\pi} \ V_{\pi,\mu} = \arg\max_{\pi} \ \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} [A_{\pi_k}(s,a)]$$

- So, we can get the next policy without actually evaluating it (which is expensive)!
- Caveat:
  - We know A_{π_k} and we can optimize over π;

Why PDL is Useful?

- PDL: $V_{\pi,\mu} - V_{\bar{\pi},\mu} = \frac{1}{(1-\gamma)} \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\bar{\pi}}(s,a) \right]$
- Let π_k be the current policy. Then, for any policy π,

$$V_{\pi,\mu} = V_{\pi_k,\mu} + \frac{1}{(1-\gamma)} \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- We can then get π_{k+1} as

$$\pi_{k+1} = \arg\max_{\pi} \; V_{\pi,\mu} = \arg\max_{\pi} \; \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- So, we can get the next policy without actually evaluating it (which is expensive)!
- Caveat:
  - We know A_{π_k} and we can optimize over π;
  - However, we don't know $\rho_{\pi,\mu}$ and evaluating that expectation requires sampling according to π

Trust Region Methods

- "Ideal" policy update step:

$$\pi_{k+1} = \arg\max_{\pi} \; \mathbb{E}_{s\sim\rho_{\pi,\mu}} \mathbb{E}_{a\sim\pi(s,\cdot)} \left[A_{\pi_k}(s,a)\right]$$

Trust Region Methods

- "Ideal" policy update step:

$$\pi_{k+1} = \arg\max_{\pi} \; \mathbb{E}_{s\sim\rho_{\pi,\mu}} \mathbb{E}_{a\sim\pi(s,\cdot)} \left[A_{\pi_k}(s,a)\right]$$

  - We know A_{π_k}, but expensive to evaluate expectation w.r.t. $\rho_{\pi,\mu}$ for all possible policies

Trust Region Methods

- "Ideal" policy update step:

$$\pi_{k+1} = \arg\max_{\pi} \; \mathbb{E}_{s\sim\rho_{\pi,\mu}} \mathbb{E}_{a\sim\pi(s,\cdot)} \left[A_{\pi_k}(s,a)\right]$$

  - We know A_{π_k}, but expensive to evaluate expectation w.r.t. $\rho_{\pi,\mu}$ for all possible policies
- Can we replace $\rho_{\pi,\mu}$ by $\rho_{\pi_k,\mu}$ (which we can evaluate)?

Trust Region Methods

- "Ideal" policy update step:

$$\pi_{k+1} = \arg\max_{\pi} \; \mathbb{E}_{s\sim\rho_{\pi,\mu}} \mathbb{E}_{a\sim\pi(s,\cdot)} \left[A_{\pi_k}(s,a)\right]$$

  - We know A_{π_k}, but expensive to evaluate expectation w.r.t. $\rho_{\pi,\mu}$ for all possible policies
- Can we replace $\rho_{\pi,\mu}$ by $\rho_{\pi_k,\mu}$ (which we can evaluate)?
  - Is $\rho_{\pi_k,\mu}$ a good approximation for $\rho_{\pi,\mu}$ for all π

Trust Region Methods

- "Ideal" policy update step:

$$\pi_{k+1} = \arg\max_{\pi} \; \mathbb{E}_{s\sim\rho_{\pi,\mu}} \mathbb{E}_{a\sim\pi(s,\cdot)} \left[A_{\pi_k}(s,a)\right]$$

  - We know A_{π_k}, but expensive to evaluate expectation w.r.t. $\rho_{\pi,\mu}$ for all possible policies
- Can we replace $\rho_{\pi,\mu}$ by $\rho_{\pi_k,\mu}$ (which we can evaluate)?
  - Is $\rho_{\pi_k,\mu}$ a good approximation for $\rho_{\pi,\mu}$ for all π
- It may be a good approximation for all policies within a trust region around π_k

$$\texttt{trust-region}(\pi_{\texttt{k}}) = \{\pi | D(\pi_k, \pi) \leq \alpha\}$$

Trust Region Methods

- "Ideal" policy update step:

$$\pi_{k+1} = \arg\max_{\pi} \; \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

  - We know A_{π_k}, but expensive to evaluate expectation w.r.t. $\rho_{\pi,\mu}$ for all possible policies
- Can we replace $\rho_{\pi,\mu}$ by $\rho_{\pi_k,\mu}$ (which we can evaluate)?
  - Is $\rho_{\pi_k,\mu}$ a good approximation for $\rho_{\pi,\mu}$ for all π
- It may be a good approximation for all policies within a trust region around π_k

$$\texttt{trust-region}(\pi_\texttt{k}) = \{\pi | D(\pi_k, \pi) \le \alpha\}$$

- "Practical" policy update step:

$$\pi_{k+1} = \arg\max_{\pi \in \texttt{trust-region}(\pi_\texttt{k})} \; \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

Trust Region Methods

- "Ideal" policy update step:

$$\pi_{k+1} = \arg\max_{\pi} \ \mathbb{E}_{s \sim \rho_{\pi,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

  - We know A_{π_k}, but expensive to evaluate expectation w.r.t. $\rho_{\pi,\mu}$ for all possible policies
- Can we replace $\rho_{\pi,\mu}$ by $\rho_{\pi_k,\mu}$ (which we can evaluate)?
  - Is $\rho_{\pi_k,\mu}$ a good approximation for $\rho_{\pi,\mu}$ for all π
- It may be a good approximation for all policies within a trust region around π_k

$$\texttt{trust-region}(\pi_k) = \{\pi | D(\pi_k, \pi) \leq \alpha\}$$

- "Practical" policy update step:

$$\pi_{k+1} = \arg\max_{\pi \in \texttt{trust-region}(\pi_k)} \ \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- Will this "practical" policy update step give an improvement?

Conservative Policy Iteration

- "Practical" policy update step:

$$\pi_{k+1} = \underset{\pi \in \texttt{trust-region}(\pi_k)}{\arg\max} \; \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

Conservative Policy Iteration

- "Practical" policy update step:

$$\pi_{k+1} = \underset{\pi \in \texttt{trust-region}(\pi_k)}{\arg\max} \; \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- How doe we ensure a trust region?

Conservative Policy Iteration

- "Practical" policy update step:

$$\pi_{k+1} = \underset{\pi \in \texttt{trust-region}(\pi_k)}{\arg\max} \; \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- How doe we ensure a trust region?
- Conservative Policy Iteration (CPI) algorithm:

Conservative Policy Iteration

- "Practical" policy update step:

$$\pi_{k+1} = \underset{\pi \in \texttt{trust-region}(\pi_k)}{\arg\max} \; \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- How doe we ensure a trust region?
- Conservative Policy Iteration (CPI) algorithm:
  1. $\pi' = \arg\max_{\pi \in \Pi} \; \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$

Conservative Policy Iteration

- "Practical" policy update step:

$$\pi_{k+1} = \underset{\pi \in \texttt{trust-region}(\pi_k)}{\arg\max} \; \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- How doe we ensure a trust region?
- Conservative Policy Iteration (CPI) algorithm:
  1. $\pi' = \arg\max_{\pi \in \Pi} \; \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$
  2. $\pi_{k+1}(s,\cdot) = (1-\alpha)\pi_k(s,\cdot) + \alpha \pi'(s,\cdot)$ for all $s \in \mathcal{S}$

Conservative Policy Iteration

- "Practical" policy update step:

$$\pi_{k+1} = \underset{\pi \in \texttt{trust-region}(\pi_k)}{\arg\max} \; \mathbb{E}_{s \sim \rho_{\pi_k, \mu}} \mathbb{E}_{a \sim \pi(s, \cdot)} \left[A_{\pi_k}(s, a) \right]$$

- How doe we ensure a trust region?
- Conservative Policy Iteration (CPI) algorithm:
  1. $\pi' = \arg\max_{\pi \in \Pi} \; \mathbb{E}_{s \sim \rho_{\pi_k, \mu}} \mathbb{E}_{a \sim \pi(s, \cdot)} \left[A_{\pi_k}(s, a) \right]$
  2. $\pi_{k+1}(s, \cdot) = (1 - \alpha) \pi_k(s, \cdot) + \alpha \pi'(s, \cdot)$ for all $s \in \mathcal{S}$
- In CPI, we get $\left\| \pi_{k+1}(s, \cdot) - \pi_k(s, \cdot) \right\|_1 \leq 2\alpha$

Conservative Policy Iteration

- "Practical" policy update step:

$$\pi_{k+1} = \operatorname*{arg\,max}_{\pi \in \texttt{trust-region}(\pi_k)} \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$$

- How doe we ensure a trust region?
- Conservative Policy Iteration (CPI) algorithm:
  1. $\pi' = \arg\max_{\pi \in \Pi} \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a) \right]$
  2. $\pi_{k+1}(s,\cdot) = (1-\alpha)\pi_k(s,\cdot) + \alpha\pi'(s,\cdot)$ for all $s \in \mathcal{S}$
- In CPI, we get $\|\pi_{k+1}(s,\cdot) - \pi_k(s,\cdot)\|_1 \leq 2\alpha$

Conservative Policy Iteration

- "Practical" policy update step:

$$\pi_{k+1} = \arg\max_{\pi \in \texttt{trust-region}(\pi_k)} \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a)\right]$$

- How doe we ensure a trust region?
- Conservative Policy Iteration (CPI) algorithm:
  1. $\pi' = \arg\max_{\pi \in \Pi} \ \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi(s,\cdot)} \left[A_{\pi_k}(s,a)\right]$
  2. $\pi_{k+1}(s,\cdot) = (1-\alpha)\pi_k(s,\cdot) + \alpha\pi'(s,\cdot)$ for all $s \in \mathcal{S}$
- In CPI, we get $\left\|\pi_{k+1}(s,\cdot) - \pi_k(s,\cdot)\right\|_1 \leq 2\alpha$

Theorem (Monotone improvement)

Let $\bar{A}_k = \mathbb{E}_{s \sim \rho_{\pi_k,\mu}} \mathbb{E}_{a \sim \pi'(s,\cdot)}[A_{\pi_k}(s,a)]$. *Then,* $V_{\pi_{k+1},\mu} - V_{\pi_k,\mu} \geq \frac{\alpha}{(1-\gamma)}\left(\bar{A}_k - \frac{2\alpha\gamma}{(1-\gamma)^2}\right)$.

Set $\alpha = \frac{\bar{A}_k(1-\gamma)^2}{4\gamma}$. *Then,* $V_{\pi_{k+1},\mu} - V_{\pi_k,\mu} \geq \frac{\bar{A}_k^2(1-\gamma)}{8\gamma}$.

Auxiliary results

- We will state the following supporting result without proof (For proof, see [AJKS], Chapter 12)

Lemma

Suppose we have $\|\pi(s,\cdot) - \pi_k(s,\cdot)\|_1 \leq 2\alpha$ *for all* s. *Then, we have*

$$\|\rho_{\pi,\mu} - \rho_{\pi_k,\mu}\|_1 \leq \frac{2\alpha\gamma}{(1-\gamma)}$$

Auxiliary results

- We will state the following supporting result without proof (For proof, see [AJKS], Chapter 12)

Lemma

Suppose we have $\|\pi(s,\cdot) - \pi_k(s,\cdot)\|_1 \leq 2\alpha$ *for all* s. *Then, we have*

$$\|\rho_{\pi,\mu} - \rho_{\pi_k,\mu}\|_1 \leq \frac{2\alpha\gamma}{(1-\gamma)}$$

- We will also use the following simple result

Lemma

Let z *be a random variable and* p, q *be two distributions. Then,*

$$|\mathbb{E}_{z\sim p}[f(z)] - \mathbb{E}_{z\sim q}[f(z)]| \leq \|p-q\|_1 \; \max_z |f(z)|$$

Monotone Improvement: Proof

Proof: From PDL,

□

Monotone Improvement: Proof

Proof: From PDL,

$$
\begin{aligned}
(1-\gamma)(V_{\pi_{k+1},\mu} - V_{\pi_k,\mu}) &= \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}} \mathbb{E}_{a\sim\pi_{k+1}(s,\cdot)}[A_{\pi_k}(s,a)] \\
&= \alpha \ \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}} \mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)]
\end{aligned}
$$

□

Monotone Improvement: Proof

Proof: From PDL,

$$
\begin{aligned}
(1-\gamma)(V_{\pi_{k+1},\mu} - V_{\pi_k,\mu}) &= \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi_{k+1}(s,\cdot)}[A_{\pi_k}(s,a)] \\
&= \alpha\ \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] \\
&= \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi_{k+1}(s,\cdot)}[A_{\pi_k}(s,a)] \\
&\qquad + \alpha\ \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)]
\end{aligned}
$$

□

Monotone Improvement: Proof

Proof: From PDL,

$$
\begin{aligned}
(1-\gamma)(V_{\pi_{k+1},\mu} - V_{\pi_k,\mu}) &= \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi_{k+1}(s,\cdot)}[A_{\pi_k}(s,a)] \\
&= \alpha\ \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] \\
&= \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi_{k+1}(s,\cdot)}[A_{\pi_k}(s,a)] \\
&\qquad + \alpha\ \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] \\
&\geq \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \alpha\ \max_{s,a,\pi}|A_\pi(s,a)|\ \|\rho_{\pi_{k+1},\mu} - \rho_{\pi_k,\mu}\|_1
\end{aligned}
$$

$\square$

Monotone Improvement: Proof

Proof: From PDL,

$$
\begin{aligned}
(1-\gamma)(V_{\pi_{k+1},\mu} - V_{\pi_k,\mu}) &= \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}} \mathbb{E}_{a\sim\pi_{k+1}(s,\cdot)}[A_{\pi_k}(s,a)] \\
&= \alpha\ \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}} \mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] \\
&= \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}} \mathbb{E}_{a\sim\pi_{k+1}(s,\cdot)}[A_{\pi_k}(s,a)] \\
&\qquad + \alpha\ \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}} \mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}} \mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] \\
&\geq \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}} \mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \alpha \max_{s,a,\pi} |A_\pi(s,a)|\ \|\rho_{\pi_{k+1},\mu} - \rho_{\pi_k,\mu}\|_1 \\
&\geq \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}} \mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \frac{\alpha}{(1-\gamma)}\ \|\rho_{\pi_{k+1},\mu} - \rho_{\pi_k,\mu}\|_1
\end{aligned}
$$

$\square$

Monotone Improvement: Proof

Proof: From PDL,

$$
\begin{aligned}
(1-\gamma)(V_{\pi_{k+1},\mu} - V_{\pi_k,\mu}) &= \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi_{k+1}(s,\cdot)}[A_{\pi_k}(s,a)] \\
&= \alpha\ \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] \\
&= \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi_{k+1}(s,\cdot)}[A_{\pi_k}(s,a)] \\
&\qquad + \alpha\ \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] \\
&\geq \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \alpha \max_{s,a,\pi} |A_{\pi}(s,a)|\ \|\rho_{\pi_{k+1},\mu} - \rho_{\pi_k,\mu}\|_1 \\
&\geq \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \frac{\alpha}{(1-\gamma)}\ \|\rho_{\pi_{k+1},\mu} - \rho_{\pi_k,\mu}\|_1 \\
&\geq \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \frac{\alpha}{(1-\gamma)}\ \frac{2\alpha\gamma}{(1-\gamma)}
\end{aligned}
$$

$\square$

Monotone Improvement: Proof

Proof: From PDL,

$$
\begin{aligned}
(1-\gamma)(V_{\pi_{k+1},\mu} - V_{\pi_k,\mu}) &= \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi_{k+1}(s,\cdot)}[A_{\pi_k}(s,a)] \\
&= \alpha\ \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] \\
&= \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi_{k+1}(s,\cdot)}[A_{\pi_k}(s,a)] \\
&\qquad + \alpha\ \mathbb{E}_{s\sim\rho_{\pi_{k+1},\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] \\
&\geq \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \alpha \max_{s,a,\pi} |A_\pi(s,a)|\ \|\rho_{\pi_{k+1},\mu} - \rho_{\pi_k,\mu}\|_1 \\
&\geq \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \frac{\alpha}{(1-\gamma)}\ \|\rho_{\pi_{k+1},\mu} - \rho_{\pi_k,\mu}\|_1 \\
&\geq \alpha\ \mathbb{E}_{s\sim\rho_{\pi_k,\mu}}\mathbb{E}_{a\sim\pi'(s,\cdot)}[A_{\pi_k}(s,a)] - \frac{\alpha}{(1-\gamma)}\ \frac{2\alpha\gamma}{(1-\gamma)} \\
&\geq \alpha\left(\bar{A}_k - \frac{2\alpha\gamma}{(1-\gamma)^2}\right)
\end{aligned}
$$

Select α to get the maximum improvement. This will complete the proof.

$\square$

TRPO and PPO

References

- Schulman, et al. "Trust region policy optimization", *International Conference on Machine Learning (ICML)*, 2015.
- Schulman, et al. "Proximal policy optimization algorithms", 2017.
- [AJKS], Section 3

Some Distance Metric

- Total variation distance between two distributions p, q:

$$D_{TV}(p, q) = \frac{1}{2}\|p - q\|_1$$

Some Distance Metric

- Total variation distance between two distributions p, q:

$$D_{TV}(p,q) = \frac{1}{2}\|p-q\|_1$$

- Total variation distance between two policies π_1, π_2 :

$$D_{TV}^{\max}(\pi_1,\pi_2) = \max_s D_{TV}(\pi_1(s,\cdot),\pi_2(s,\cdot))$$

Some Distance Metric

- Total variation distance between two distributions p, q:

$$D_{TV}(p,q) = \frac{1}{2}\|p-q\|_1$$

- Total variation distance between two policies π_1, π_2 :

$$D_{TV}^{\max}(\pi_1,\pi_2) = \max_s D_{TV}(\pi_1(s,\cdot),\pi_2(s,\cdot))$$

- KL divergence between two distributions p, q:

$$D_{KL}(p,q) = \sum_z p(z) \ \log \frac{p(z)}{q(z)}$$

Some Distance Metric

- Total variation distance between two distributions p, q:

$$D_{TV}(p,q) = \frac{1}{2}\|p-q\|_1$$

- Total variation distance between two policies π_1, π_2 :

$$D_{TV}^{\max}(\pi_1,\pi_2) = \max_s D_{TV}(\pi_1(s,\cdot),\pi_2(s,\cdot))$$

- KL divergence between two distributions p, q:

$$D_{KL}(p,q) = \sum_z p(z)\ \log\frac{p(z)}{q(z)}$$

- KL divergence between two policies π_1, π_2 :

$$D_{KL}^{\max}(\pi_1,\pi_2) = \max_s D_{KL}(\pi_1(x,\cdot),\pi_2(x,\cdot))$$

$$D_{KL}^{\pi_1}(\pi_1,\pi_2) = \mathbb{E}_{s\sim\rho_{\pi_1}} D_{KL}(\pi_1(s,\cdot),\pi_2(s,\cdot))$$

Trust region Policy Optimization (TRPO)

- TRPO performs parameter update by solving the optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{x \sim \rho_{\pi_{\theta_k}}, \mu} \mathbb{E}_{a \sim \pi_\theta(x, \cdot)} [A_{\pi_{\theta_k}}(x, a)] \\ \text{s.t.} \quad & D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k}, \pi_\theta) \leq \alpha \end{aligned}$$

Trust region Policy Optimization (TRPO)

- TRPO performs parameter update by solving the optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{x \sim \rho_{\pi_{\theta_k}, \mu}} \mathbb{E}_{a \sim \pi_\theta(x, \cdot)} [A_{\pi_{\theta_k}}(x, a)] \\ \text{s.t.} \quad & D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k}, \pi_\theta) \leq \alpha \end{aligned}$$

- We are interested in small local update in parameters

Trust region Policy Optimization (TRPO)

- TRPO performs parameter update by solving the optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{x \sim \rho_{\pi_{\theta_k}, \mu}} \mathbb{E}_{a \sim \pi_\theta(x, \cdot)} [A_{\pi_{\theta_k}}(x, a)] \\ \text{s.t.} \quad & D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k}, \pi_\theta) \leq \alpha \end{aligned}$$

- We are interested in small local update in parameters
- We can perform *sequential quadratic programming*

Trust region Policy Optimization (TRPO)

- TRPO performs parameter update by solving the optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{x\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_\theta(x,\cdot)}[A_{\pi_{\theta_k}}(x,a)] \\ \text{s.t.} \quad & D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k},\pi_\theta) \leq \alpha \end{aligned}$$

- We are interested in small local update in parameters
- We can perform *sequential quadratic programming*
  - We can find a linear approximation of the objective function

Trust region Policy Optimization (TRPO)

- TRPO performs parameter update by solving the optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{x\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_\theta(x,\cdot)}[A_{\pi_{\theta_k}}(x,a)] \\ \text{s.t.} \quad & D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k},\pi_\theta) \leq \alpha \end{aligned}$$

- We are interested in small local update in parameters
- We can perform *sequential quadratic programming*
    - We can find a linear approximation of the objective function
    - We can find a quadratic approximation of the constraint

Trust region Policy Optimization (TRPO)

- TRPO performs parameter update by solving the optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{x \sim \rho_{\pi_{\theta_k}, \mu}} \mathbb{E}_{a \sim \pi_\theta(x, \cdot)} [A_{\pi_{\theta_k}}(x, a)] \\ \text{s.t.} \quad & D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k}, \pi_\theta) \leq \alpha \end{aligned}$$

- We are interested in small local update in parameters
- We can perform *sequential quadratic programming*
  - We can find a linear approximation of the objective function
  - We can find a quadratic approximation of the constraint
- Linear approximation of the objective function is $(\delta\theta)^\top \nabla V_{\pi_{\theta_k}}$ (*prove this!*)

Trust region Policy Optimization (TRPO)

- TRPO performs parameter update by solving the optimization problem

$$\max_{\theta} \quad \mathbb{E}_{x \sim \rho_{\pi_{\theta_k}, \mu}} \mathbb{E}_{a \sim \pi_\theta(x, \cdot)} [A_{\pi_{\theta_k}}(x, a)]$$
$$\text{s.t.} \quad D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k}, \pi_\theta) \leq \alpha$$

- We are interested in small local update in parameters
- We can perform *sequential quadratic programming*
  - We can find a linear approximation of the objective function
  - We can find a quadratic approximation of the constraint
- Linear approximation of the objective function is $(\delta\theta)^\top \nabla V_{\pi_{\theta_k}}$ (*prove this!*)
- Quadratic approximation of the constraint is $\delta\theta^\top F(\theta_k) \delta\theta$ (we have proved this when we discussed NPG)

TRPO with Paramterized Policies

- TRPO performs parameter update by solving the optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{x \sim \rho_{\pi_{\theta_k}, \mu}} \mathbb{E}_{a \sim \pi_\theta(x, \cdot)} [A_{\pi_{\theta_k}}(x, a)] \\ \text{s.t.} \quad & D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k}, \pi_\theta) \leq \alpha \end{aligned}$$

TRPO with Paramterized Policies

- TRPO performs parameter update by solving the optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{x \sim \rho_{\pi_{\theta_k}, \mu}} \mathbb{E}_{a \sim \pi_\theta(x, \cdot)} [A_{\pi_{\theta_k}}(x, a)] \\ \text{s.t.} \quad & D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k}, \pi_\theta) \leq \alpha \end{aligned}$$

- We are interested in small local update in parameters

TRPO with Paramterized Policies

- TRPO performs parameter update by solving the optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{x \sim \rho_{\pi_{\theta_k}, \mu}} \mathbb{E}_{a \sim \pi_\theta(x, \cdot)} [A_{\pi_{\theta_k}}(x, a)] \\ \text{s.t.} \quad & D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k}, \pi_\theta) \leq \alpha \end{aligned}$$

- We are interested in small local update in parameters

TRPO with Paramterized Policies

- TRPO performs parameter update by solving the optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{x \sim \rho_{\pi_{\theta_k}}, \mu} \mathbb{E}_{a \sim \pi_{\theta}(x, \cdot)} [A_{\pi_{\theta_k}}(x, a)] \\ \text{s.t.} \quad & D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k}, \pi_{\theta}) \leq \alpha \end{aligned}$$

- We are interested in small local update in parameters
- Update the parameter $\theta_{k+1} = \theta_k + \delta\theta$, where $\delta\theta$ is the solution of the optimization problem

$$\arg\max_{\delta} \quad \delta^{\top} \nabla V_{\pi_{\theta_k}}, \quad \text{s.t.} \quad \delta^{\top} F(\theta_k) \delta \leq \alpha$$

TRPO with Paramterized Policies

- TRPO performs parameter update by solving the optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{x \sim \rho_{\pi_{\theta_k},\mu}} \mathbb{E}_{a \sim \pi_\theta(x,\cdot)} [A_{\pi_{\theta_k}}(x,a)] \\ \text{s.t.} \quad & D_{KL}^{\pi_{\theta_k}}(\pi_{\theta_k}, \pi_\theta) \leq \alpha \end{aligned}$$

- We are interested in small local update in parameters
- Update the parameter $\theta_{k+1} = \theta_k + \delta\theta$, where $\delta\theta$ is the solution of the optimization problem

$$\arg\max_{\delta} \quad \delta^\top \nabla V_{\pi_{\theta_k}}, \quad \text{s.t.} \;\; \delta^\top F(\theta_k) \delta \leq \alpha$$

- This reduces to NPG

$$\theta_{k+1} \leftarrow \theta_k + \sqrt{\frac{\alpha}{(\nabla V_{\pi_{\theta_k}})^\top F(\theta_k)^{-1} (\nabla V_{\pi_{\theta_k}})}} \, F(\theta_k)^{-1} (\nabla V_{\pi_{\theta_k}})$$

Importance Sampling

- Before discussing PPO, let us take a look at the idea of importance sampling

Importance Sampling

- Before discussing PPO, let us take a look at the idea of importance sampling
- Importance sampling is a technique for estimating expectations using samples drawn from a different distribution

Importance Sampling

- Before discussing PPO, let us take a look at the idea of importance sampling
- Importance sampling is a technique for estimating expectations using samples drawn from a different distribution
- Let z be a random variable. Let p, q be two distributions

Importance Sampling

- Before discussing PPO, let us take a look at the idea of importance sampling
- Importance sampling is a technique for estimating expectations using samples drawn from a different distribution
- Let z be a random variable. Let p, q be two distributions
- We want to estimate $\mathbb{E}_{z \sim p}[f(z)]$

Importance Sampling

- Before discussing PPO, let us take a look at the idea of importance sampling
- Importance sampling is a technique for estimating expectations using samples drawn from a different distribution
- Let z be a random variable. Let p, q be two distributions
- We want to estimate $\mathbb{E}_{z\sim p}[f(z)]$
- One obvious approach is: generate n i.i.d. samples $(z_i)_{i=1}^n$ according to p. Then, $\mathbb{E}_{z\sim p}[f(z)] \approx \frac{1}{n}\sum_{i=1}^n f(z_i)$

Importance Sampling

- Before discussing PPO, let us take a look at the idea of importance sampling
- Importance sampling is a technique for estimating expectations using samples drawn from a different distribution
- Let z be a random variable. Let p, q be two distributions
- We want to estimate $\mathbb{E}_{z \sim p}[f(z)]$
- One obvious approach is: generate n i.i.d. samples $(z_i)_{i=1}^n$ according to p. Then, $\mathbb{E}_{z \sim p}[f(z)] \approx \frac{1}{n} \sum_{i=1}^n f(z_i)$
- Suppose we can only generate i.i.d. samples according q. How do we get the estimate for $\mathbb{E}_{z \sim p}[f(z)]$?

Importance Sampling

- We have

$$\mathbb{E}_{z\sim p}[f(z)] = \mathbb{E}_{z\sim q}[\frac{p(z)}{q(z)}f(z)] \approx \frac{1}{n}\sum_{i=1}^{n}\frac{p(z_i)}{q(z_i)}f(z_i), \quad \text{where}, \quad z_i \sim q$$

Importance Sampling

- We have

$$\mathbb{E}_{z\sim p}[f(z)] = \mathbb{E}_{z\sim q}[\frac{p(z)}{q(z)}f(z)] \approx \frac{1}{n}\sum_{i=1}^{n}\frac{p(z_i)}{q(z_i)}f(z_i), \quad \text{where,} \quad z_i \sim q$$

- Importance sampling gives an unbiased estimate

Importance Sampling

- We have

$$\mathbb{E}_{z\sim p}[f(z)] = \mathbb{E}_{z\sim q}[\frac{p(z)}{q(z)}f(z)] \approx \frac{1}{n}\sum_{i=1}^{n}\frac{p(z_i)}{q(z_i)}f(z_i), \quad \text{where}, \quad z_i \sim q$$

- Importance sampling gives an unbiased estimate
- What is the variance of the importance sampling based estimate?

Importance Sampling

- We have

$$\mathbb{E}_{z\sim p}[f(z)] = \mathbb{E}_{z\sim q}[\frac{p(z)}{q(z)}f(z)] \approx \frac{1}{n}\sum_{i=1}^{n}\frac{p(z_i)}{q(z_i)}f(z_i), \quad \text{where,} \quad z_i \sim q$$

- Importance sampling gives an unbiased estimate
- What is the variance of the importance sampling based estimate?

$$\begin{aligned}\text{Var}\left(\frac{p(z)}{q(z)}f(z)\right) &= \mathbb{E}_{z\sim q}\left[\left(\frac{p(z)}{q(z)}f(z)\right)^2\right] - \left(\mathbb{E}_{z\sim q}\left[\frac{p(z)}{q(z)}f(z)\right]\right)^2 \\ &= \mathbb{E}_{z\sim p}\left[\frac{p(z)}{q(z)}f^2(z)\right] - \left(\mathbb{E}_{z\sim p}[f(z)]\right)^2\end{aligned}$$

Importance Sampling

- We have

$$\mathbb{E}_{z\sim p}[f(z)] = \mathbb{E}_{z\sim q}[\frac{p(z)}{q(z)} f(z)] \approx \frac{1}{n}\sum_{i=1}^{n} \frac{p(z_i)}{q(z_i)} f(z_i), \quad \text{where,} \quad z_i \sim q$$

- Importance sampling gives an unbiased estimate
- What is the variance of the importance sampling based estimate?

$$\begin{aligned} \text{Var}\left(\frac{p(z)}{q(z)} f(z)\right) &= \mathbb{E}_{z\sim q}\left[\left(\frac{p(z)}{q(z)} f(z)\right)^2\right] - \left(\mathbb{E}_{z\sim q}\left[\frac{p(z)}{q(z)} f(z)\right]\right)^2 \\ &= \mathbb{E}_{z\sim p}\left[\frac{p(z)}{q(z)} f^2(z)\right] - (\mathbb{E}_{z\sim p}[f(z)])^2 \end{aligned}$$

- Importance sampling weight, $p(z)/q(z)$, can be very large for some z. Then the variance can be very large.

Importance Sampling in RL

- We want to solve $\max_\theta \; \mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_\theta(s,\cdot)}[A_{\pi_{\theta_k}}(s,a)]$

Importance Sampling in RL

- We want to solve $\max_\theta \; \mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_\theta(s,\cdot)}[A_{\pi_{\theta_k}}(s,a)]$
- The objective function can be written as

$$\mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_{\theta_k}(s,\cdot)}\left[\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(s,a)}A_{\pi_{\theta_k}}(s,a)\right]$$

Importance Sampling in RL

- We want to solve $\max_\theta \ \mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_\theta(s,\cdot)}[A_{\pi_{\theta_k}}(s,a)]$
- The objective function can be written as

$$\mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_{\theta_k}(s,\cdot)}\left[\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(s,a)}A_{\pi_{\theta_k}}(s,a)\right]$$

- To estimate the above, we can sample $s_0 \sim \mu$ and then generating a trajectory $(x_\tau, a_\tau)_\tau^T$ according to policy π_{θ_k}. The objective can be estimated by generating multiple trajectories

Importance Sampling in RL

- We want to solve $\max_\theta \; \mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_\theta(s,\cdot)}[A_{\pi_{\theta_k}}(s,a)]$
- The objective function can be written as

$$\mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_{\theta_k}(s,\cdot)}\left[\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(s,a)}A_{\pi_{\theta_k}}(s,a)\right]$$

- To estimate the above, we can sample $s_0 \sim \mu$ and then generating a trajectory $(x_\tau, a_\tau)_\tau^T$ according to policy π_{θ_k}. The objective can be estimated by generating multiple trajectories
- The optimization problem then can then be solved using a direct stochastic gradient ascent approach

Proximal Policy Optimization (PPO)

- PPO solves the following optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{s \sim \rho_{\pi_{\theta_k}, \mu}} \mathbb{E}_{a \sim \pi_\theta(s, \cdot)} [A_{\pi_{\theta_k}}(s, a)] \\ \text{s.t.} \quad & D_{TV}^{\max}(\pi_{\theta_k}, \pi_\theta) \leq \alpha \end{aligned}$$

Proximal Policy Optimization (PPO)

- PPO solves the following optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_\theta(s,\cdot)}[A_{\pi_{\theta_k}}(s,a)] \\ \text{s.t.} \quad & D_{TV}^{\max}(\pi_{\theta_k},\pi_\theta) \le \alpha \end{aligned}$$

- Instead of solving this using sequential approximation, PPO proposes a direct stochastic gradient ascent approach

Proximal Policy Optimization (PPO)

- PPO solves the following optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_\theta(s,\cdot)}[A_{\pi_{\theta_k}}(s,a)] \\ \text{s.t.} \quad & D_{TV}^{\max}(\pi_{\theta_k},\pi_\theta) \le \alpha \end{aligned}$$

- Instead of solving this using sequential approximation, PPO proposes a direct stochastic gradient ascent approach
- First, the objective function is rewritten as

$$\mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_{\theta_k}(s,\cdot)}\left[\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(s,a)}A_{\pi_{\theta_k}}(s,a)\right]$$

Proximal Policy Optimization (PPO)

- PPO solves the following optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_\theta(s,\cdot)}[A_{\pi_{\theta_k}}(s,a)] \\ \text{s.t.} \quad & D_{TV}^{\max}(\pi_{\theta_k},\pi_\theta) \leq \alpha \end{aligned}$$

- Instead of solving this using sequential approximation, PPO proposes a direct stochastic gradient ascent approach
- First, the objective function is rewritten as

$$\mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_{\theta_k}(s,\cdot)}\left[\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(s,a)}A_{\pi_{\theta_k}}(s,a)\right]$$

- This is a standard *importance sampling* technique

Proximal Policy Optimization (PPO)

- PPO solves the following optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{s \sim \rho_{\pi_{\theta_k}, \mu}} \mathbb{E}_{a \sim \pi_\theta(s, \cdot)} [A_{\pi_{\theta_k}}(s, a)] \\ \text{s.t.} \quad & D_{TV}^{\max}(\pi_{\theta_k}, \pi_\theta) \leq \alpha \end{aligned}$$

- Instead of solving this using sequential approximation, PPO proposes a direct stochastic gradient ascent approach
- First, the objective function is rewritten as

$$\mathbb{E}_{s \sim \rho_{\pi_{\theta_k}, \mu}} \mathbb{E}_{a \sim \pi_{\theta_k}(s, \cdot)} \left[\frac{\pi_\theta(s, a)}{\pi_{\theta_k}(s, a)} A_{\pi_{\theta_k}}(s, a) \right]$$

- This is a standard *importance sampling* technique
- We can approximate the objective using the trajectories generated according to the policy π_{θ_k}

Proximal Policy Optimization (PPO)

- PPO solves the following optimization problem

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_\theta(s,\cdot)}[A_{\pi_{\theta_k}}(s,a)] \\ \text{s.t.} \quad & D_{TV}^{\max}(\pi_{\theta_k},\pi_\theta) \leq \alpha \end{aligned}$$

- Instead of solving this using sequential approximation, PPO proposes a direct stochastic gradient ascent approach
- First, the objective function is rewritten as

$$\mathbb{E}_{s\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_{\theta_k}(s,\cdot)}\left[\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(s,a)}A_{\pi_{\theta_k}}(s,a)\right]$$

- This is a standard *importance sampling* technique
- We can approximate the objective using the trajectories generated according to the policy π_{θ_k}
- The constraints are enforced by a *clipping trick*

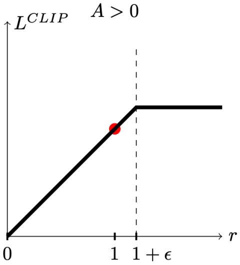
Proximal Policy Optimization (PPO)

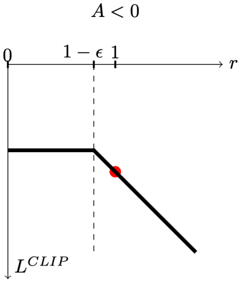
- For ensuring that $\pi_{\theta_k}(s,a)$ and $\pi_\theta(s,a)$ are not very different, PPO modifies the objective function as follows

$$L(\theta) = \mathbb{E}_{x\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_{\theta_k}(s,\cdot)}\left[\min\left\{\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(s,a)}A_{\pi_{\theta_k}}(s,a), \text{clip}\left(\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(x,a)}; 1-\epsilon, 1+\epsilon\right)A_{\pi_{\theta_k}}(s,a)\right\}\right],$$

where

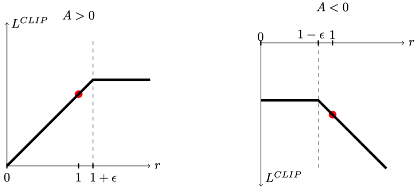
$$\text{clip}(z; 1-\epsilon, 1+\epsilon) = \begin{cases} 1-\epsilon & z \leq 1-\epsilon \\ 1+\epsilon & z \geq 1+\epsilon \\ z & \text{otherwise} \end{cases}$$





Proximal Policy Optimization (PPO)

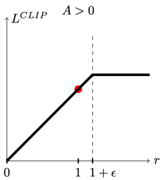
$$L(\theta) = \mathbb{E}_{x\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_{\theta_k}(s,\cdot)}\left[\min\left\{\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(s,a)}A_{\pi_{\theta_k}}(s,a), \text{clip}\left(\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(x,a)};1-\epsilon,1+\epsilon\right)A_{\pi_{\theta_k}}(s,a)\right\}\right],$$

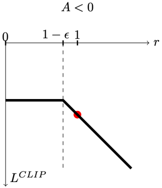


- Clipping ensure that for any (s,a) such that $\frac{\pi_\theta(s,a)}{\pi_{\theta_t}(x,a)} \notin [1-\epsilon, 1+\epsilon]$, we get $\nabla_\theta[\text{clip}\left(\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(x,a)};1-\epsilon,1+\epsilon\right)A_{\pi_{\theta_k}}(s,a)] = 0$

Proximal Policy Optimization (PPO)

$$L(\theta) = \mathbb{E}_{x\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_{\theta_k}(s,\cdot)}\left[\min\left\{\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(s,a)}A_{\pi_{\theta_k}}(s,a), \text{clip}\left(\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(x,a)}; 1-\epsilon, 1+\epsilon\right)A_{\pi_{\theta_k}}(s,a)\right\}\right],$$

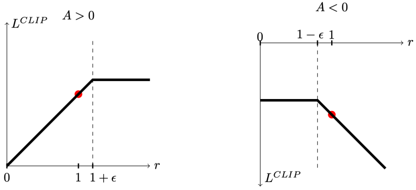




- Clipping ensure that for any (s, a) such that $\frac{\pi_\theta(s,a)}{\pi_{\theta_t}(x,a)} \notin [1-\epsilon, 1+\epsilon]$, we get $\nabla_\theta[\text{clip}\left(\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(x,a)}; 1-\epsilon, 1+\epsilon\right)A_{\pi_{\theta_k}}(s,a)] = 0$
- The *minimum* ensures that the objective function $L(\theta)$ is a lower bound of the original objective

Proximal Policy Optimization (PPO)

$$L(\theta) = \mathbb{E}_{x\sim\rho_{\pi_{\theta_k}},\mu}\mathbb{E}_{a\sim\pi_{\theta_k}(s,\cdot)}\left[\min\left\{\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(s,a)}A_{\pi_{\theta_k}}(s,a), \text{clip}\left(\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(x,a)};1-\epsilon,1+\epsilon\right)A_{\pi_{\theta_k}}(s,a)\right\}\right],$$



- Clipping ensure that for any (s,a) such that $\frac{\pi_\theta(s,a)}{\pi_{\theta_t}(x,a)} \notin [1-\epsilon, 1+\epsilon]$, we get $\nabla_\theta[\text{clip}\left(\frac{\pi_\theta(s,a)}{\pi_{\theta_k}(x,a)};1-\epsilon,1+\epsilon\right)A_{\pi_{\theta_k}}(s,a)] = 0$
- The *minimum* ensures that the objective function $L(\theta)$ is a lower bound of the original objective
- PPO optimizes the objective function using mini-batch stochastic gradient ascent (instead of the Taylor series expansion approach of NPG or TRPO)