Proximal Policy Optimization with Dynamic Clipping

Student: Rishikesh Vaishnav
Mentor: Sicun Gao

August 13, 2018

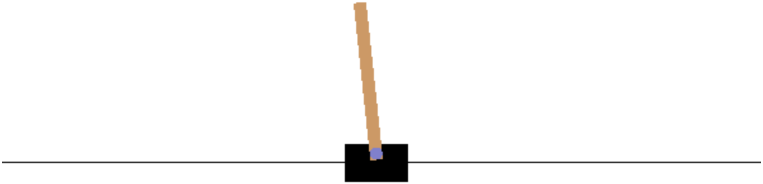
Background

Reinforcement Learning

- A set of algorithms that seek to replicate behavioral learning.
- Basic vocabulary:
  - **Environment**: a general setting with changeable parameters in which actions can be performed that affect these parameters
  - **State** (denoted s): a specific configuration (i.e. "snapshot") of an environment
  - **Agent**: an entity that learns to accomplish a task in a specific evironment
  - **Action** (denoted a): a decision made by the agent that is intended to affect subsequent states
  - **Episode**: a sequence of states and actions in an environment
  - **Reward** (denoted r): a number associated with a state-action pair
- Overall goal: train an agent that picks actions such that the sum of the rewards over an episode is maximimized.

Background (contd.)

- Example: cart-pole demo



Background (contd.)

Policy Gradient Methods

- An agent can be provided with a **policy**, usually denoted π, that completely specifies the probabilty distribution of the action that should be taken at any particular state.
- π is parameterized by some vector θ and can be any function of a state s_t.
- The task of the agent is to learn θ.

Background (contd.)

Generic Policy Gradient Algorithm

Algorithm Generic Policy Gradient

Initialize θ arbitrarily

while True **do** ▷ loop forever

    $\theta_{old} \leftarrow \theta$

    $rollout \leftarrow (s, a, r)$ from multiple π_θ episodes

    Set θ to maximize the loss function $L(rollout, \theta, \theta_{old})$

Background (contd.)

Trust Region Policy Optimization (TRPO)

- The theory behind TRPO suggests using the loss function:

$$L_{\theta_{old}}(\theta) - CD_{KL}^{max}(\theta, \theta_{old})$$

where C is a constant and

$$L_{\theta_{old}}(\theta) = \mathbb{E}_t \left[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)} A_t \right]$$

.

- Using this loss function guarantees monotonic improvement.
- Using the penalty term $CD_{KL}^{max}(\theta, \theta_{old})$ leads to small step sizes in practice, so TRPO uses a hard constraint on the KL divergence.

Background (contd.)

Proximal Policy Optimization (PPO)

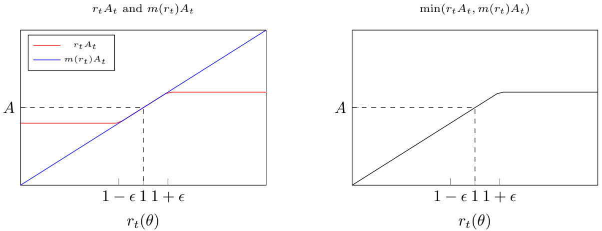
- PPO uses a loss function that is an approximation to the TRPO loss:

$$L^{CLIP}(\theta) = \mathbb{E}\left[\min\left(r_t A_t, \text{clip}(r_t, 1-\epsilon, 1+\epsilon)A_t\right)\right]$$

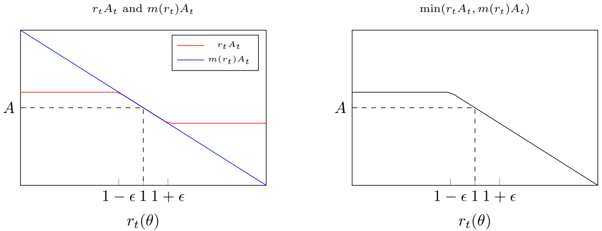
where $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$.

Background (contd.)

- Resultant clipping behavior:



Case $A > 0$



Case $A > 0$

Background (contd.)

- Research question: how can we more precisely control penalties introduced through the clipping objective?

Potential Shortcoming of PPO

- We can separate the loss into its positive and negative components:

$$L^{CLIP}(\theta) = \mathbb{E}_t \left[\min\left(r_t A_t, \mathrm{clip}(r_t, 1-\epsilon, 1+\epsilon)A_t\right)\right]$$
$$= \mathbb{E}_t \left[\begin{cases} \min\left(r_t, \mathrm{clip}(r_t, 1+\epsilon)\right) A_t & A_t > 0 \\ \max\left(r_t, \mathrm{clip}(r_t, 1-\epsilon)\right) A_t & A_t < 0 \end{cases}\right]$$

- Let:

$$r^+_{t,CLIP} = \min\left(r_t, \mathrm{clip}(r_t, 1+\epsilon)\right)$$
$$r^-_{t,CLIP} = \max\left(r_t, \mathrm{clip}(r_t, 1-\epsilon)\right)$$

- Because $\mathbb{E}_t[r_t] = 1$, we know that:

$$\mathbb{E}_t[r^+_{t,CLIP}] < 1$$
$$\mathbb{E}_t[r^-_{t,CLIP}] > 1$$

Potential Shortcoming of PPO (contd.)

- Now, we can define the "expected penalty contributions" of positive and negative advantages:

$$1 - \mathbb{E}_t[r^+_{t,CLIP}]$$

and

$$\mathbb{E}_t[r^-_{t,CLIP}] - 1$$

- Because r_t and A_t are not independent, these expected penalty contributions do not suggest actual penalty contributions.
- However, they can indicate inherent imbalances in the system.

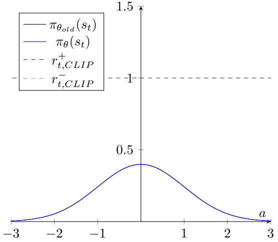
Potential Shortcoming of PPO (contd.)

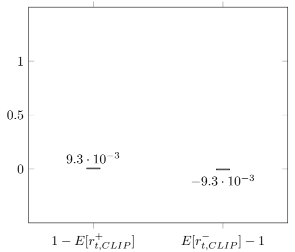
Conceptual Example

- Consider a typical example in reinforcement learning where:
  - We have an agent using a continuous action space (continuous control).
  - The policy is encoded by a gaussian with state-dependent means but constant standard deviation.

Potential Shortcoming of PPO (contd.)

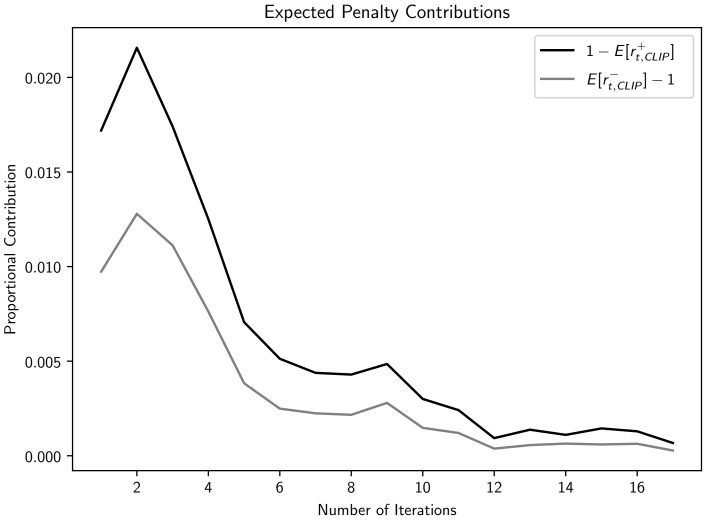
What happens as we learn?





Potential Shortcoming of PPO (contd.)

This discrepancy also appears empirically:



Potential Shortcoming of PPO (contd.)

- The discrepancy between positive and negative penalty contributions is unintentional and highly dependent on the shape of the distribution.
- The goal of this project is to investigate the effects of controlling this discrepancy directly.

Idea

– In the gaussian example, we can precisely calculate the discrepancy at a particular state using the equation:

$$
\begin{aligned}
(1 - E[r^{+}_{t,CLIP}]) - (E[r^{-}_{t,CLIP}] - 1) = \epsilon + (1-\epsilon)\int_{x^-}^{x^+} p(\mu_{old}, x)dx \\
- \left(\int_{x^-}^{x^+} p(\mu, x)dx + 2\epsilon \int_{x^+}^{\infty} p(\mu_{old}, x)dx \right)
\end{aligned}
$$

where

$$x^{+} = \frac{(\mu^2 - \mu_{old}^2) + 2\sigma^2 \ln(1+\epsilon)}{2(\mu - \mu_{old})}$$

and

$$x^{-} = \frac{(\mu^2 - \mu_{old}^2) + 2\sigma^2 \ln(1-\epsilon)}{2(\mu - \mu_{old})}$$

Idea (contd.)

- If we want to minimize this using ϵ, this equation doesn't give us very much control.
- Generalizing to two ϵ by redefining the clipping function as $\text{clip}(r_t, 1-\epsilon^-, 1+\epsilon^+)$:

$$(1 - E[r^+_{t,CLIP}]) - (E[r^-_{t,CLIP}] - 1) = \epsilon^- + (1-\epsilon^-)\int_{x^-}^{x^+} p(\mu_{old}, x)dx \\ - \left(\int_{x^-}^{x^+} p(\mu, x)dx + (\epsilon^+ + \epsilon^-)\int_{x^+}^{\infty} p(\mu_{old}, x)dx\right)$$

- We can also define the total expected penalty contributions:

$$(1 - E[r^+_{t,CLIP}]) + (E[r^-_{t,CLIP}] - 1) = 2\int_{x^+}^{\infty} p(\mu, x)dx - (2 + \epsilon^+ - \epsilon^-)\int_{x^+}^{\infty} p(\mu_{old}, x)dx \\ - \epsilon^- - (1-\epsilon^-)\int_{x^-}^{x^+} p(\mu_{old}, x)dx + \int_{x^-}^{x^+} p(\mu, x)dx$$

Idea (contd.)

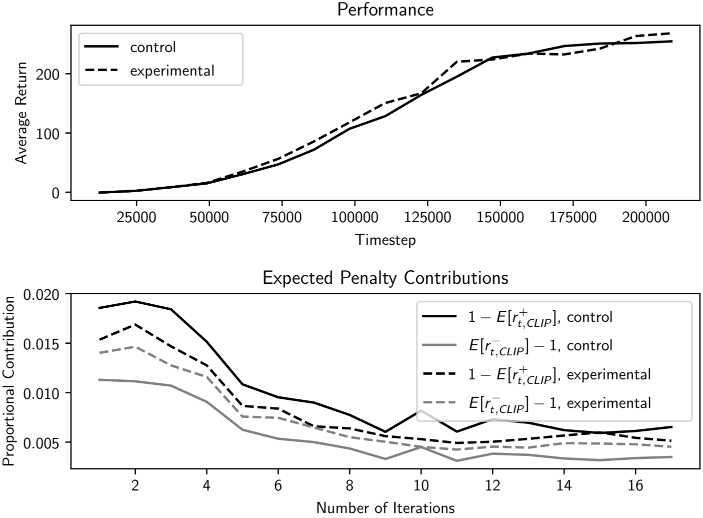
- The new algorithm is identical to PPO, except we use two ϵ, and at the end of every model update, we optimize ϵ^- and ϵ^+ so that:
  - $(1 - E[r^+_{t,CLIP}]) - (E[r^-_{t,CLIP}] - 1)$ is minimized.
  - $(1 - E[r^+_{t,CLIP}]) + (E[r^-_{t,CLIP}] - 1)$ remains the same.

Results

- Overall, we observed approximately the same performance or modest improvements.
- The optimization was always successful in reducing the empirical discrepancy.

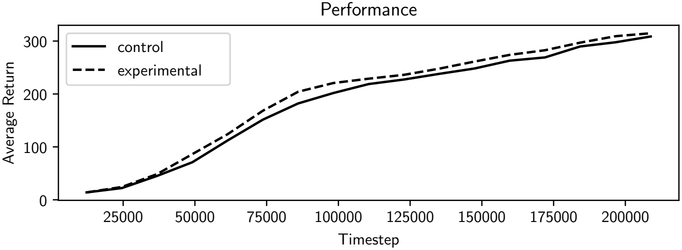
Results (contd.)

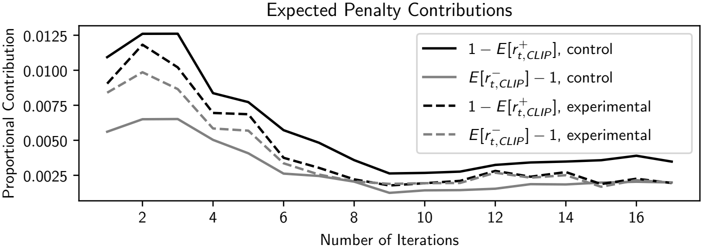
Example: Walker2d-v2 environment



Results (contd.)

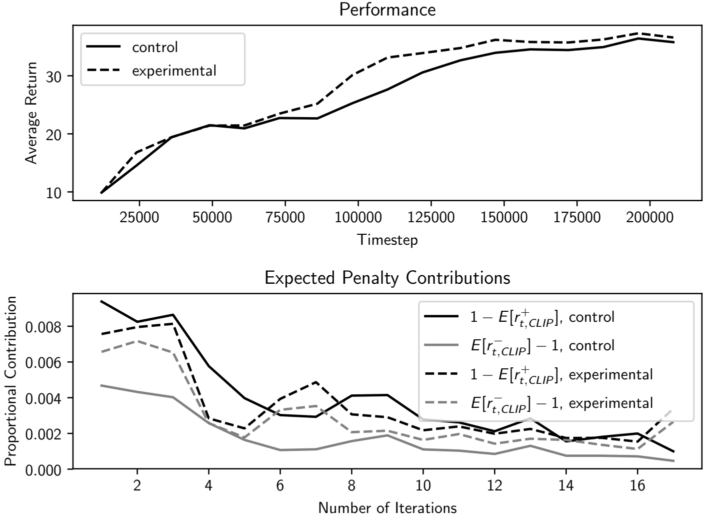
Example: Hopper-v2 environment





Results (contd.)

Example: Swimmer-v2 environment



Future Directions

Some questions:

- Is there a simpler, problem-independent way to control the discrepancy?
- How does increasing the discrepancy in some direction affect performance?
- How, specifically, does the expected discrepancy relate to the actual penalty difference?