Reinforcement Learning

—

Exercise 8: Actor-Critic Methods

Nico Meyer

Fraunhofer IIS

Fraunhofer-Institut für Integrierte Schaltungen IIS

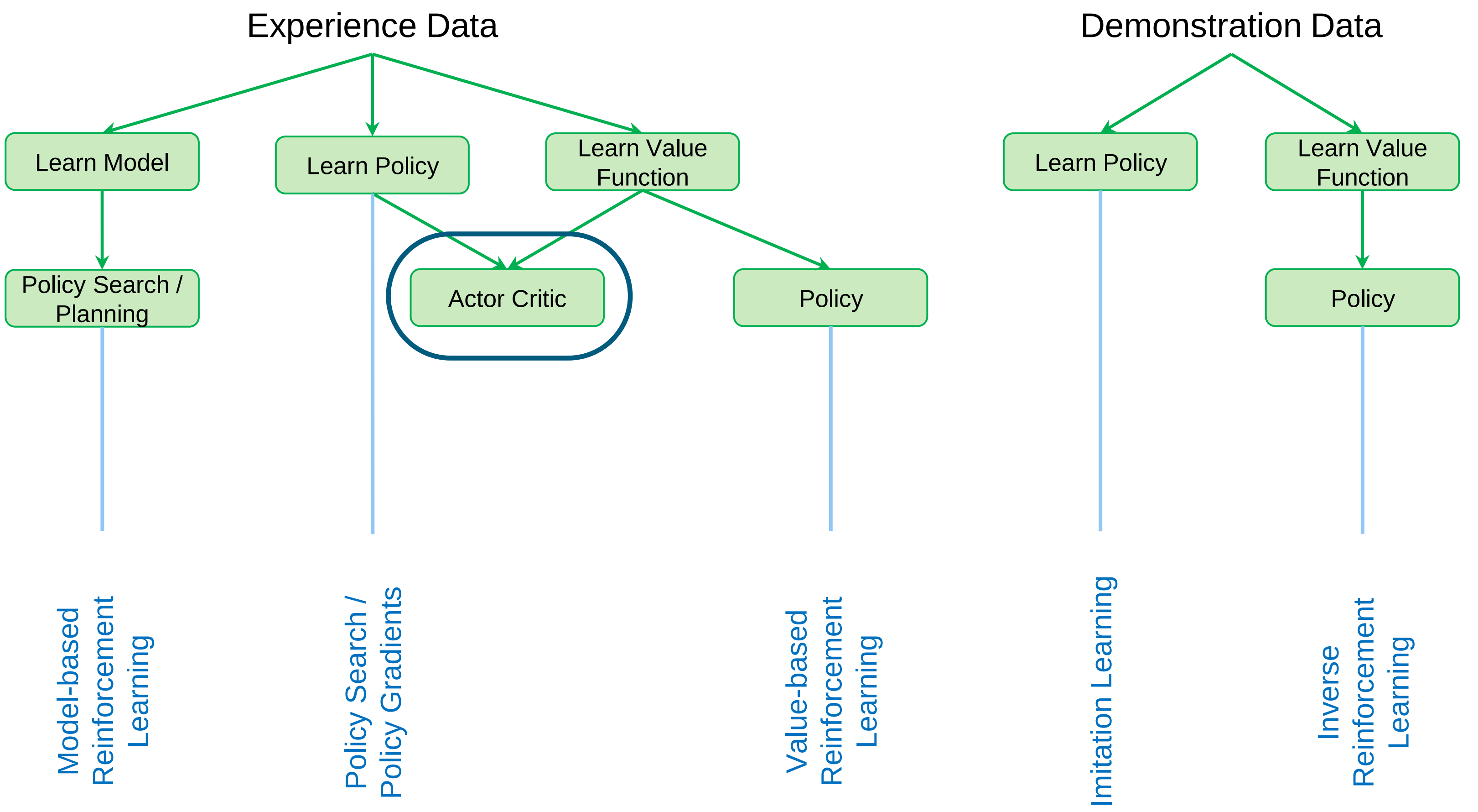
Overview

Exercise Content

Week	*Date*	*Topic*	*Material*	*Who?*
0			*no exercises*	
1	23.04.	MDPs		Nico
2	30.04.	Dynamic Programming		Alex
3	07.05.	OpenAI Gym, PyTorch-Intro		Alex
4	14.05.	TD-Learning		Nico
5	22.05.	Practical Session (zoom@home)	**Attention: Lecture Slot!**	Nico + Alex
6	28.05.	TD-Control		Nico
7	04.06.	DQN		Nico
8	11.06.	VPG		Alex
9	18.06.	A2C		Nico
10	25.06.	Multi-armed Bandits		Alex
11	02.07.	RND/ICM		Alex
12	09.07.	MCTS		Alex
13	16.07.	BCQ		Nico

Overview

General Picture



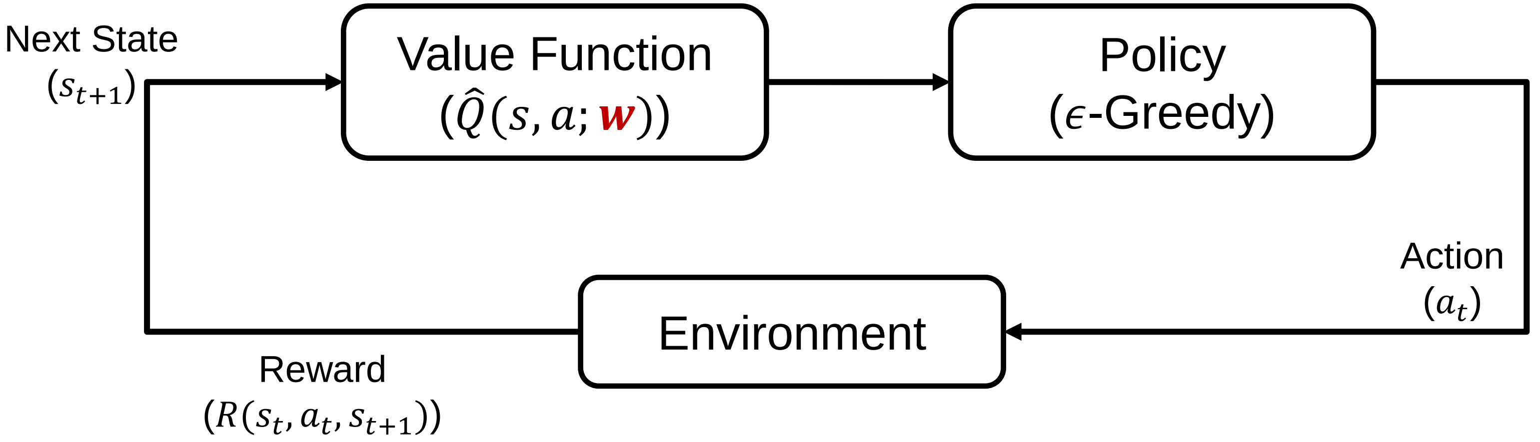
Actor-Critic Methods

Advantage Actor Critic

Brief Recap

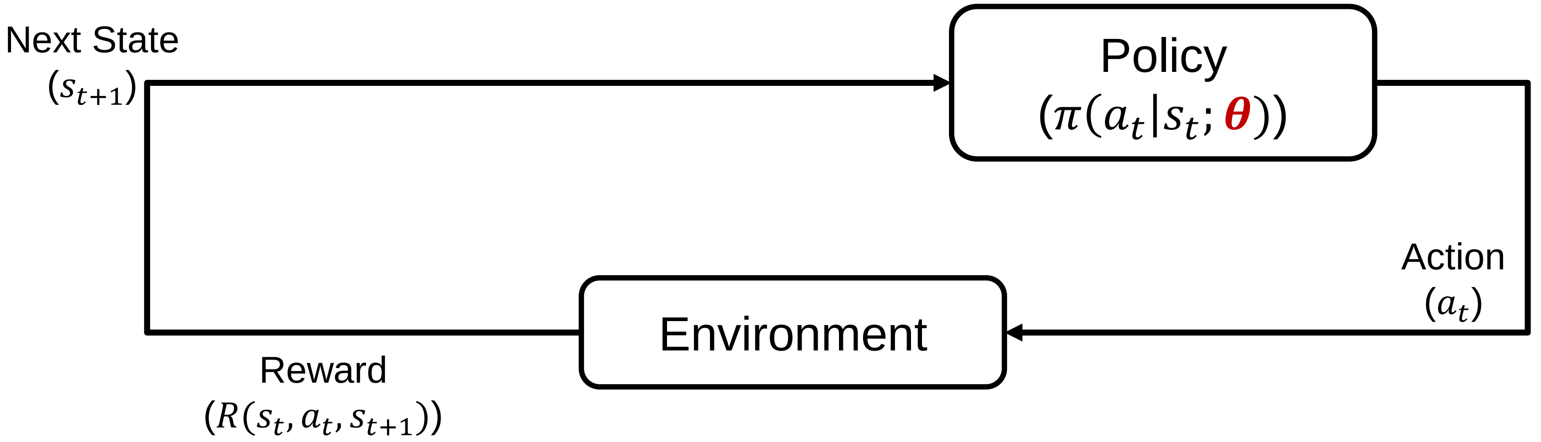
Policy-based Reinforcement Learning

Value-based:



Goal: find w that approximates the true Q-function

Policy-based:



Goal: find θ that maximizes long term reward

Fraunhofer IIS

Recap

Policy Gradients

- Our goal is to maximize the expected reward:

$$G(\tau) \coloneqq \sum_{t=0}^{T-1} \gamma^t R(s_t, a_t)$$

$$\max_\theta \mathbb{E}_{\pi_\theta} G(\tau)$$

(where π_θ is a parameterized policy, e.g., a neural network)

- But how do we maximize this?

→ Gradient Ascent! *Suppose* we know how to calculate the gradient w.r.t. the parameters:

$$\nabla_\theta \mathbb{E}_{\tau \sim \pi_\theta} G(\tau)$$

Policy Gradient

often in literature referred to as $\nabla_\theta J(\pi_\theta)$

- Then we can update our parameters θ in the direction of the gradient:

$$\theta \leftarrow \theta + \alpha \nabla_\theta \mathbb{E}_{\tau \sim \pi_\theta} G(\tau)$$

Fraunhofer IIS

Actor-Critic Methods

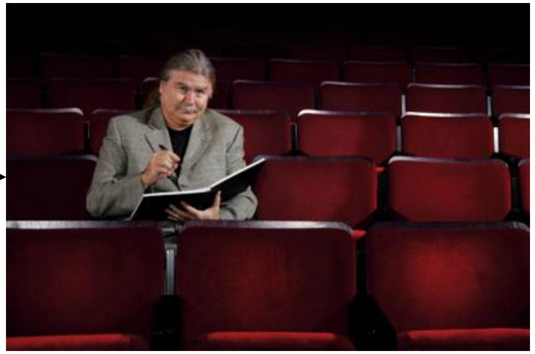
Reducing Variance

$$\nabla_\theta \mathbb{E}_{\pi_\theta} G(\tau) \approx \frac{1}{L} \sum_{\tau} \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \underbrace{\sum_{t'=t}^{T} \gamma^{t'-t} R(s_{t'}, a_{t'})}_{= \, Q^\pi(s_t, a_t)} - b(s_t)$$

- Monte-Carlo policy gradient is sampled and has high variance
- Idea: we can use a critic that estimates the Q

actor

critic



Fraunhofer IIS

Actor-Critic Methods

Introduce critic that estimates Q

- The policy gradient we used so far (without baseline to begin with):

$$\nabla_\theta \mathbb{E}_{\pi_\theta} G(\tau) \approx \frac{1}{L}\sum_{\tau}\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t)\, G(\tau)$$

$$\approx \frac{1}{L}\sum_{\tau}\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \underbrace{\sum_{t'=t}^{T} \gamma^{t'-t} R(s_{t'}, a_{t'})}_{= \; Q^\pi(s_t, a_t)}$$

➢ Use e.g. a neural network to approximate Q: $\phi_k = \arg\min_\phi \mathbb{E}_{s_t; a_t, \hat{R}_t \sim \pi_k} \left[\left(Q_\phi(s_t, a_t) - \hat{R}_t \right)^2 \right]$

- In practice: estimate $v^\pi(s_t; \phi)$ explicitly, and then sample

$$q^\pi(s_t, a_t) \approx G_t^{(n)} \qquad \text{i.e. } \hat{G}_t^{(1)} = R_t + \gamma v^\pi(s_{t+1}; \phi)$$

Actor-Critic Methods

Advantage Actor Critic (A2C)

- Introduce a baseline:

$$\nabla_\theta \mathbb{E}_{\pi_\theta} G(\tau) \approx \frac{1}{L}\sum_{\tau} \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \overbrace{\sum_{t'=t}^{T} \gamma^{t'-t} R(s_{t'}, a_{t'})}^{=\ Q^\pi(s_t, a_t)} - b(s_t)$$

$$= \frac{1}{L}\sum_{\tau} \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \underbrace{\boxed{(\hat{G}_t - b(s_t))}}_{:=\ A^\pi(s_t, a_t)}$$

➢ Calculate via MC estimation:

$$A^\pi(s_t, a_t) = R(s_t, a_t) - V^\pi(s_t)$$

Actor-Critic Methods

Advantage Actor Critic (A2C)

- Calculate via TD error:

$$\begin{aligned} A^{\pi}(s_t, a_t) &= Q^{\pi}(s_t, a_t) - V^{\pi}(s_t) \\ &= r + \gamma \cdot v^{\pi}({s'}_t) - v^{\pi}(s_t) \end{aligned}$$

- Or multi-step TD error: "Generalized Advantage Estimation (GAE)"

$$\begin{aligned} \hat{A}_t^{(1)} &:= \delta_t^V && = -V(s_t) + r_t + \gamma V(s_{t+1}) \\ \hat{A}_t^{(2)} &:= \delta_t^V + \gamma \delta_{t+1}^V && = -V(s_t) + r_t + \gamma V(s_{t+1}) + \gamma^2 V(s_{t+2}) \\ & && \dots \\ \hat{A}_t^{(k)} &:= \sum_{l=0}^{k-1} \gamma^l \delta_{t+l}^V && = -V(s_t) + r_t + \gamma r_{t+1} + \cdots + \gamma^{k-1} r_{t+k-1} + \gamma^k V(s_{t+k}) \end{aligned}$$

$$\hat{A}_t^{(\infty)} = \sum_{l=0}^{\infty} \gamma^l \delta_{t+l}^V = -V(s_t) + \sum_{l=0}^{\infty} \gamma^l + r_{t+l}$$

➢ Less variance, more bias than MC

Exercise Sheet 8

Advantage Actor Critic (A2C)

Better Control of Improvement Steps

Potential problems with gradient-based updates

- Note: the advantage function (which is a noisy estimate) may not be accurate
  - Too large steps may lead to a disaster (even *if* the gradient is *correct*)
  - Too small steps are also bad
- Mathematical formulization:
  - First-order derivatives approximate the (parameter) surface to be flat
  - But if the surface exhibits high curvature it gets dangerous
  - Projection: small changes in parameter space might lead to large changes in policy space!
- **Parameters θ get updated to areas too far out of the range from where previous data was collected**

➢ Regularize updates to the policy parameters such that the policy does not change too much

Images taken from https://medium.com/@jonathan_hui/rl-trust-region-policy-optimization-trpo-explained-a6ee04eeeee9 and http://www.taiwanoffthebeatentrack.com/2012/08/23/mount-hua-华山-the-most-dangerous-hike-in-the-world/

Fraunhofer IIS

Better Control of Improvement Steps

Natural Policy Gradients

TRPO tries to approximated inverse of Fisher information (i.e. Hessian)

- First-order derivatives approximate the (parameter) surface to be flat
- But if the surface exhibits high curvature it gets dangerous

➢ Small changes in parameter space might lead to large changes in policy space!

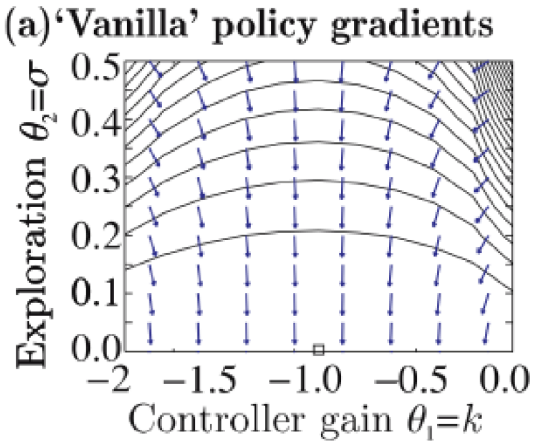
What we essentially do
(optimization perspective on 1st order gradient descent)

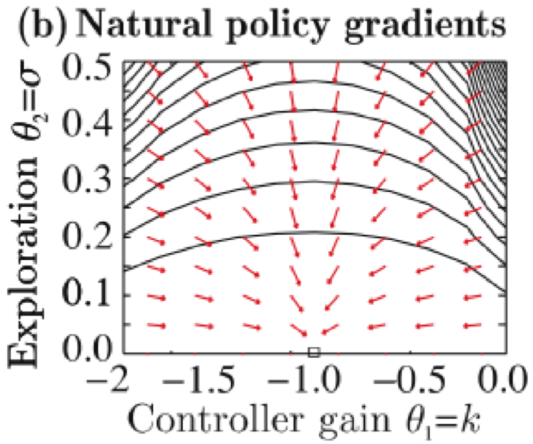
$$\theta' \leftarrow \arg\max_{\theta'} (\theta' - \theta)^T \nabla_\theta J(\theta), \text{ subject to } \|\theta' - \theta\|^2 \leq \epsilon$$

What we want to do
(incorporate 2nd order information)

$$\theta' \leftarrow \arg\max_{\theta'} (\theta' - \theta)^T \nabla_\theta J(\theta), \text{ subject to } \|\theta' - \theta\|_F^2 \leq \epsilon$$

→ $\theta \leftarrow \theta + \alpha \mathbf{F}^{-1} \nabla_\theta J(\theta)$ with e.g. KL-divergence





Peters et al.: Natural Actor-Critic. 2018

Fraunhofer IIS

Better Control of Improvement Steps

Proximal Policy Optimization (PPO)

- The main motivation behind PPO is the same as for TRPO:
  - Make the biggest possible improvement step
  - Do not step too far such that the performance accidentally collapses

- PPO addresses the shortcomings of TRPO:
  - PPO uses 1st order methods with a few tricks
  - Significantly simpler to implement
  - Shows similar performance to TRPO (empirically)

- <u>PPO-Clip:</u> The PPO objective we want to maximize is given by

removes the incentive for moving r_t outside of the interval $[1-\epsilon, 1+\epsilon]$

$= g\left(\epsilon, \hat{A}_t(s,a)\right)$ $\quad g(\epsilon, \hat{A}_t) = \begin{cases}(1+\epsilon)A, \text{if } A \geq 0 \\ (1-\epsilon)A, \text{if } A < 0\end{cases}$

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

where ϵ is a hyperparameter (i.e., 0.1 or 0.2) that defines how far π_{new} may go away from π_{old}

➢ the final objective is a lower bound (i.e., a pessimistic bound) on the unclipped objective

Fraunhofer IIS

Fraunhofer-Institut für Integrierte Schaltungen IIS

Thank you for your attention!