Lecture 9: Deep Reinforcement Learning

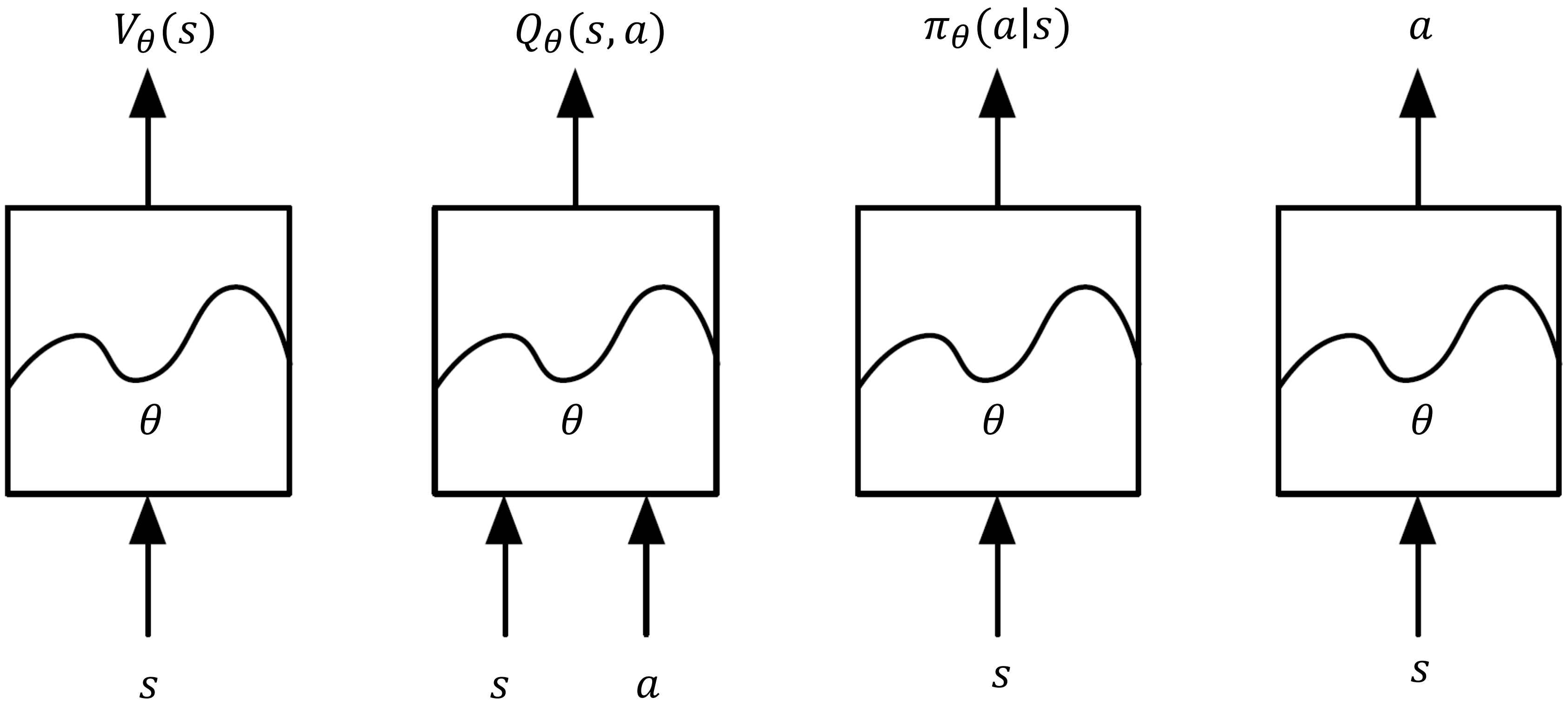
Shuai Li

John Hopcroft Center, Shanghai Jiao Tong University

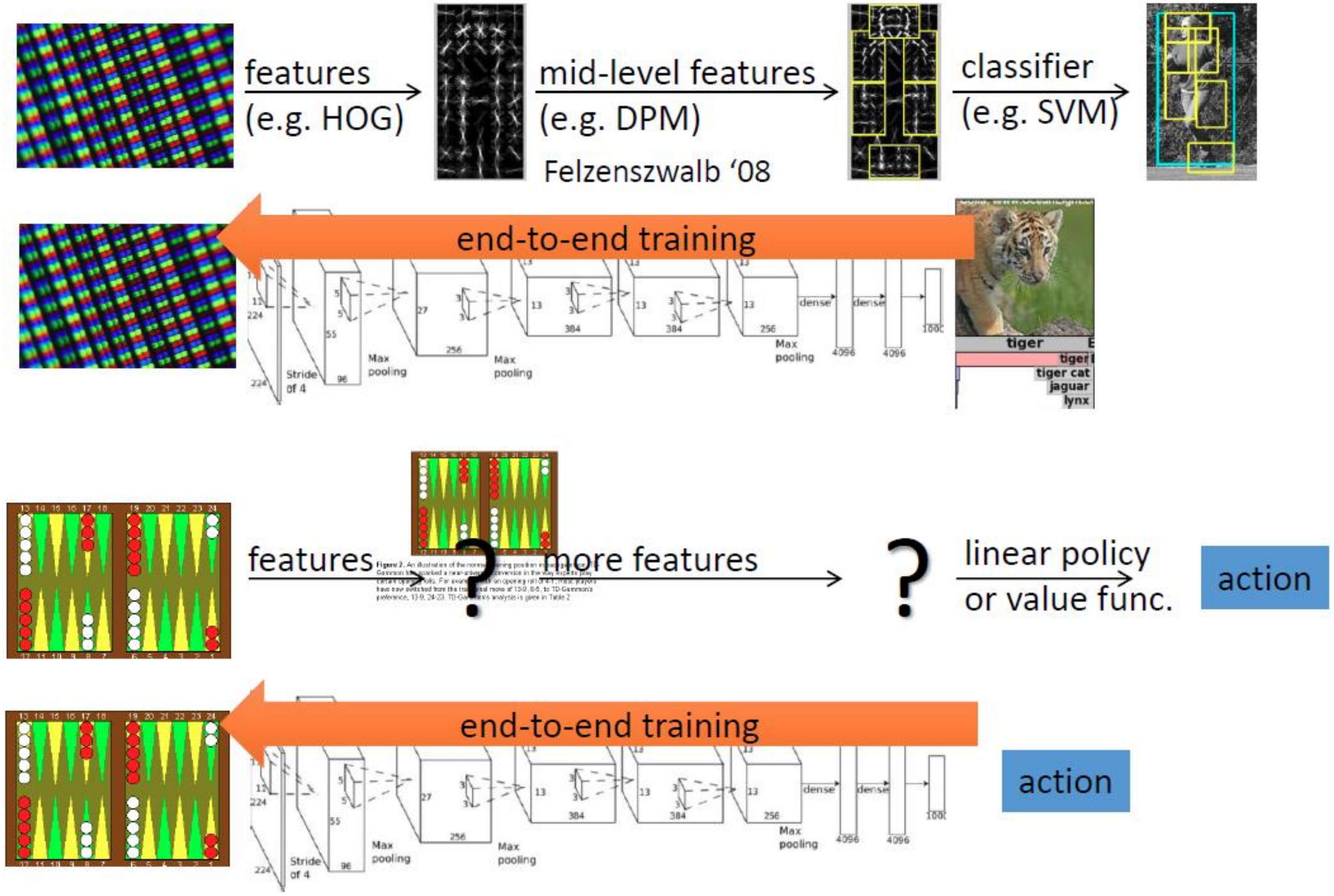
https://shuaili8.github.io

https://shuaili8.github.io/Teaching/CS3317/index.html

Reinforcement Learning w/ Function Approximation

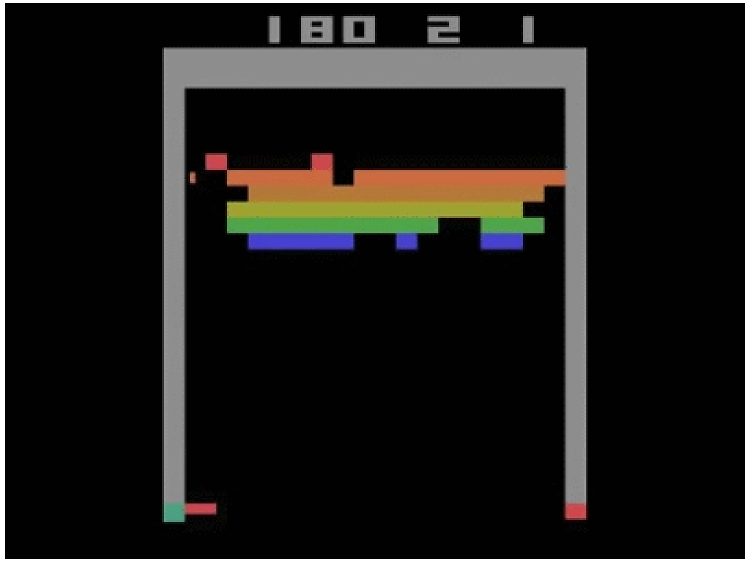


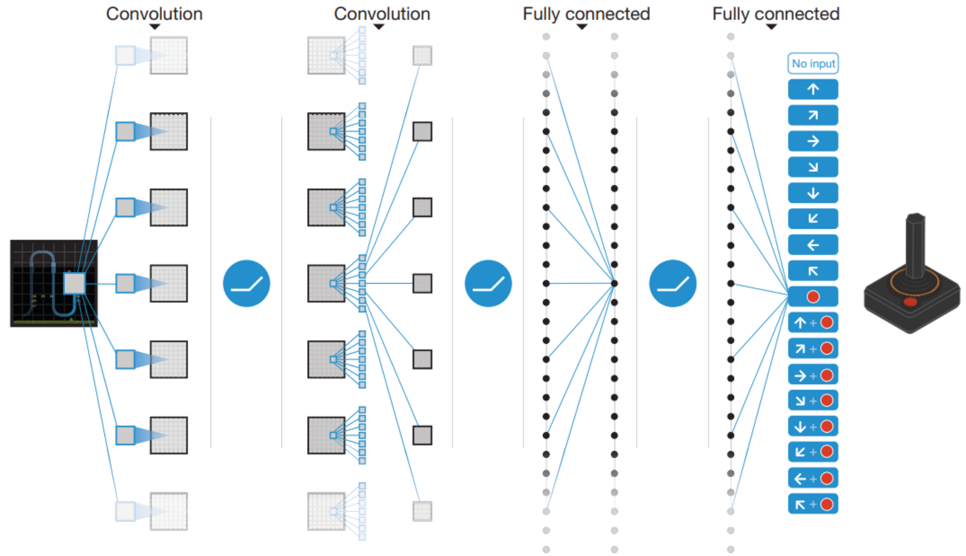
End-to-end Training of RL



Deep Reinforcement Learning

- Use Neural Network to approximate Value and Policy
- To make RL training end-to-end





Volodymyr Mnih, Koray Kavukcuoglu, David Silver et al. Playing Atari with Deep Reinforcement Learning. NIPS 2013 workshop.

Chellanges of DRL

- What would happen if we combine Deep Learning and RL?
  - Value function and policy now become deep network
  - High dimensional parameters
  - Unstanble training
  - Easily overfit
  - Require large amount of data
  - High computing power
  - Trade-off between CPU (for collecting data) and GPU (for training NN)
  - …

These new problems advance the development of DRL

Deep Q-Network

- TD Q-value Learning with parametrized $Q_\theta(s,a)$
  - Target sample: $y_{\mathrm{t}} = r_t + \gamma \max_{a'} Q_\theta(s_{t+1}, a')$
  - Learning objective:

no gradient

$$\theta^\star \leftarrow \arg\min_\theta \frac{1}{2} \sum_{(s_t, a_t) \in D} \left(Q_\theta(s_t, a_t) - \left(r + \gamma \max_{a'} Q_\theta(s_{t+1}, a')\right)\right)^2$$

  - Update $Q_\theta(s_t, a_t) \leftarrow Q_\theta(s_t, a_t) + \alpha\left(r_t + \gamma \max_{\alpha'} Q_\theta(s_{t+1}, a') - Q_\theta(s_t, a_t)\right)$

Deep Q-Network 2

- Challenges: Use NN to approximate $Q_\theta(s, a)$
  - Training of the algorithm is unstable
  - $\{(s_t, a_t, s_{t+1}, r_t)\}$ not i.i.d.
  - Frequent update of $Q_\theta(s, a)$
- Solution:
  - Experience replay
  - Target network and evaluation network

Experience Replay

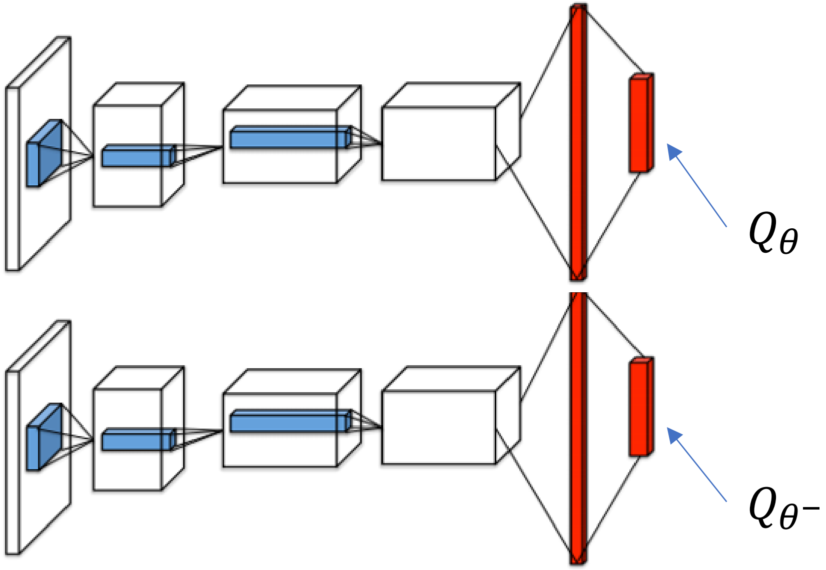
- Store every $e_t = (s_t, a_t, s_{t+1}, r_t)$ in a replay buffer D, then sample uniformly

- Prioritized sampling
  - Compute priority score $p_t = \ |r_t + \gamma \max_{a'} Q_\theta(s_{t+1}, a') - Q_\theta(s_t, a_t)|$
  - Store $e_t = (s_t, a_t, s_{t+1}, r_t, p_t + \epsilon)$
  - Sample e_t with probability $P(t) = \frac{p_t^\alpha}{\sum_k p_k^\alpha}$
  - Update with importance weight $\omega_t = \frac{\left(N \times P(t)\right)^{-\beta}}{\max_i \omega_i}$

Target Network

- Target network $Q_{\theta^-}(s, a)$
  - Use old network to set target value, sync to evaluation network every C updates

$$L_i(\theta_i) = \mathbb{E}_{s_t, a_t, s_{t+1}, r_t, p_t \sim D}\left[\frac{1}{2}\, \omega_t \big(\underbrace{r_t + \gamma \max_{a'} Q_{\theta_i^-}(s_{t+1}, a')}_{target} - Q_{\theta_i}(s_t, a_t)\big)^2\right]$$



DQN Algorithm

Algorithm 1 Double DQN with proportional prioritization

Input: minibatch k, step-size η, replay period K and size N, exponents α and β, budget T.
Initialize replay memory $\mathcal{H} = \emptyset$, $\Delta = 0$, $p_1 = 1$
Observe S_0 and choose $A_0 \sim \pi_\theta(S_0)$
for $t = 1$ **to** T **do**
    Observe S_t, R_t, γ_t
    Store transition $(S_{t-1}, A_{t-1}, R_t, \gamma_t, S_t)$ in $\mathcal{H}$ with maximal priority $p_t = \max_{i<t} p_i$
    if $t \equiv 0 \mod K$ **then**
        for $j = 1$ **to** k **do**
            Sample transition $j \sim P(j) = p_j^\alpha / \sum_i p_i^\alpha$
            Compute importance-sampling weight $w_j = (N \cdot P(j))^{-\beta} / \max_i w_i$
            Compute TD-error $\delta_j = R_j + \gamma_j Q_{\text{target}}(S_j, \arg\max_a Q(S_j, a)) - Q(S_{j-1}, A_{j-1})$
            Update transition priority $p_j \leftarrow |\delta_j|$
            Accumulate weight-change $\Delta \leftarrow \Delta + w_j \cdot \delta_j \cdot \nabla_\theta Q(S_{j-1}, A_{j-1})$
        end for
        Update weights $\theta \leftarrow \theta + \eta \cdot \Delta$, reset $\Delta = 0$
        From time to time copy weights into target network $\theta_{\text{target}} \leftarrow \theta$
    end if
    Choose action $A_t \sim \pi_\theta(S_t)$
end for

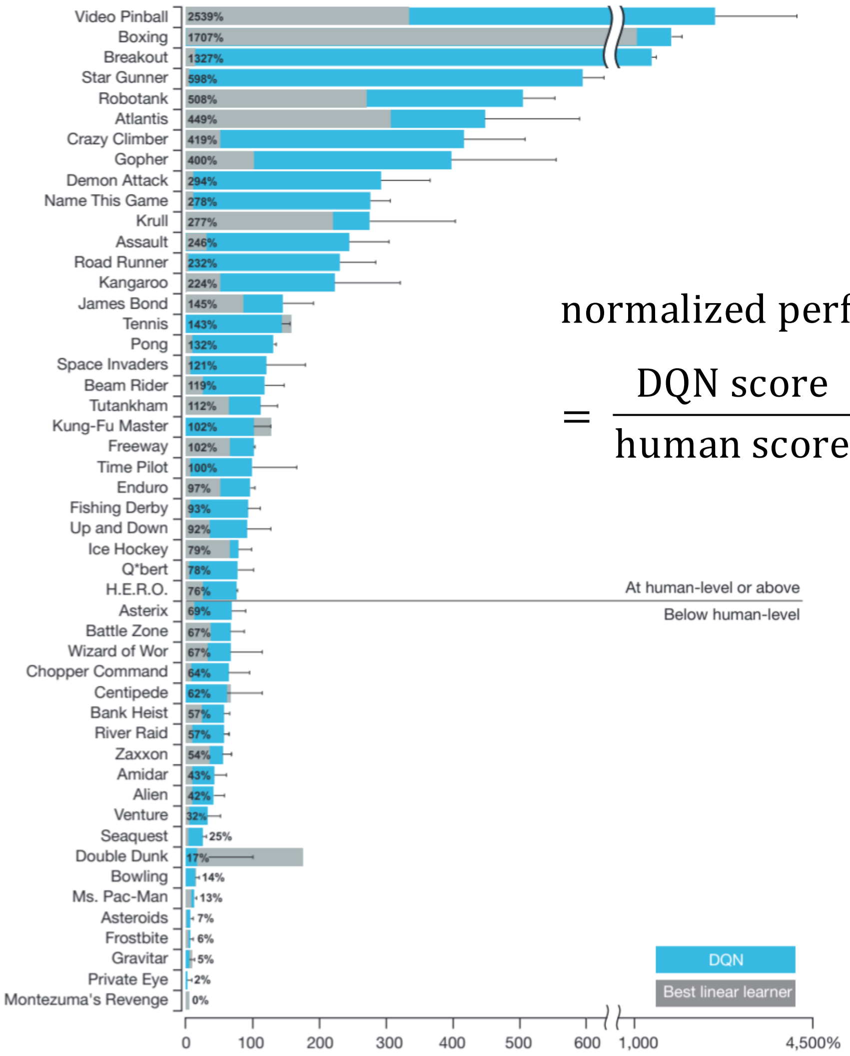
Prioritized Sampling

Importance sampling

Learning objective is uniform distribution

"Prioritized Experience Replay", Schaul et al. (2016)

Results on Atari Environments



$$\text{normalized performance} = \frac{\text{DQN score} - \text{random play score}}{\text{human score} - \text{random play score}}$$

The performance of DQN is normalized with respect to a professional human games tester (that is, 100% level)

"Human-Level Control Through Deep Reinforcement Learning", Mnih, Kavukcuoglu, Silver et al. (2015)

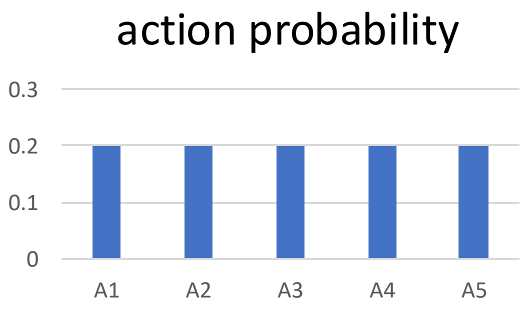
Policy Parametrization

- Parametrized policy $\pi_\theta(a|s)$
  - Deterministic policy $a = \pi_\theta(s)$
  - Random policy $\pi_\theta(a|s) = P(a|s;\theta)$
- Could generalize to unseen states
- Advantage:
  - Good convergence property
  - Effective in high-dimensional space or continuous action space
- Disadvantage:
  - Usually converge to a local (not global) optimum
  - High variance to evaluate policy

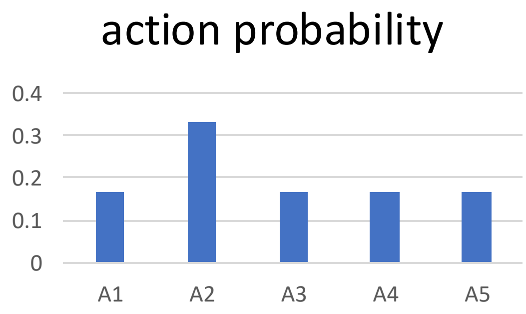
Policy Gradient

- For random policy $\pi_\theta(a|s) = P(a|s;\theta)$
- We should
  - decrease probability of bad actions
  - increase probability of good actions
- Example: A discrete 5-action space

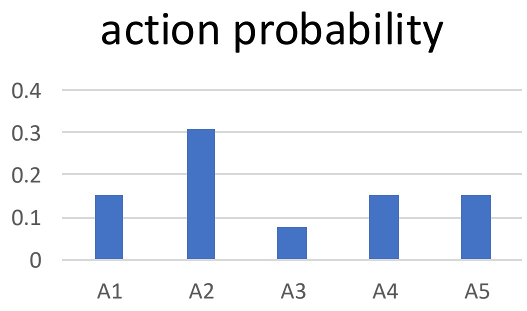
1. Initialize θ



3. Update θ



5. Update θ



2. Select A2
Observe positive reward

4. Select A3
Observe negative reward

Policy Gradient for 1-step MDP

- Consider 1-step MDP
  - Starting state $s \sim d(s)$
  - Selects action a and stops. Receive reward r_{sa}
- Expected utility of the policy

$$J(\theta) = \mathbb{E}_{\pi_\theta}[r] = \sum_{s \in S} d(s) \sum_{a \in A} \pi_\theta(a|s) r_{sa}$$

  and its gradient

$$\frac{\partial J(\theta)}{\partial \theta} = \sum_{s \in S} d(s) \sum_{a \in A} \frac{\partial \pi_\theta(a|s)}{\partial \theta} r_{sa}$$

Policy Gradient for 1-step MDP 2

- $\frac{\partial J(\theta)}{\partial \theta} = \sum_{s \in S} d(s) \sum_{a \in A} \frac{\partial \pi_\theta(a|s)}{\partial \theta} r_{sa}$
- $= \sum_{s \in S} d(s) \sum_{a \in A} \pi_\theta(a|s) \frac{1}{\pi_\theta(a|s)} \frac{\partial \pi_\theta(a|s)}{\partial \theta} r_{sa}$
- $= \sum_{s \in S} d(s) \sum_{a \in A} \pi_\theta(a|s) \frac{\partial \log \pi_\theta(a|s)}{\partial \theta} r_{sa}$
- $= \mathbb{E}_{\pi_\theta}\left[\frac{\partial \log \pi_\theta(a|s)}{\partial \theta} r_{sa}\right]$

could be replaced with samples

Policy Gradient Theorem

- Extend previous result to multi-step MDP

- Theorem. For differentiable $\pi_\theta(a|s)$, with averaged return or discounted return J, its policy gradient is

$$\frac{\partial J(\theta)}{\partial \theta} = \mathbb{E}_{\pi_\theta}\left[\frac{\partial \log \pi_\theta(a|s)}{\partial \theta} Q^{\pi_\theta}(s,a)\right]$$

$\partial \log \pi_\theta(a|s)$ for softmax policy

- $\pi_\theta(a|s) = \frac{e^{f_\theta(s,a)}}{\sum_{a'} e^{f_\theta(s,a')}}$

- $\frac{\partial \log \pi_\theta(a|s)}{\partial \theta} = \frac{\partial f_\theta(s,a)}{\partial \theta} - \frac{1}{\sum_{a'} e^{f_\theta(s,a')}} \sum_{a''} e^{f_\theta(s,a'')} \frac{\partial f_\theta(s,a'')}{\partial \theta}$
- $= \frac{\partial f_\theta(s,a)}{\partial \theta} - \mathbb{E}_{a' \sim \pi_\theta(a'|s)} \left[\frac{\partial f_\theta(s,a')}{\partial \theta} \right]$

backpropogate gradients

Recall: Monte-Carlo Estimate / Direct Evaluation

- Trajectories: $s_0^{(i)} \xrightarrow[R_1^{(i)}]{a_0^{(i)}} s_1^{(i)} \xrightarrow[R_2^{(i)}]{a_1^{(i)}} s_2^{(i)} \xrightarrow[R_3^{(i)}]{a_2^{(i)}} s_3^{(i)} \ldots s_T^{(i)} \sim \pi$
- Return: $G_t = R_{t+1} + \gamma R_{t+2} + \cdots \gamma^{T-1} R_T$

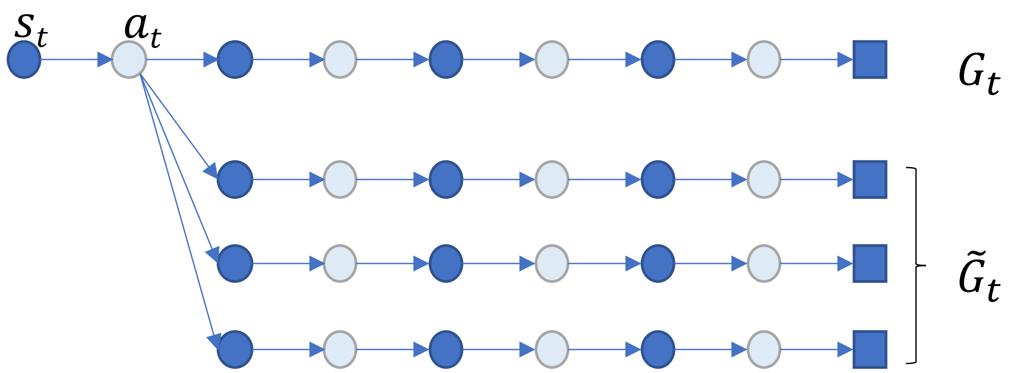
- $V^\pi(s) = \mathbb{E}[R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \cdots | s_0 = s, \pi]$
- $= \mathbb{E}[G_t | s_t = s, \pi]$
- $\simeq \frac{1}{N} \sum_{i=1}^{N} G_t^{(i)}$

REINFORCE

- Use sample discounted reward G_t as the unbiased estimation for $Q^{\pi_\theta}(s,a)$
- REINFORCE

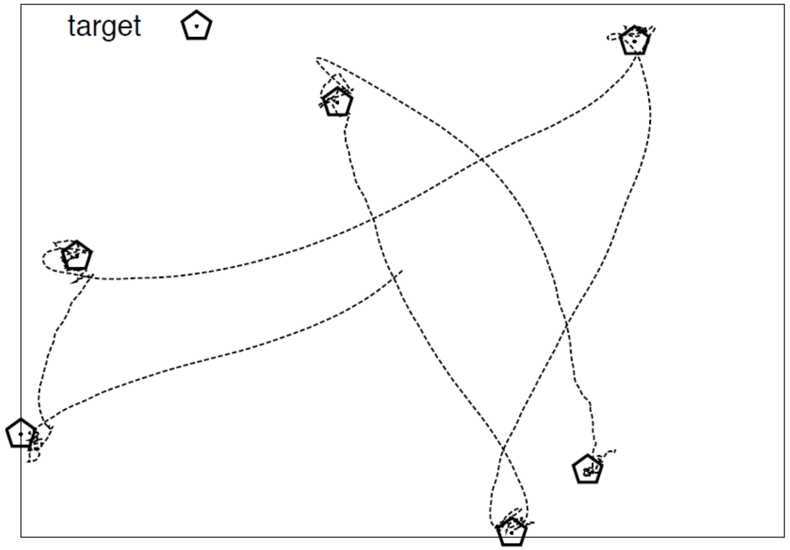
```
initialize θ arbitrarily
for each episode {s_1, a_1, r_2, …, s_{T−1}, a_{T−1}, r_T} ~ π_θ do
    for t = 1 to T − 1 do
        θ ← θ + α ∂/∂θ log π_θ(a_t|s_t) G_t
    end for
end for
return θ
```

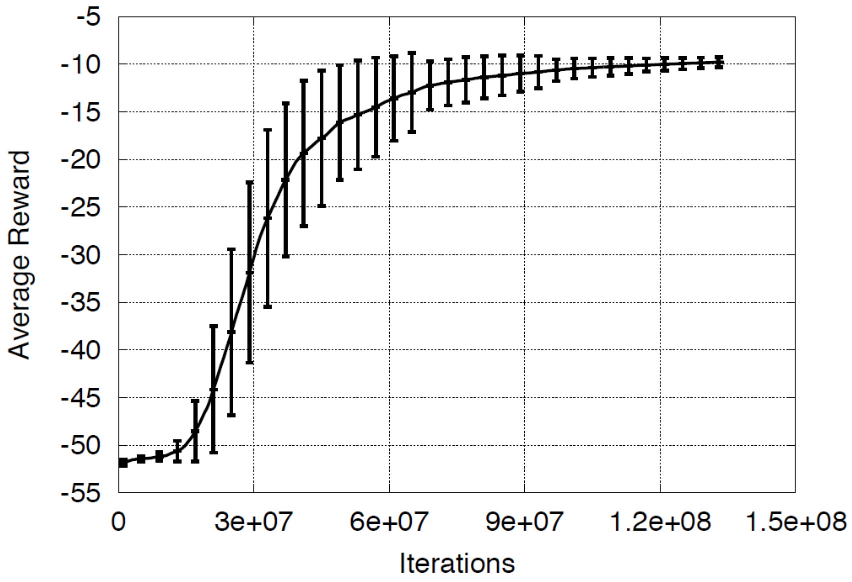
$$\theta \leftarrow \theta + \alpha \frac{\partial}{\partial \theta} \log \pi_\theta(a_t|s_t) G_t$$



could use multi-roll out $\tilde{G}_t = \frac{1}{N}\sum_{i=1}^{n} G_t^{(i)}$ to estimate $Q^{\pi_\theta}(s,a)$

Experimental results in Puck World 冰球世界





- Continuous actions on the puck ball
- Receive reward when near the target
- Target reset every 30s

REINFORCE

Actor-Critic

- Drawbacks of REINFORCE
  - Only have estimate G_t for a complete trajectory
  - Require large amount of data
  - Though unbiased, but high variance
- Actor-Critic: Train a critic Q_Φ to replace G_t

Actor $\pi_\theta(a|s)$

Adopt actions to satisfy the critic

Critic $Q_\Phi(s,a)$

Learn to evaluate well on actions

Actor-Critic: Training

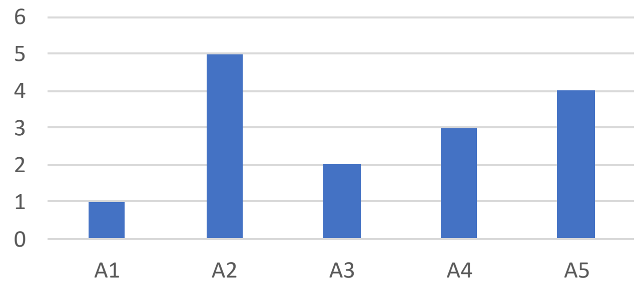
- Critic: $Q_\Phi(s, a)$
- $Q_\Phi(s, a) \simeq r(s, a) + \gamma \mathbb{E}_{s' \sim p(s'|s,a), a' \sim \pi_\theta(a'|s')}[Q_\Phi(s', a')]$

- Actor: $\pi_\theta(a|s)$
- $J(\theta) = \mathbb{E}_{s \sim p, \pi_\theta}[\pi_\theta(a|s) Q_\Phi(s, a)]$
- $\frac{\partial J(\theta)}{\partial \theta} = \mathbb{E}_{\pi_\theta}\left[\frac{\partial \log \pi_\theta(a|s)}{\partial \theta} Q_\Phi(s, a)\right]$

A2C: Advantageous Actor-Critic

- Further reduce variance
- Advantage: $A^{\pi}(s,a) = Q^{\pi}(s,a) - V^{\pi}(s)$

$Q^{\pi}(s,a)$



$A^{\pi}(s,a)$

- $\frac{\partial J(\theta)}{\partial \theta} \equiv \mathbb{E}_{\pi_\theta}\left[\frac{\partial \log \pi_\theta(a|s)}{\partial \theta}\left(Q_\Phi(s,a) - f(s)\right)\right]$
- $\frac{\partial J(\theta)}{\partial \theta} = \mathbb{E}_{\pi_\theta}\left[\frac{\partial \log \pi_\theta(a|s)}{\partial \theta} A_\Phi(s,a)\right]$

A2C: Advantageous Actor-Critic 2

- $Q^{\pi}(s,a) = r(s,a) + \gamma \mathbb{E}_{s' \sim p(s'|s,a), a' \sim \pi_\theta(a'|s')} [Q_\Phi(s',a')]$
- $= r(s,a) + \gamma \mathbb{E}_{s' \sim p(s'|s,a)}[V^{\pi}(s')]$

- $A^{\pi}(s,a) = Q^{\pi}(s,a) - V^{\pi}(s)$
- $= r(s,a) + \gamma \mathbb{E}_{s' \sim p(s'|s,a)}[V^{\pi}(s')] - V^{\pi}(s)$
- $\simeq r(s,a) + \gamma V^{\pi}(s') - V^{\pi}(s)$

sample next state s'

Learn $V_\Phi(s) \simeq V^{\pi_\theta}(s)$ is enough!

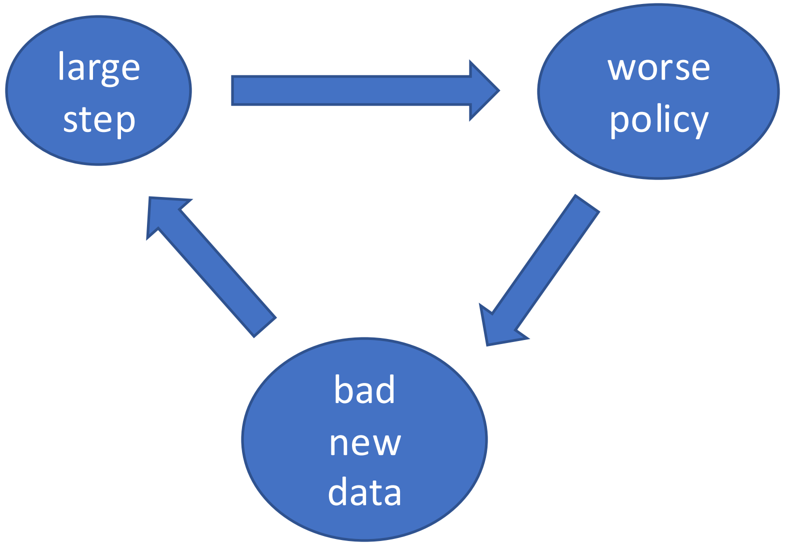
TRPO: Trust-Region Policy Optimization

- Limitations of REINFORCE

- Idea: Optimize in a trust region

- REINFORCE

initialize θ arbitrarily
for each episode $\{s_1, a_1, r_2, \ldots, s_{T-1}, a_{T-1}, r_T\} \sim \pi_\theta$ do
  for $t = 1$ to $T - 1$ do
    $\theta \leftarrow \theta + \alpha \frac{\partial}{\partial \theta} \log \pi_\theta(a_t | s_t) G_t$
  end for
end for
return θ



TRPO 2

$$J(\theta) = \mathbb{E}_{\tau\sim p_\theta(\tau)}[\textstyle\sum_t \gamma^t r(s_t, a_t)]$$
$$J(\theta) = \mathbb{E}_{s_0\sim p_\theta(s_0)}[V^{\pi_\theta}(s_0)]$$

- $$\begin{aligned} J(\theta') - J(\theta) &= J(\theta') - \mathbb{E}_{s_0\sim p(s_0)}[\mathrm{V}^{\pi_\theta}(s_0)] \\ &= J(\theta') - \mathbb{E}_{\tau\sim p_{\theta'}(\tau)}[\mathrm{V}^{\pi_\theta}(s_0)] \\ &= J(\theta') - \mathbb{E}_{\tau\sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty}\gamma^t \mathrm{V}^{\pi_\theta}(s_t) - \sum_{t=1}^{\infty}\gamma^t \mathrm{V}^{\pi_\theta}(s_t)\right] \\ &= J(\theta') + \mathbb{E}_{\tau\sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty}\gamma^t(\gamma \mathrm{V}^{\pi_\theta}(s_{t+1}) - \mathrm{V}^{\pi_\theta}(s_t))\right] \\ &= \mathbb{E}_{\tau\sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t, a_t)\right] + \mathbb{E}_{\tau\sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty}\gamma^t(\gamma \mathrm{V}^{\pi_\theta}(s_{t+1}) - \mathrm{V}^{\pi_\theta}(s_t))\right] \\ &= \mathbb{E}_{\tau\sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty}\gamma^t(r(s_t, a_t) + \gamma \mathrm{V}^{\pi_\theta}(s_{t+1}) - \mathrm{V}^{\pi_\theta}(s_t))\right] \\ &= \mathbb{E}_{\tau\sim p_{\theta'}(\tau)}[\sum_{t=0}^{\infty}\gamma^t A^{\pi_\theta}(s_t, a_t)] \end{aligned}$$

$$A^{\pi_\theta}(s_t, a_t) = Q^{\pi_\theta}(s_t, a_t) - \mathrm{V}^{\pi_\theta}(s_t)$$

TRPO 3

- $J(\theta') - J(\theta)$

$$= \mathbb{E}_{\tau \sim p_{\theta'}(\tau)}\left[\sum_{t=0}^{\infty} \gamma^t A^{\pi_\theta}(s_t, a_t)\right]$$

$$= \sum_t \mathbb{E}_{s_t \sim p_{\theta'}(s_t)}[\mathbb{E}_{a_t \sim \pi_{\theta'}(a_t|s_t)}[\gamma^t A^{\pi_\theta}(s_t, a_t)]]$$

$$= \sum_t \mathbb{E}_{s_t \sim p_{\theta'}(s_t)}[\mathbb{E}_{a_t \sim \pi_\theta(a_t|s_t)}[\frac{\pi_{\theta'}(a_t|s_t)}{\pi_\theta(a_t|s_t)} \gamma^t A^{\pi_\theta}(s_t, a_t)]]$$

$$J(\theta') - J(\theta) \approx \sum_t \mathbb{E}_{s_t \sim p_\theta(s_t)}[\mathbb{E}_{a_t \sim \pi_\theta(a_t|s_t)}[\frac{\pi_{\theta'}(a_t|s_t)}{\pi_\theta(a_t|s_t)} \gamma^t A^{\pi_\theta}(s_t, a_t)]]$$

TRPO 4

- $\theta' \leftarrow \arg\max_{\theta'} \sum_t \mathbb{E}_{s_t \sim p_\theta(s_t)}[\mathbb{E}_{a_t \sim \pi_\theta(a_t|s_t)}[\frac{\pi_{\theta'}(a_t|s_t)}{\pi_\theta(a_t|s_t)} \gamma^t A^{\pi_\theta}(s_t, a_t)]]$

$$s.t.\quad \mathbb{E}_{s_t \sim p(s_t)}[D_{KL}\left(\pi_{\theta'}(a_t|s_t) \parallel \pi_\theta(a_t|s_t)\right)] \leq \epsilon$$

- $\theta' \leftarrow \arg\max_{\theta'} \begin{array}{l} \sum_t \mathbb{E}_{s_t \sim p_\theta(s_t)}[\mathbb{E}_{a_t \sim \pi_\theta(a_t|s_t)}[\frac{\pi_{\theta'}(a_t|s_t)}{\pi_\theta(a_t|s_t)} \gamma^t A^{\pi_\theta}(s_t, a_t)]] \\ \qquad -\lambda(D_{KL}(\pi_{\theta'}(a_t|s_t) \parallel \pi_\theta(a_t|s_t)) - \epsilon) \end{array}$

TRPO 5: Natural Policy Gradient

schemes. The natural policy gradient (Kakade, 2002) can be obtained as a special case of the update in Equation (12) by using a linear approximation to L and a quadratic approximation to the $\overline{D}_{\mathrm{KL}}$ constraint, resulting in the following problem:

$$\underset{\theta}{\text{maximize}} \left[\nabla_\theta L_{\theta_{\text{old}}}(\theta)\big|_{\theta=\theta_{\text{old}}} \cdot (\theta - \theta_{\text{old}}) \right] \tag{17}$$

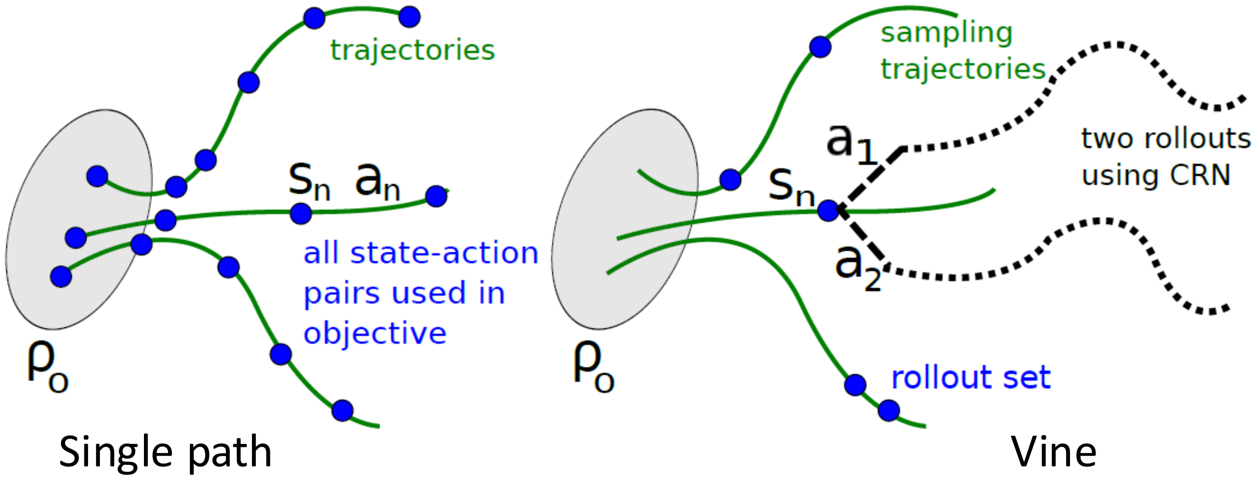
$$\text{subject to } \frac{1}{2}(\theta_{\text{old}} - \theta)^T A(\theta_{\text{old}})(\theta_{\text{old}} - \theta) \leq \delta,$$

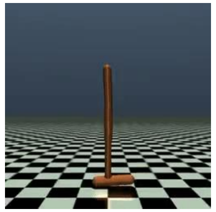
$$\text{where } A(\theta_{\text{old}})_{ij} =$$

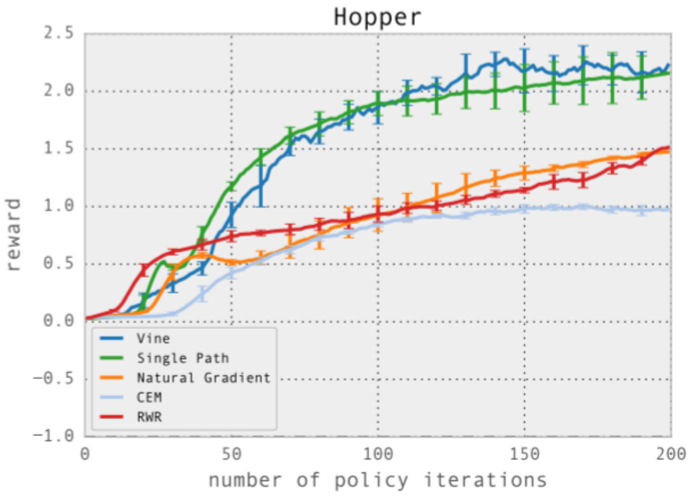
$$\frac{\partial}{\partial \theta_i} \frac{\partial}{\partial \theta_j} \mathbb{E}_{s\sim\rho_\pi} \left[D_{\mathrm{KL}}(\pi(\cdot|s, \theta_{\text{old}}) \,\|\, \pi(\cdot|s, \theta)) \right] \big|_{\theta=\theta_{\text{old}}}.$$

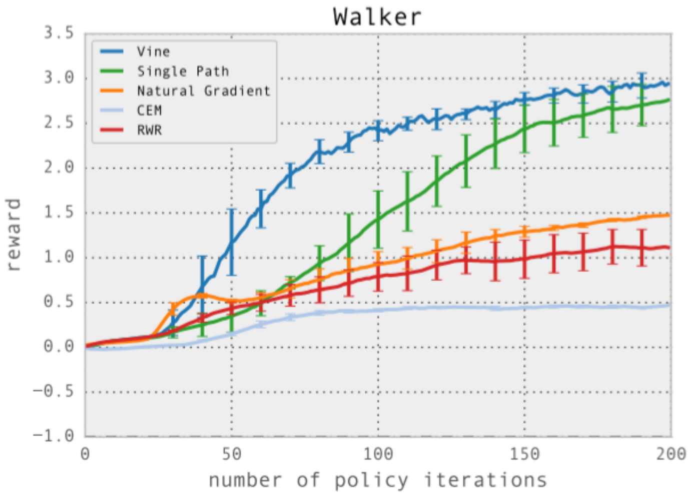
The update is $\theta_{\text{new}} = \theta_{\text{old}} + \frac{1}{\lambda} A(\theta_{\text{old}})^{-1} \nabla_\theta L(\theta)\big|_{\theta=\theta_{\text{old}}}$,

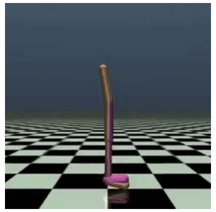
TRPO 6











“Trust Region Policy Optimization”, John Schulman, et al. (2017)

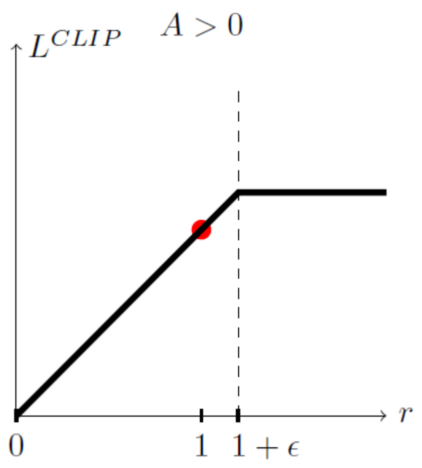
PPO: Proximal Policy Optimization

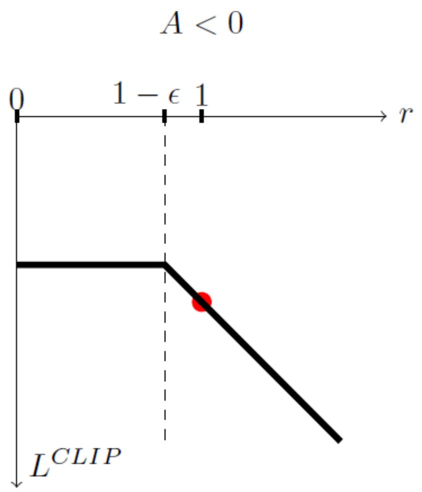
1. Cut-off the importance ratio

conservative policy iteration

$$L^{CPI}(\theta) = \hat{\mathbb{E}}_t\left[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}\hat{A}_t\right] = \hat{\mathbb{E}}_t\left[r_t(\theta)\hat{A}_t\right]$$

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





A lower bound

$L^{CLIP}(\theta) \leq L^{CPI}(\theta)$

near $r = 1$

$L^{CLIP}(\theta) = L^{CPI}(\theta)$

Proximal Policy Optimization Algorithms, John Schulman, et al. (2017)

PPO 2

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

- Use multi-step bootstrap to estimate advantage
- $\hat{A}_t = -V(s_t) + r_t + \gamma r_{t+1} + \cdots + \gamma^{T-t+1} r_{T-1} + \gamma^{T-t} V(s_T)$
- can be collected distributedly

- $L^{KLPEN}(\theta) = \hat{\mathbb{E}}_t\left[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}\hat{A}_t - \beta\ \text{KL}[\pi_{\theta_{\text{old}}}(\cdot\ |s_t)|\pi_\theta(\cdot\ |s_t)]\right]$
- If $KL < \frac{KL_{\text{targ}}}{1.5}, \beta \leftarrow \frac{\beta}{2}$
- If $KL > KL_{\text{targ}} \times 1.5, \beta \leftarrow \beta \times 2$

PPO 3

No clipping or penalty:	$L_t(\theta) = r_t(\theta)\hat{A}_t$
Clipping:	$L_t(\theta) = \min(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta)), 1-\epsilon, 1+\epsilon)\hat{A}_t$
KL penalty (fixed or adaptive)	$L_t(\theta) = r_t(\theta)\hat{A}_t - \beta \, \text{KL}[\pi_{\theta_{\text{old}}}, \pi_\theta]$

- 7 environments with continuous control
- 3 random seeds
- 100 episodes for each algorithm, average over 21 runs
- Normalize scores to 1

algorithm	avg. normalized score
No clipping or penalty	-0.39
Clipping, $\epsilon = 0.1$	0.76
Clipping, $\epsilon = 0.2$	**0.82**
Clipping, $\epsilon = 0.3$	0.70
Adaptive KL $d_{\text{targ}} = 0.003$	0.68
Adaptive KL $d_{\text{targ}} = 0.01$	0.74
Adaptive KL $d_{\text{targ}} = 0.03$	0.71
Fixed KL, $\beta = 0.3$	0.62
Fixed KL, $\beta = 1.$	0.71
Fixed KL, $\beta = 3.$	0.72
Fixed KL, $\beta = 10.$	0.69

Proximal Policy Optimization Algorithms, John Schulman, et al. (2017)

Summary

- Deep Q-Network
- Policy gradient
    - REINFORCE
    - Actor-Critic, A2C
    - TRPO, PPO

Shuai Li

https://shuaili8.github.io

Questions?

Supplementary: Policy gradient with averaged return

$$J(\pi) = \lim_{n\to\infty} \frac{1}{n} \mathbb{E}[r_1 + r_2 + \cdots + r_n | \pi] = \sum_s d^\pi(s) \sum_a \pi(a|s) r(s,a)$$

$$Q^\pi(s,a) = \sum_{t=1}^{\infty} \mathbb{E}[r_t - J(\pi) | s_0 = s, a_0 = a, \pi]$$

$$\begin{aligned}
\frac{\partial V^\pi(s)}{\partial \theta} &= \frac{\partial}{\partial \theta} \sum_a \pi(a|s) Q^\pi(s,a), \quad \forall s \\
&= \sum_a \left[\frac{\partial \pi(a|s)}{\partial \theta} Q^\pi(s,a) + \pi(a|s) \frac{\partial}{\partial \theta} Q^\pi(s,a) \right] \\
&= \sum_a \left[\frac{\partial \pi(a|s)}{\partial \theta} Q^\pi(s,a) + \pi(a|s) \frac{\partial}{\partial \theta} \left(r(s,a) - J(\pi) + \sum_{s'} P_{ss'}^a V^\pi(s') \right) \right] \\
&= \sum_a \left[\frac{\partial \pi(a|s)}{\partial \theta} Q^\pi(s,a) + \pi(a|s) \left(-\frac{\partial J(\pi)}{\partial \theta} + \frac{\partial}{\partial \theta} \sum_{s'} P_{ss'}^a V^\pi(s') \right) \right]
\end{aligned}$$

$$\Rightarrow \frac{\partial J(\pi)}{\partial \theta} = \sum_a \left[\frac{\partial \pi(a|s)}{\partial \theta} Q^\pi(s,a) + \pi(a|s) \sum_{s'} P_{ss'}^a \frac{\partial V^\pi(s')}{\partial \theta} \right] - \frac{\partial V^\pi(s)}{\partial \theta}$$

Supplementary: Policy gradient with averaged return 2

$$\frac{\partial J(\pi)}{\partial \theta} = \sum_a [\frac{\partial \pi(a|s)}{\partial \theta} Q^\pi(s,a) + \pi(a|s) \sum_{s'} P^a_{ss'} \frac{\partial V^\pi(s')}{\partial \theta}] - \frac{\partial V^\pi(s)}{\partial \theta}$$

$$\sum_s d^\pi(s) \frac{\partial J(\pi)}{\partial \theta} = \sum_s d^\pi(s) \sum_a \frac{\partial \pi(a|s)}{\partial \theta} Q^\pi(s,a) + \sum_s d^\pi(s) \sum_a \pi(a|s) \sum_{s'} P^a_{ss'} \frac{\partial V^\pi(s')}{\partial \theta} - \sum_s d^\pi(s) \frac{\partial V^\pi(s)}{\partial \theta}$$

$$\sum_s d^\pi(s) \sum_a \pi(a|s) \sum_{s'} P^a_{ss'} \frac{\partial V^\pi(s')}{\partial \theta} = \sum_s \sum_a \sum_{s'} d^\pi(s)\, \pi(a|s) P^a_{ss'} \frac{\partial V^\pi(s')}{\partial \theta}$$

$$= \sum_s \sum_{s'} d^\pi(s) \left(\sum_a \pi(a|s) P^a_{ss'} \right) \frac{\partial V^\pi(s')}{\partial \theta} = \sum_s \sum_{s'} d^\pi(s)\, P_{ss'} \frac{\partial V^\pi(s')}{\partial \theta}$$

$$= \sum_{s'} \left(\sum_s d^\pi(s) P_{ss'} \right) \frac{\partial V^\pi(s')}{\partial \theta} = \sum_{s'} d^\pi(s') \frac{\partial V^\pi(s')}{\partial \theta}$$

$$\Rightarrow \sum_s d^\pi(s) \frac{\partial J(\pi)}{\partial \theta} = \sum_s d^\pi(s) \sum_a \frac{\partial \pi(a|s)}{\partial \theta} Q^\pi(s,a) + \sum_{s'} d^\pi(s') \frac{\partial V^\pi(s')}{\partial \theta} - \sum_s d^\pi(s) \frac{\partial V^\pi(s)}{\partial \theta}$$

$$\Rightarrow \frac{\partial J(\pi)}{\partial \theta} = \sum_s d^\pi(s) \sum_a \frac{\partial \pi(a|s)}{\partial \theta} Q^\pi(s,a)$$

Supplementary: Policy gradient w/ discounted reward

$$J(\pi) = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t \middle| s_0, \pi\right]$$

$$Q^{\pi}(s,a) = \mathbb{E}\left[\sum_{k=1}^{\infty} \gamma^{k-1} r_{t+k} \middle| s_t = s, a_t = a, \pi\right]$$

$$\begin{aligned}
\frac{\partial V^{\pi}(s)}{\partial \theta} &= \frac{\partial}{\partial \theta} \sum_a \pi(s,a) Q^{\pi}(s,a), \quad \forall s \\
&= \sum_a \left[\frac{\partial \pi(s,a)}{\partial \theta} Q^{\pi}(s,a) + \pi(s,a) \frac{\partial}{\partial \theta} Q^{\pi}(s,a)\right] \\
&= \sum_a \left[\frac{\partial \pi(s,a)}{\partial \theta} Q^{\pi}(s,a) + \pi(s,a) \frac{\partial}{\partial \theta}\left(r(s,a) + \sum_{s'} \gamma P_{ss'}^{a} V^{\pi}(s')\right)\right] \\
&= \sum_a \frac{\partial \pi(s,a)}{\partial \theta} Q^{\pi}(s,a) + \sum_a \pi(s,a) \gamma \sum_{s'} P_{ss'}^{a} \frac{\partial V^{\pi}(s')}{\partial \theta}
\end{aligned}$$

Supplementary: Policy gradient w/ discounted reward 2

$$\frac{\partial V^\pi(s)}{\partial\theta} = \sum_a \frac{\partial\pi(s,a)}{\partial\theta} Q^\pi(s,a) + \sum_a \pi(s,a)\gamma \sum_{s_1} P_{ss_1}^a \frac{\partial V^\pi(s_1)}{\partial\theta}$$

$$\sum_a \frac{\partial\pi(s,a)}{\partial\theta} Q^\pi(s,a) = \gamma^0 \Pr(s \to s, 0, \pi) \sum_a \frac{\partial\pi(s,a)}{\partial\theta} Q^\pi(s,a)$$

$$\sum_a \pi(s,a)\gamma \sum_{s_1} P_{ss_1}^a \frac{\partial V^\pi(s_1)}{\partial\theta} = \sum_{s_1} \sum_a \pi(s,a)\gamma\, P_{ss_1}^a \frac{\partial V^\pi(s_1)}{\partial\theta}$$

$$= \sum_{s_1} \gamma P_{ss_1} \frac{\partial V^\pi(s_1)}{\partial\theta} = \gamma^1 \sum_{s_1} \Pr(s \to s_1, 1, \pi) \frac{\partial V^\pi(s_1)}{\partial\theta}$$

$$\frac{\partial V^\pi(s_1)}{\partial\theta} = \sum_a \frac{\partial\pi(s_1,a)}{\partial\theta} Q^\pi(s_1,a) + \gamma^1 \sum_{s_2} \Pr(s_1 \to s_2, 1, \pi) \frac{\partial V^\pi(s_2)}{\partial\theta}$$

Supplementary: Policy gradient w/ discounted reward 3

$$\frac{\partial V^\pi(s)}{\partial\theta} = \gamma^0 \Pr(s \to s, 0, \pi) \sum_a \frac{\partial \pi(s,a)}{\partial\theta} Q^\pi(s,a) + \gamma^1 \sum_{s_1} \Pr(s \to s_1, 1, \pi) \sum_a \frac{\partial \pi(s_1,a)}{\partial\theta} Q^\pi(s_1,a)$$

$$+\gamma^2 \sum_{s_1} \Pr(s \to s_1, 1, \pi) \sum_{s_2} \Pr(s_1 \to s_2, 1, \pi) \frac{\partial V^\pi(s_2)}{\partial\theta}$$

$$= \gamma^0 \Pr(s \to s, 0, \pi) \sum_a \frac{\partial \pi(s,a)}{\partial\theta} Q^\pi(s,a) + \gamma^1 \sum_{s_1} \Pr(s \to s_1, 1, \pi) \sum_a \frac{\partial \pi(s_1,a)}{\partial\theta} Q^\pi(s_1,a)$$

$$+\gamma^2 \sum_{s_2} \Pr(s \to s_2, 2, \pi) \frac{\partial V^\pi(s_2)}{\partial\theta}$$

$$= \sum_{k=0}^{\infty} \sum_x \gamma^k \Pr(s \to x, k, \pi) \sum_a \frac{\partial \pi(x,a)}{\partial\theta} Q^\pi(x,a) = \sum_x \sum_{k=0}^{\infty} \gamma^k \Pr(s \to x, k, \pi) \sum_a \frac{\partial \pi(x,a)}{\partial\theta} Q^\pi(x,a)$$

$$\Rightarrow \frac{\partial J(\pi)}{\partial\theta} = \frac{\partial V^\pi(s_0)}{\partial\theta} = \sum_s \sum_{k=0}^{\infty} \gamma^k \Pr(s_0 \to s, k, \pi) \sum_a \frac{\partial \pi(s,a)}{\partial\theta} Q^\pi(s,a\,) = \sum_s d^\pi(s) \sum_a \frac{\partial \pi(s,a)}{\partial\theta} Q^\pi(s,a)$$