# Policy Gradient

**Insoon Yang**

Department of Electrical and Computer Engineering
Seoul National University

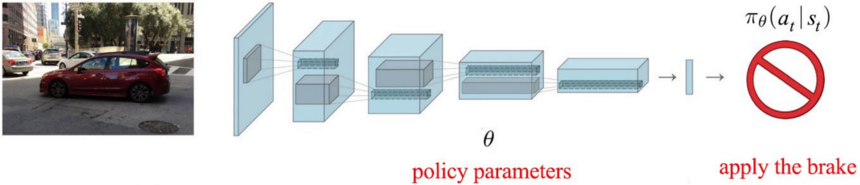
CORE
Control + Optimization Research Lab

# Parameterizing Policy

Q) What's the meaning of $\pi(a|s)$?

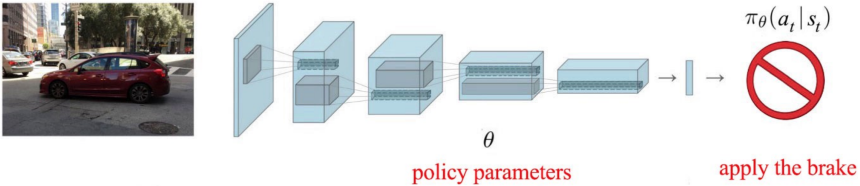
# Parameterizing Policy

Q) What's the meaning of $\pi(a|s)$?



# Parameterizing Policy

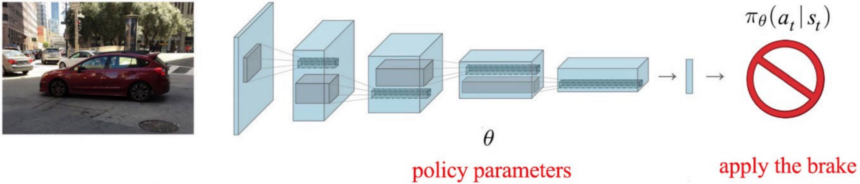
Q) What's the meaning of $\pi(a|s)$?




Q) How can we find a good $\pi(a|s)$, which is a **function**?

# Parameterizing Policy

Q) What's the meaning of $\pi(a|s)$?

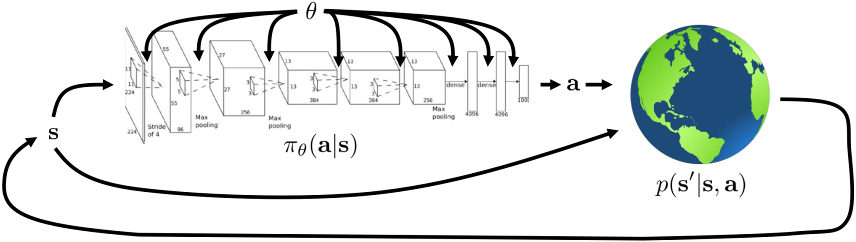



Q) How can we find a good $\pi(a|s)$, which is a **function**?

Idea:

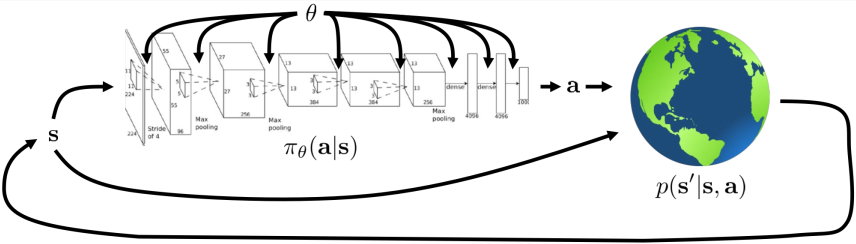
- Parameterize policy by a parameter vector $\theta \in \mathbb{R}^\ell$: $\pi_\theta(a|s)$
- Find an optimal $\theta$

# How to find optimal parameters $\theta$?




[S. Levine, CS285]

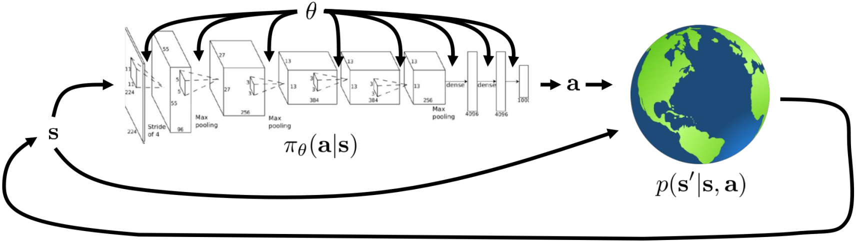
# How to find optimal parameters $\theta$?




[S. Levine, CS285]

- Let $\tau := (s_0, a_0, \ldots, s_T, a_T)$ denote the state-action trajectory

# How to find optimal parameters $\theta$?

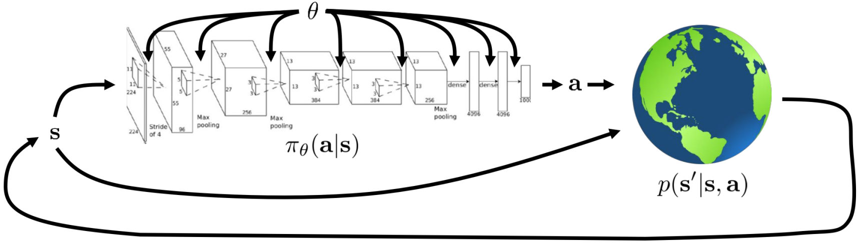



[S. Levine, CS285]

- Let $\tau := (s_0, a_0, \ldots, s_T, a_T)$ denote the state-action trajectory
- By Markov property,

$$p_\theta(\tau) = p(s_0) \prod_{t=0}^{T} \pi_\theta(a_t|s_t) p(s_{t+1}|s_t, a_t)$$

# How to find optimal parameters $\theta$?




[S. Levine, CS285]

- Let $\tau := (s_0, a_0, \ldots, s_T, a_T)$ denote the state-action trajectory
- By Markov property,

$$p_\theta(\tau) = p(s_0) \prod_{t=0}^{T} \pi_\theta(a_t|s_t) p(s_{t+1}|s_t, a_t)$$

- Approximate MDP problem:

$$\max_\theta \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[ \sum_t r(s_t, a_t) \right] =: J(\theta)$$

# Basics of Optimization

$$\max_{\theta} \; J(\theta)$$

Gradient ascent:

# Basics of Optimization

$$\max_{\theta} \; J(\theta)$$

Gradient ascent:

1. Initialize $\theta_0$;

# Basics of Optimization

$$\max_{\theta} \; J(\theta)$$

Gradient ascent:

1. Initialize $\theta_0$;
2. For $k = 1, 2, \ldots$ until converges, do

# Basics of Optimization

$$\max_{\theta} \; J(\theta)$$

Gradient ascent:

1. Initialize $\theta_0$;
2. For $k = 1, 2, \ldots$ until converges, do
   - Set
   $$\theta_{k+1} \leftarrow \theta_k + \alpha \nabla_\theta J(\theta_k),$$
   where

# Basics of Optimization

$$\max_{\theta} \; J(\theta)$$

Gradient ascent:

1. Initialize $\theta_0$;
2. For $k = 1, 2, \ldots$ until converges, do
   - Set
   $$\theta_{k+1} \leftarrow \theta_k + \alpha \nabla_\theta J(\theta_k),$$
   where
   $$\begin{cases} \alpha : \ \text{stepsize} \\ \nabla_\theta J(\theta_k) : \ \text{gradient of } J \text{ at } \theta_k \end{cases}$$

# Basics of Optimization

$$\max_{\theta} \; J(\theta)$$

Gradient ascent:

1. Initialize $\theta_0$;
2. For $k = 1, 2, \ldots$ until converges, do
   - Set
     $$\theta_{k+1} \leftarrow \theta_k + \alpha \nabla_\theta J(\theta_k),$$
     where
     $$\begin{cases} \alpha : \text{ stepsize} \\ \nabla_\theta J(\theta_k) : \text{ gradient of } J \text{ at } \theta_k \end{cases}$$
   - Set $k \leftarrow k + 1$;

# How to find the gradient $\nabla_\theta J(\theta)$?

Recall that

$$J(\theta) := \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[ \sum_t r(s_t, a_t) \right]$$

# How to find the gradient $\nabla_\theta J(\theta)$?

Recall that

$$J(\theta) := \mathbb{E}_{\tau \sim p_\theta(\tau)}\left[\sum_t r(s_t, a_t)\right]$$

## How to find the gradient $\nabla_\theta J(\theta)$?

Recall that

$$J(\theta) := \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[ \sum_t r(s_t, a_t) \right]$$

- Let $r(\tau) := \sum_t r(s_t, a_t)$

## How to find the gradient $\nabla_\theta J(\theta)$?

Recall that

$$J(\theta) := \mathbb{E}_{\tau \sim p_\theta(\tau)}\left[\sum_t r(s_t, a_t)\right]$$

- Let $r(\tau) := \sum_t r(s_t, a_t)$
- Rewrite

$$J(\theta) = \int p_\theta(\tau) r(\tau) d\tau$$

## How to find the gradient $\nabla_\theta J(\theta)$?

Recall that

$$J(\theta) := \mathbb{E}_{\tau \sim p_\theta(\tau)}\left[\sum_t r(s_t, a_t)\right]$$

- Let $r(\tau) := \sum_t r(s_t, a_t)$
- Rewrite

$$J(\theta) = \int p_\theta(\tau) r(\tau) d\tau$$

- Differentiate $J$ w.r.t $\theta$:

$$\nabla_\theta J(\theta) = \int \nabla_\theta p_\theta(\tau) r(\tau) d\tau$$

## Can we simplify the gradient $\nabla_\theta J(\theta)$?

# Can we simplify the gradient $\nabla_\theta J(\theta)$?

Note that

$$\nabla_\theta \log p_\theta(\tau) = \frac{\nabla_\theta p_\theta(\tau)}{p_\theta(\tau)}$$

# Can we simplify the gradient $\nabla_\theta J(\theta)$?

Note that

$$\nabla_\theta \log p_\theta(\tau) = \frac{\nabla_\theta p_\theta(\tau)}{p_\theta(\tau)}$$

Therefore, the gradient can be written as

$$\begin{aligned}
\nabla_\theta J(\theta) &= \int \nabla_\theta p_\theta(\tau) r(\tau) d\tau \\
&= \int p_\theta(\tau) \nabla_\theta \log p_\theta(\tau) r(\tau) d\tau \\
&= \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[ \nabla_\theta \log p_\theta(\tau) r(\tau) \right]
\end{aligned}$$

## Can we further simplify the gradient $\nabla_\theta J(\theta)$?

## Can we further simplify the gradient $\nabla_\theta J(\theta)$?

Note that

$$\begin{aligned}
\log p_\theta(\tau) &= \log \left[ p(s_0) \prod_{t=0}^{T} \pi_\theta(a_t|s_t) p(s_{t+1}|s_t, a_t) \right] \\
&= \log p(s_0) + \sum_{t=0}^{T} \log \pi_\theta(a_t|s_t) + \log p(s_{t+1}|s_t, a_t)
\end{aligned}$$

## Can we further simplify the gradient $\nabla_\theta J(\theta)$?

Note that

$$\begin{aligned}
\log p_\theta(\tau) &= \log \left[ p(s_0) \prod_{t=0}^{T} \pi_\theta(a_t|s_t) p(s_{t+1}|s_t, a_t) \right] \\
&= \log p(s_0) + \sum_{t=0}^{T} \log \pi_\theta(a_t|s_t) + \log p(s_{t+1}|s_t, a_t)
\end{aligned}$$

Therefore, the gradient can be written as

$$\begin{aligned}
\nabla_\theta J(\theta) &= \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[ \nabla_\theta \log p_\theta(\tau) r(\tau) \right] \\
&= \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[ \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) r(\tau) \right] \\
&= \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[ \left( \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \right) \left( \sum_{t=0}^{T} r(s_t, a_t) \right) \right]
\end{aligned}$$

# Evaluating the policy gradient

# Evaluating the policy gradient

- So far, we have the **policy gradient theorem**:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim p_\theta(\tau)}\left[\left(\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t)\right)\left(\sum_{t=0}^{T} r(s_t, a_t)\right)\right]$$

# Evaluating the policy gradient

- So far, we have the **policy gradient theorem**:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[ \left( \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t | s_t) \right) \left( \sum_{t=0}^{T} r(s_t, a_t) \right) \right]$$

- REINFORCE algorithm: using empirical estimate of $\mathbb{E}_{\tau \sim p_\theta(\tau)}$

Machine Learning, 8, 229–256 (1992)


Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning

RONALD J. WILLIAMS rjw@corwin.ccs.northeastern.edu
*College of Computer Science, 161 CN, Northeastern University, 360 Huntington Ave., Boston, MA 02115*

# REINFORCE algorithm

Initialize $\theta$;

1. Sample $\{\tau^i\}_{i=1}^N := \{(s_0^i, a_0^i, \ldots, s_T^i, a_T^i)\}_{i=1}^N$ using the current policy $\pi_\theta(a_t|s_t)$

2. Estimate the gradient

$$\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_{i=1}^N \left( \sum_{t=0}^T \nabla_\theta \log \pi_\theta(a_t^i|s_t^i) \right) \left( \sum_{t=0}^T r(s_t^i, a_t^i) \right)$$

3. Perform gradient ascent:

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta);$$

# Example: Gaussian Policy

Set

$$\pi_\theta(\cdot|s_t) \sim \mathcal{N}(f_{NN}(s_t); \Sigma)$$

# Example: Gaussian Policy

Set

$$\pi_\theta(\cdot|s_t) \sim \mathcal{N}(f_{NN}(s_t); \Sigma)$$

In other words,

$$\pi_\theta(a_t|s_t) = \frac{1}{\sqrt{\det(2\pi\Sigma}} \exp\left( -\frac{1}{2}(a_t - f_{NN}(s_t))^\top \Sigma^{-1}(a_t - f_{NN}(s_t))\right)$$

# Example: Gaussian Policy

Set

$$\pi_\theta(\cdot|s_t) \sim \mathcal{N}(f_{NN}(s_t); \Sigma)$$

In other words,

$$\pi_\theta(a_t|s_t) = \frac{1}{\sqrt{\det(2\pi\Sigma}} \exp\left( -\frac{1}{2}(a_t - f_{NN}(s_t))^\top \Sigma^{-1}(a_t - f_{NN}(s_t))\right)$$

Therefore,

- $\log \pi_\theta(a_t|s_t) = -\frac{1}{2}(a_t - f_{NN}(s_t))^\top \Sigma^{-1}(a_t - f_{NN}(s_t)) + \text{constant}$

# Example: Gaussian Policy

Set

$$\pi_\theta(\cdot|s_t) \sim \mathcal{N}(f_{NN}(s_t); \Sigma)$$

In other words,

$$\pi_\theta(a_t|s_t) = \frac{1}{\sqrt{\det(2\pi\Sigma}} \exp\left( -\frac{1}{2}(a_t - f_{NN}(s_t))^\top \Sigma^{-1}(a_t - f_{NN}(s_t))\right)$$

Therefore,

- $\log \pi_\theta(a_t|s_t) = -\frac{1}{2}(a_t - f_{NN}(s_t))^\top \Sigma^{-1}(a_t - f_{NN}(s_t)) + \text{constant}$
- $\nabla_\theta \log \pi_\theta(a_t|s_t) = \frac{1}{2}(a_t - f_{NN}(s_t))\nabla_\theta f_{NN}(s_t)$

# Advantages and Disadvantages

Advantages:

- Simple
- Unbiased gradient
- Locally optimal solution

# Advantages and Disadvantages

Advantages:

- Simple
- Unbiased gradient
- Locally optimal solution

Disadvantages:

- High variance of the gradient
- On policy: Must use the most recent policy
  (Huge # of samples required)

## How to reduce variance?

# How to reduce variance?

1. Increase the batch size

## How to reduce variance?

1. Increase the batch size
2. Use a baseline, $b$, not related to $\theta$:

$$
\begin{aligned}
&\mathbb{E}_{\tau\sim p_\theta}[\nabla_\theta \log p_\theta(\tau)(r(\tau)-b)] = \nabla_\theta J(\theta) - \mathbb{E}_{\tau\sim p_\theta}[\nabla_\theta \log p_\theta(\tau) b] \\
&= \nabla_\theta J(\theta) - \int p_\theta(\tau)\nabla_\theta \log p_\theta(\tau) b d\tau \\
&= \nabla_\theta J(\theta) - \int \nabla_\theta p_\theta(\tau) b d\tau \\
&= \nabla_\theta J(\theta) - b\nabla_\theta \int p_\theta(\tau) d\tau \\
&= \nabla_\theta J(\theta) - b\nabla_\theta 1 \\
&= \nabla_\theta J(\theta)
\end{aligned}
$$

$\Longrightarrow$ Subtracting a baseline $b$ is unbiased in expectation!

# Why baseline helps to reduce variance?

# Why baseline helps to reduce variance?

No baseline:

$$\mathrm{Var}[\nabla_\theta J^{NB}(\theta)] = \mathbb{E}[(\nabla_\theta \log p_\theta(\tau) r(\tau))^2] - \mathbb{E}[\nabla_\theta \log p_\theta(\tau) r(\tau)]^2$$

# Why baseline helps to reduce variance?

No baseline:

$$\mathrm{Var}[\nabla_\theta J^{NB}(\theta)] = \mathbb{E}[(\nabla_\theta \log p_\theta(\tau) r(\tau))^2] - \mathbb{E}[\nabla_\theta \log p_\theta(\tau) r(\tau)]^2$$

With baseline:

$$\begin{aligned}\mathrm{Var}[\nabla_\theta J^{B}(\theta)] &= \mathbb{E}[(\nabla_\theta \log p_\theta(\tau)(r(\tau) - b))^2] - \mathbb{E}[\nabla_\theta \log p_\theta(\tau)(r(\tau) - b)]^2 \\ &= \mathbb{E}[(\nabla_\theta \log p_\theta(\tau)(r(\tau) - b))^2] - \mathbb{E}[\nabla_\theta \log p_\theta(\tau) r(\tau)]^2\end{aligned}$$

# Why baseline helps to reduce variance?

No baseline:

$$\mathrm{Var}[\nabla_\theta J^{NB}(\theta)] = \mathbb{E}[(\nabla_\theta \log p_\theta(\tau) r(\tau))^2] - \mathbb{E}[\nabla_\theta \log p_\theta(\tau) r(\tau)]^2$$

With baseline:

$$\begin{aligned}\mathrm{Var}[\nabla_\theta J^{B}(\theta)] &= \mathbb{E}[(\nabla_\theta \log p_\theta(\tau)(r(\tau) - b))^2] - \mathbb{E}[\nabla_\theta \log p_\theta(\tau)(r(\tau) - b)]^2 \\ &= \mathbb{E}[(\nabla_\theta \log p_\theta(\tau)(r(\tau) - b))^2] - \mathbb{E}[\nabla_\theta \log p_\theta(\tau) r(\tau)]^2\end{aligned}$$

Therefore,

$$\mathrm{Var}[\nabla_\theta J^{B}(\theta)] \leq \mathrm{Var}[\nabla_\theta J^{NB}(\theta)]$$

if $b \in [0, 2r(\tau)]$.

# Which baseline to choose?

- Recall

$$\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_i \left( \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t^i | s_t^i) \right) \left( \sum_{t=0}^{T} r(s_t^i, a_t^i) \right)$$

- Further approximate it by

$$\begin{aligned} \nabla_\theta J(\theta) &\approx \frac{1}{N} \sum_i \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t^i | s_t^i) \sum_{t'=t}^{T} r(s_{t'}^i, a_{t'}^i) \\ &= \frac{1}{N} \sum_i \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t^i | s_t^i) Q(s_t^i, a_t^i) \end{aligned}$$

- Choose baseline $b := v(s_t^i)$:

$$\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_i \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t^i | s_t^i) \underbrace{\left[ Q(s_t^i, a_t^i) - v(s_t^i) \right]}_{=: A(s_t^i, a_t^i)}$$

# Intuition behind policy gradient with baseline

1. Case I: Trajectory A receives +10 rewards and Trajectory B receives -10 rewards
2. Case II: Trajectory A receives +10 rewards and Trajectory B receives +1 rewards

# Intuition behind policy gradient with baseline

1. Case I: Trajectory A receives $+10$ rewards and Trajectory B receives -10 rewards
2. Case II: Trajectory A receives $+10$ rewards and Trajectory B receives $+1$ rewards

$\Longrightarrow$ PG will increase the probability of both trajectories in Case II

# Intuition behind policy gradient with baseline

1. Case I: Trajectory A receives $+10$ rewards and Trajectory B receives -10 rewards
2. Case II: Trajectory A receives $+10$ rewards and Trajectory B receives $+1$ rewards

$\Longrightarrow$ PG will increase the probability of both trajectories in Case II

Now, Consider $\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_i \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t^i | s_t^i) [Q(s_t^i, a_t^i) - b]$ with baseline $b = 5$

# Intuition behind policy gradient with baseline

1. Case I: Trajectory A receives $+10$ rewards and Trajectory B receives -10 rewards
2. Case II: Trajectory A receives $+10$ rewards and Trajectory B receives $+1$ rewards

$\Longrightarrow$ PG will increase the probability of both trajectories in Case II

Now, Consider $\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_i \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t^i | s_t^i)[Q(s_t^i, a_t^i) - b]$ with baseline $b = 5$

1. Case I: Trajectory A receives $+5$ rewards and Trajectory B receives -15 rewards
2. Case II: Trajectory A receives $+5$ rewards and Trajectory B receives -4 rewards

# Intuition behind policy gradient with baseline

1. Case I: Trajectory A receives +10 rewards and Trajectory B receives -10 rewards
2. Case II: Trajectory A receives +10 rewards and Trajectory B receives +1 rewards

$\Longrightarrow$ PG will increase the probability of both trajectories in Case II

Now, Consider $\nabla_\theta J(\theta) \approx \frac{1}{N} \sum_i \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t^i | s_t^i)[Q(s_t^i, a_t^i) - b]$ with baseline $b = 5$

1. Case I: Trajectory A receives +5 rewards and Trajectory B receives -15 rewards
2. Case II: Trajectory A receives +5 rewards and Trajectory B receives -4 rewards

$\Longrightarrow$ PG will increase the probability of Trajectory A but decrease the probability of Trajectory B

# Practical Tips

# Practical Tips

- Keep in mind that gradient will be very noisy

# Practical Tips

- Keep in mind that gradient will be very noisy
- Use much larger batches ($100\times$ larger than DQN)

# Practical Tips

- Keep in mind that gradient will be very noisy
- Use much larger batches ($100\times$ larger than DQN)
- Tuning learning rates is very hard
  Adaptive size rule like Adam can be fine (but not the best)

# Practical Tips

- Keep in mind that gradient will be very noisy
- Use much larger batches ($100\times$ larger than DQN)
- Tuning learning rates is very hard
  Adaptive size rule like Adam can be fine (but not the best)
- Use Actor-Critic with advanced PG methods
  Will learn DDPG, TRPO, SAC