17-740 Spring 2025

Algorithmic Foundations of Interactive Learning

Lecture 13: Natural Policy Gradient (NPG)

Lecturer : Steven Wu Scribe: Zora Wang, Lindia Tjuatja, Grace Liu, Nuoya Xiong

1.1 Introduction

This lecture presents three views of how one might arrive at the Natural Policy Gradient (NPG) algorithm.

- KL Regularization for stability
- Least squares regression
- Soft Policy Iteration (Hedge at every state)

Then we will present "practical" algorithms inspired by NPG

- Trust Region Policy Optimization (TRPO)
- Proximal Policy Optimization (PPO)

1.2 Natural Policy Gradient

Recall that

$$\begin{aligned}\nabla_\theta J(\pi_\theta) &= \mathbb{E}_{\tau\sim P_\theta}\left[\sum_{h=0}^{H-1} A_h^{\pi_\theta}(s_h,a_h)\nabla_\theta \log(\pi(a_h \mid s_h))\right] \\ &= H\cdot \mathbb{E}_{h\sim U[H-1], s_h\sim d_h^{\pi_\theta}, a_h\sim\pi_\theta(s_h)}[A_h^{\pi_\theta}(s_h,a_h)\nabla_\theta \log \pi_\theta(a_h \mid s_h)].\end{aligned}$$

Also recall the policy gradient update:

$$\theta_{t+1} \leftarrow \theta_t + \eta\nabla_\theta J(\pi_\theta).$$

1.2.1 KL Regularization View

The **regularization view** of the above equation is

$$\theta_{t+1} = \arg\max_{\theta} \langle \nabla_\theta J(\pi_{\theta_t}), \theta \rangle - \frac{1}{2\eta} \|\theta - \theta_t\|^2.$$

Notice that the above is a concave function. We can show that the regression view is equivalent to the policy gradient by solving this concave maximization problem, i.e. taking the derivative with respect to θ and setting it equal to zero.

We can explicitly enforce a hard limit on how far the parameters move by adding a constraint, giving us the **constrained optimization view**:

$$\max_{\theta} \langle \nabla_\theta J(\pi_{\theta_t}), \theta \rangle, \quad s.t. \quad \|\theta - \theta_t\|_2^2 \leq \delta.$$

Both the regularization view and constrained optimization view are looking at changes in parameter space. However, it can very well be the case that two sets of very different parameters lead to near-identical policies.

A more principled approach would instead constrain the differences between policies themselves. We can do this by constraining the **KL-divergence** between the old and new policies, instead of the ℓ_2 norm of their parameters:

$$\max_{\theta} J(\pi_\theta), \quad s.t. \quad \frac{1}{H} \mathrm{KL}(\mathbb{P}^{\pi_{\theta_t}} \| \mathbb{P}^{\pi_\theta}) \leq \delta,$$

By decomposing the KL divergence, we can get

$$\begin{aligned}
\frac{1}{H} \mathrm{KL}(\mathbb{P}^{\pi_{\theta_t}} \| \mathbb{P}^{\pi_\theta}) &= \frac{1}{H} \sum_{\tau} \mathbb{P}^{\theta_t}(\tau) \log \frac{\mathbb{P}^{\theta_t}(\tau)}{\mathbb{P}^{\theta}(\tau)} \\
&= \frac{1}{H} \sum_{\tau} \mathbb{P}^{\theta_t}(\tau) \sum_{h=0}^{H-1} \log \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_\theta(a_h \mid s_h)} \\
&= \mathbb{E}_{(s_h, a_h) \sim d^{\pi_{\theta_t}}} \left[\log \frac{\pi_{\theta_t}(a_h \mid s_h)}{\pi_\theta(a_h \mid s_h)} \right] \\
&:= \ell(\theta).
\end{aligned}$$

How can we approximate $\ell(\theta)$? We can use a Taylor expansion of $\ell(\theta)$ around $\theta = \theta_t$:

$$\begin{aligned}
\ell(\theta_t) &= 0 \\
\nabla_\theta \ell(\theta) &= 0|_{\theta=\theta_t} \\
\nabla_\theta^2 \ell(\theta_t) &= \mathbb{E}\left[\nabla_\theta \log \pi_{\theta_t}(a_h \mid s_h) \nabla_\theta \log \pi_{\theta_t}(a_h \mid s_h)^\top \right]
\end{aligned}$$

The Hessian $\nabla_\theta^2 \ell(\theta_t)$ is also referred to as the *Fisher Information Matrix*, denoted as $F(\theta_t)$. Hence, by the second-order approximation, we can estimate $\ell(\theta)$

$$\frac{1}{H}\mathrm{KL}(\mathbb{P}^{\pi_{\theta_t}} \| \mathbb{P}^{\pi_\theta}) \approx (\theta - \theta_t)^\top F(\theta_t)(\theta - \theta_t).$$

Thus, the update rule of *natural policy gradient* (NPG) can be written as

$$\theta_{t+1} \leftarrow \theta_t + \eta F^\dagger \nabla_\theta J(\pi_{\theta_t}),$$

where $F^\dagger$ is the Moore-Penrose Inverse.

1.2.2 Regression View

We can also view NPG as solving a weighted least squares regression problem.

Lemma 1 *Define*

$$w^* = \arg\min_w \mathbb{E}_{(s_h,a_h)\sim d^{\pi_\theta}} \left[\left(\underbrace{A_h^{\pi_\theta}(s_h,a_h)}_{\text{label}} - w^\top \underbrace{\nabla_\theta \log \pi_\theta(a_h \mid s_h)}_{\text{feature}} \right)^2 \right].$$

Then, $F^\dagger(\theta)\nabla_\theta J(\pi_\theta) = H w^*$.

Proof. By first order condition:

$$\mathbb{E}\left[\left(A_h^{\pi_\theta}(s_h,a_h) - (w^*)^\top \nabla_\theta \log \pi_\theta(a_h \mid s_h) \right) \nabla_\theta \log \pi_\theta(a_h \mid s_h) \right] = 0$$

It is equivalent to

$$\underbrace{\mathbb{E}\left[(A_h^{\pi_\theta}(s_h,a_h)) \nabla_\theta \log \pi_\theta(a_h \mid s_h) \right]}_{\frac{1}{H}\nabla_\theta J(\pi_\theta)} = \underbrace{\mathbb{E}\left[\nabla_\theta \log \pi_\theta(a_h \mid s_h) \nabla_\theta \log \pi_\theta(a_h \mid s_h)^\intercal \right]}_{Fisher\ \ Matrix\ \ F(\theta)} w^*.$$

Hence we have $F^\dagger(\theta)\nabla_\theta J(\pi_\theta) = H w^*$.

The lemma above shows that the NPG update is indeed a regression problem under some linear transformation.

Invariance under reparameterization. The regression view of NPG also provides a way to demonstrate that NPG is invariant under affine reparameterization. An *affine reparameterization* of the policy parameters transforms θ via an invertible affine map:

$$\theta' = M\theta + b$$

where $M \in \mathbb{R}^{d\times d}$ is an *invertible* matrix, and $b \in \mathbb{R}^d$ is an arbitrary translation vector. In other words, the policy can be written as $\pi'(\theta') = \pi(M^{-1}(\theta' - b)) = \pi(\theta)$. Suppose we have $\theta'_t = M\theta_t + b$ and $\pi_{\theta'_t} = \pi_{\theta_t}$ (that is the parameterized policies are the same). By the chain rule of calculus, we have

$$\nabla_{\theta'} \log \pi'_{\theta'}(a \mid s) \mid_{\theta'=\theta'_t} = M^{-1}\nabla_\theta \log \pi_\theta(a \mid s) \mid_{\theta=\theta_t} .$$

Since least squares regression is *invariant under linear transformations* of the feature space, the transformation $\nabla_{\theta'} \log \pi'_{\theta'}(a \mid s) = M^{-1}\nabla_\theta \log \pi_\theta(a \mid s)$ does not change the optimal solution w^* when properly parameterized. In particular, the transformed solution becomes:

$$w'^*_t = Mw^*_t$$

In other words, after an NPG update, we continue to have $\theta'_{t+1} = M\theta_{t+1} + b$. This invariance property is also called *covariant*.

As an exercise, you can show that policy gradient does not enjoy this invariance property.

1.2.3 Soft Policy Iteration

Assume the softmax policy parameterization in the tabular MDP setting: $\pi_\theta(a_h \mid s_h) \propto \exp(\theta_{s_h,a_h})$. Then, we have the following lemma showing the relationship between NPG and SPI.

Lemma 2 *The update rule $\theta_{t+1} \leftarrow \theta_t + \eta H A^t$ is equivalent to*

$$\pi^{t+1}(a_h \mid s_h) \propto \pi^t(a_h \mid s_h) \cdot \exp(\eta H A^t_h(s_h, a_h)).$$

Thus, we can view NPG as running a copy of the hedge algorithm at every state s_h.

To show this result, we can look at the regression view of the NPG algorithm and find out that the advantage function is a solution to the least squares regression problem.

1.3 "Practical" Algorithms

1.3.1 Trust Region Policy Optimization (TRPO)

TRPO is the precursor to PPO, motivated by the NPG update. TRPO solves a constrained optimization problem that explicitly bounds the KL divergence between the updated policy and the current policy:

$$\begin{aligned} \max_{\theta} \quad & \mathbb{E}_{(s,a)\sim d^{\pi_{\theta_t}}} \left[\frac{\pi_\theta(a|s)}{\pi_{\theta_t}(a|s)} A^{\pi_{\theta_t}}(s,a) \right] \\ \text{s.t.} \quad & \mathbb{D}_{\mathrm{KL}}(\pi_{\theta_t} || \pi_\theta) \leq \delta \end{aligned}$$

The objective function is called surrogate advantage objective, which was also used in *conservative policy iteration* covered in a previous lecture. One way to make sense of the objective function is to start with the performance difference lemma:

$$J(\pi_\theta) - J(\pi_{\theta_t}) = H\mathbb{E}_{s\sim d^{\pi_\theta}} \left[\sum_a \left(\pi_\theta(a \mid s) - \pi_{\theta_t}(a \mid s)\right) Q^{\pi_{\theta_t}}(s,a) \right]$$

To ensure policy improvement, we want the performance difference above to be positive. An immediate difficulty for optimizing this quantity is that the expectation is over the state distribution d^{π_θ} induced by the new policy π_θ. Since TRPO ensures that the KL divergence between the new and old policies is small, one strategy is to replace the state distribution by $d^{\pi_{\theta_t}}$. This gives us

$$\begin{aligned} J(\pi_\theta) - J(\pi_{\theta_t}) &= H\mathbb{E}_{s\sim d^{\pi_{\theta_t}}} \left[\sum_a \left(\pi_\theta(a \mid s) - \pi_{\theta_t}(a \mid s)\right) Q^{\pi_{\theta_t}}(s,a) \right] \\ &= H\mathbb{E}_{s\sim d^{\pi_{\theta_t}}, a\sim\pi_{\theta_t}(s)} \left[\left(\frac{\pi_\theta(a \mid s) - \pi_{\theta_t}(a \mid s)}{\pi_{\theta_t}(a \mid s)} \right) Q^{\pi_{\theta_t}}(s,a) \right] \\ &= H\mathbb{E}_{s\sim d^{\pi_{\theta_t}}, a\sim\pi_{\theta_t}(s)} \left[\left(\frac{\pi_\theta(a \mid s)}{\pi_{\theta_t}(a \mid s)} \right) A^{\pi_{\theta_t}}(s,a) \right] \end{aligned}$$

which is exactly the objective function in TRPO (up to a factor of H). Similar to NPG, TRPO will also approximate the KL constraint with the Fisher information matrix, which leads to a quadratic constraint

$$(\theta - \theta_t)^\top F(\theta_t)(\theta - \theta_t) \leq \delta$$

Then TRPO proceeds to solve the quadratically constrained problem by solving the Lagrangian (which we saw in the information theory lecture). Similar to NPG, this step

requires computing $F^{\dagger}\nabla_{\theta}J(\pi_{\theta_t})$, which can be costly due to the computation of a matrix inverse. TRPO instead solves the associated linear system:

$$Fx = \nabla_{\theta}J(\pi_{\theta_t})$$

using the conjugate gradient method, which iteratively finds x without inverting F.

1.3.2 Proximal Policy Optimization (PPO)

PPO is a somewhat "hacky" approximation of TRPO. The motivation is to prevent the policy from changing too much without doing explicitly KL-constrained optimization. Instead, PPO introduces *clipping*:

$$\hat{R}_{\textbf{clip}} = \mathbb{E}_{(s,a)\sim d^{\pi_{\theta_t}}}\left[\underbrace{\text{clip}\left(\frac{\pi_{\theta}(a|s)}{\pi_{\theta_t}(a|s)}\right)A^{\pi_{\theta_t}}(s,a)}_{L(\theta,s,a)}\right]$$

where **clip** projects the value into the interval $[1-\epsilon, 1+\epsilon]$, and ϵ is a small positive hyper-parameter (e.g., 0.1 or 0.2).

The PPO objective then becomes:

$$\hat{R}_{\textbf{PPO}} = \mathbb{E}_{(s,a)\sim d^{\pi_{\theta_t}}}\left[\min\left\{\frac{\pi_{\theta}(a|s)}{\pi_{\theta_t}(a|s)}A^{\pi_{\theta_t}}(s,a), \text{clip}\left(\frac{\pi_{\theta}(a|s)}{\pi_{\theta_t}(a|s)}\right)A^{\pi_{\theta_t}}(s,a)\right\}\right]$$

We analyze the effect of the min operation separately for positive and negative advantage values. Let the probability ratio be defined as:

$$r_{\theta}(s,a) = \frac{\pi_{\theta}(a|s)}{\pi_{\theta_t}(a|s)},$$

Case 1: Positive Advantage ($A(s,a) > 0$) If the advantage is positive, we prefer to increase the probability of the chosen action. The objective $L(\theta, s, a)$ simplifies to:

$$A(s,a)\min\left(r_{\theta}(s,a),\, 1+\varepsilon\right).$$

Analyzing the behavior based on the value of r_{θ}:

$$L(\theta,s,a) = \begin{cases} r_{\theta}(s,a)A(s,a), & \text{if } r_{\theta}(s,a) \leq 1+\varepsilon \\ (1+\varepsilon)A(s,a), & \text{if } r_{\theta}(s,a) > 1+\varepsilon \end{cases}$$

Interpretation:

- If the probability ratio is within the threshold ($\leq 1 + \varepsilon$), the objective increases proportionally as we increase r_θ.
- If r_θ surpasses $1+\varepsilon$, the objective saturates at $(1+\varepsilon)A(s,a)$. Thus, there is no incentive for the policy to further increase this action's probability.

Case 2: Negative Advantage ($A(s,a) < 0$) If the advantage is negative, ideally we want to decrease the probability of the chosen action. The objective is $L(\theta, s, a)$ simplifies to:

$$A(s,a) \max\left(r_\theta(s,a),\, 1-\varepsilon\right) \quad \text{(since } A(s,a) < 0 \text{ reverses the inequality)}.$$

We analyze based on the value of r_θ:

$$L(\theta, s, a) = \begin{cases} r_\theta(s,a)A(s,a), & \text{if } r_\theta(s,a) \geq 1-\varepsilon \\ (1-\varepsilon)A(s,a), & \text{if } r_\theta(s,a) < 1-\varepsilon \end{cases}$$

Interpretation:

- If the probability ratio is within the lower bound ($\geq 1 - \varepsilon$), the objective improves as we reduce r_θ, since this makes $r_\theta A(s,a)$ less negative (larger).
- If r_θ falls below $1 - \varepsilon$, the objective hits the floor $(1 - \varepsilon)A(s,a)$. Thus, reducing the probability further yields no additional improvement.

The min operation together with the clip operation ensures stable and conservative updates by limiting the magnitude of policy changes in each optimization step.