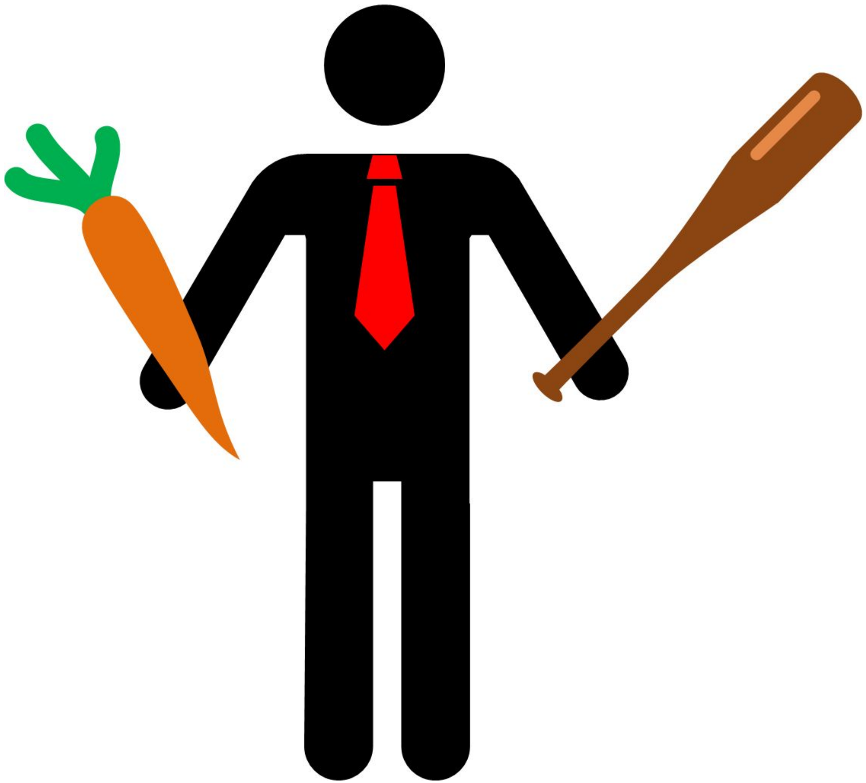
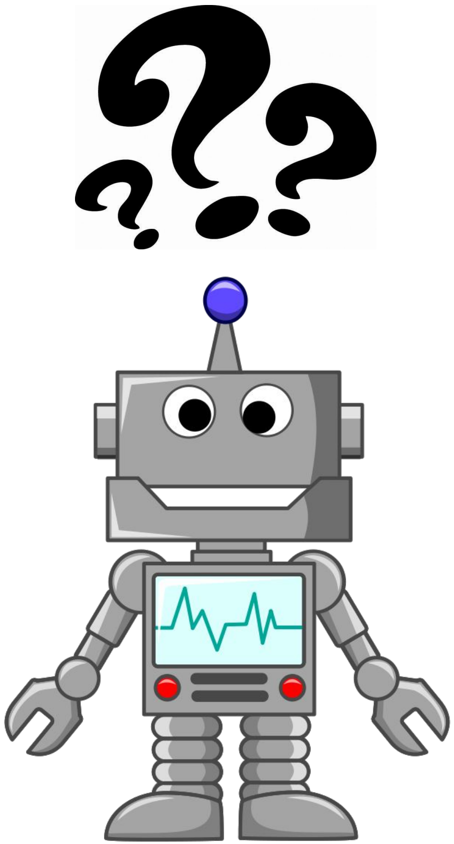
Seminar in Deep Reinforcement Learning

Introduction

Reinforcement Learning...





Disclaimer: This is a seminar...

(almost) no basics

participation required

Format

- Assigned papers
- 35 min presentation
- 10 min facilitated discussion
- Voluntary coding challenge

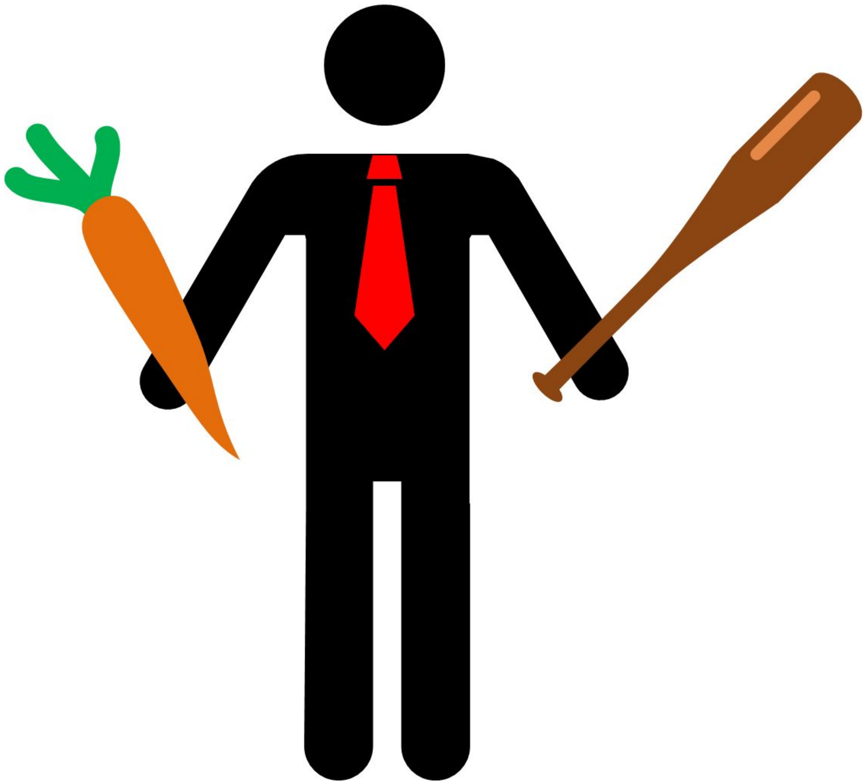
Grade =

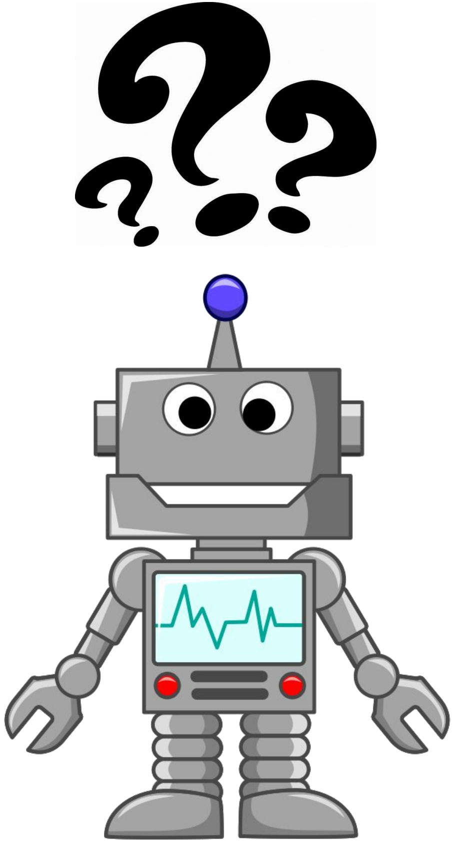
presentation + active participation (+ challenge)

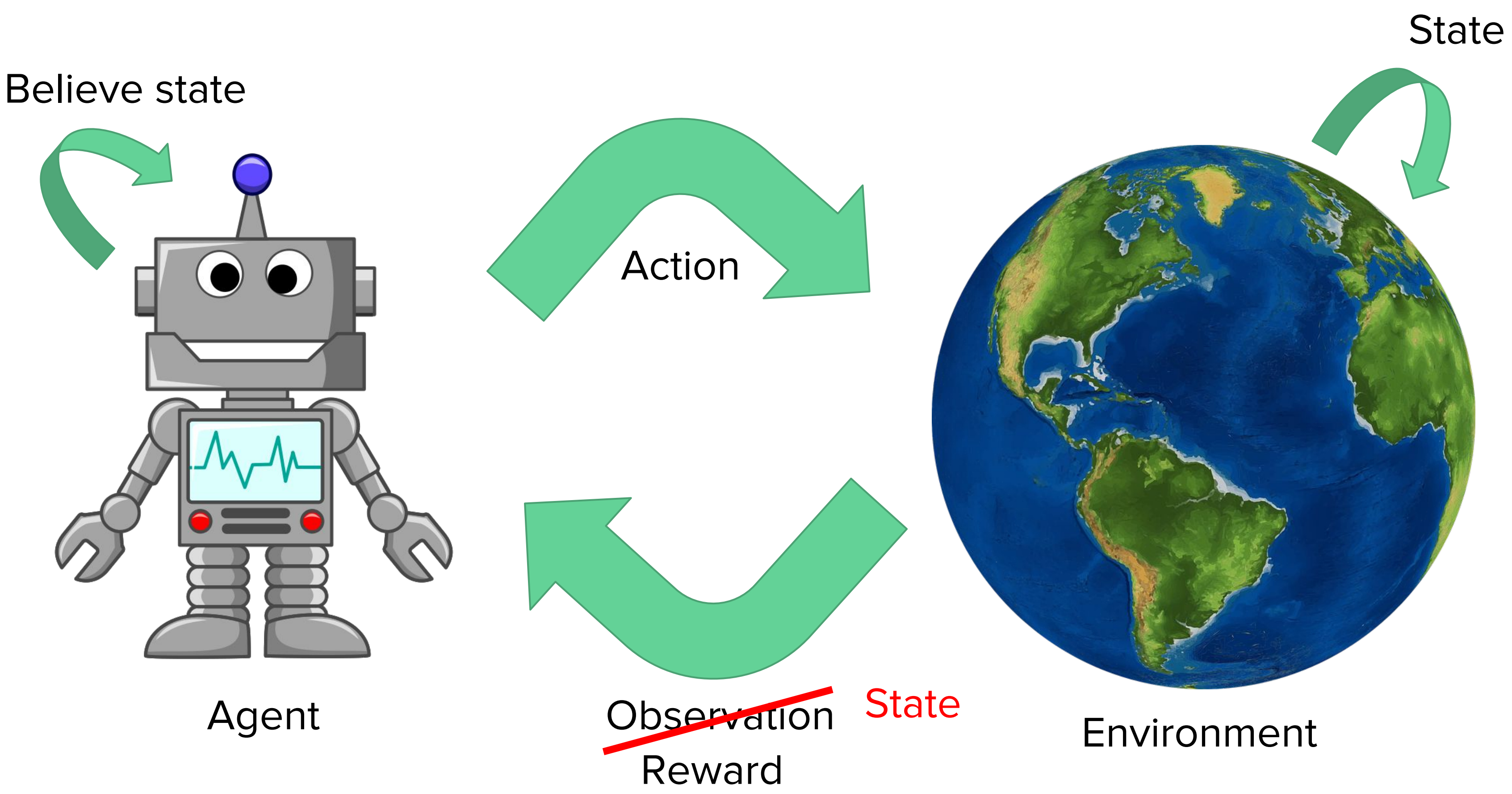
19.02.2019	Introduction
26.02.2019	Distributional Deep Reinforcement Learning
05.03.2019	Continuous Control
12.03.2019	Variance Reduction
19.03.2019	Overestimation in Q-Learning / Distributed Deep Reinforcement Learning
26.03.2019	Learning from Artificial Demonstrations
02.04.2019	Exploration-Exploitation Trade-off in Deep Reinforcement Learning

09.04.2019	Off-Policy Learning	
16.04.2019	Hierarchical Deep Reinforcement Learning	
30.04.2019	Multi-Agent Deep Reinforcement Learning	
07.05.2019	Multitask/Transfer Deep Reinforcement Learning	
14.05.2019	Model Based Deep Reinforcement Learning	
21.05.2019	Meta Learning / Human Influence	Coding Challenge Hand-In
28.05.2019	Discussion & Coding Challenge	

Reinforcement Learning...






State
Believe state
Action
Agent
~~Observation~~
Reward
State
Environment

Task?

Maximize the discounted cumulative reward in each episode by finding a good policy

Task?

Maximize the discounted cumulative reward in each episode by finding a good policy

Task?

Maximize the **discounted cumulative reward** in each episode by finding a good policy

$$R = \sum_t \gamma^t r_t$$

$$\gamma \in (0, 1]$$

Task?

Maximize the discounted cumulative reward in each **episode** by finding a good policy

$$R = \sum_{t=0}^{T_e} \gamma^t r_t$$

$$\gamma \in (0, 1]$$

Task?

Maximize the discounted cumulative reward in each episode by finding a good **policy**

$$R^{\pi} = \sum_{t=0}^{T_e} \gamma^t r_t$$

$$\pi(a|s_t) = Pr(a|s_t)$$

How?

Estimate remainder of R^{π} in each state s_t

$$V^{\pi}(\boldsymbol{s}_t) = \mathbb{E}_{\pi}[\sum_{t'=t}^{T_e} \gamma^{t'-t} \boldsymbol{r}_{t'}]$$

$$Q^{\pi}(\boldsymbol{s}_t, \boldsymbol{a}_t) = \boldsymbol{r}_t | \boldsymbol{a}_t + \mathbb{E}_{\pi|a_t}[\sum_{t'=t+1}^{T_e} \gamma^{t'-t} \boldsymbol{r}_{t'}]$$

Policy?

$$\pi^{greedy}(a|\boldsymbol{s}_t) = \mathbf{1}_{a=\max_{a'} Q^*(\boldsymbol{s}_t, a')}$$

$$
\begin{aligned}
V^{\pi}(\boldsymbol{s}_t) &= \mathbb{E}_{\pi}\left[\sum_{t'=t}^{T_e} \gamma^{t'-t} \boldsymbol{r}_{t'}\right] \\
&= \mathbb{E}_{\pi}[\boldsymbol{r}_t] + \gamma \mathbb{E}_{\pi}\left[\sum_{t'=t+1}^{T_e} \gamma^{t'-t-1} \boldsymbol{r}_{t'}\right] \\
&= \mathbb{E}_{\pi}[\boldsymbol{r}_t] + \gamma V^{\pi}(\boldsymbol{s}_{t+1})
\end{aligned}
$$

$$Q^{\pi}(\boldsymbol{s}_t, a_t) = \boldsymbol{r}_t | a_t + \mathbb{E}_{\pi | a_t}\left[\sum_{t'=t+1}^{T_e} \gamma^{t'-t} \boldsymbol{r}_{t'}\right]$$

$$= \boldsymbol{r}_t | a_t + \gamma V^{\pi}(\boldsymbol{s}_{t+1})$$

$$V^{greedy}(\boldsymbol{s}_t) = \max_{a'} Q^{greedy}(\boldsymbol{s}_t, a')$$

Q-Learning

Watkins (1989)

$$Q^{greedy}(s_t, a_t) = r_t|a_t + \gamma \max_{a'} Q^{greedy}(s_{t+1}, a')$$

$$\text{iff } Q^{greedy} \equiv Q^*$$

$$y(s_t, a_t) := r_t|a_t + \gamma \max_{a'} \tilde{Q}(s_{t+1}, a')$$

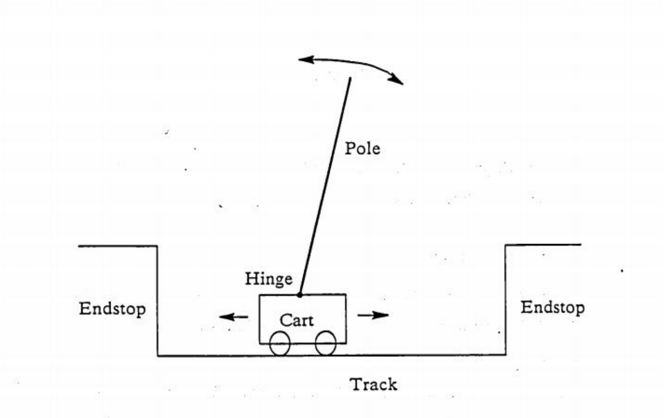
$$\delta_{TD} = y(s_t, a_t) - \tilde{Q}(s_t, a_t)$$

$$\rightarrow \text{minimize } \delta_{TD}^2$$

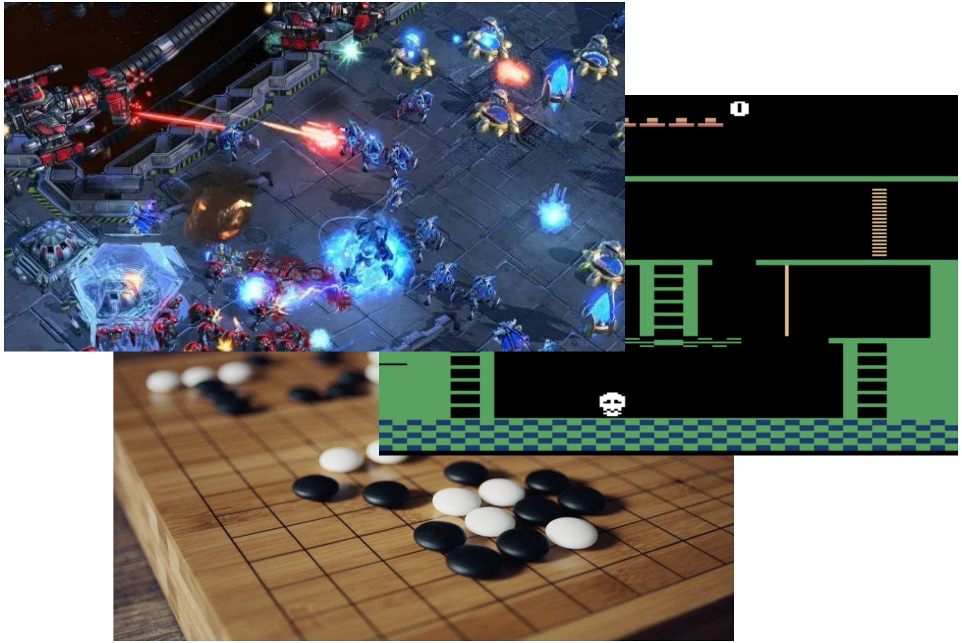
Classical RL vs Deep RL

estimate $\tilde{Q}(s, a)$

$\forall (s, a) \in \mathcal{S} \times \mathcal{A}$

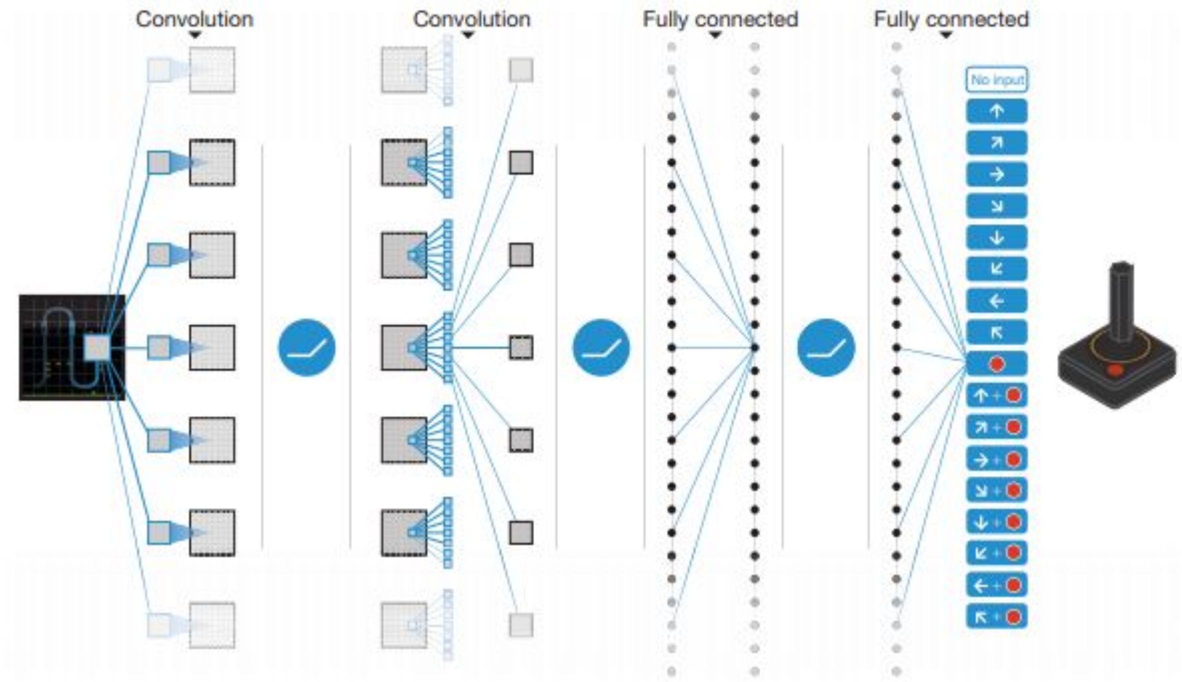


vs.



$\rightarrow$ approximate $\tilde{Q}(\cdot, \cdot)$

Human-level control through DRL (Mnih et al., 2015)



DQN

Deep Learning Works for... RL?

- ...~~large data~~ sets...
  many interactions
- ...with ~~labeled~~ data points... ➜ target network
  self-labeled
- ...which are iid ➜ replay buffer

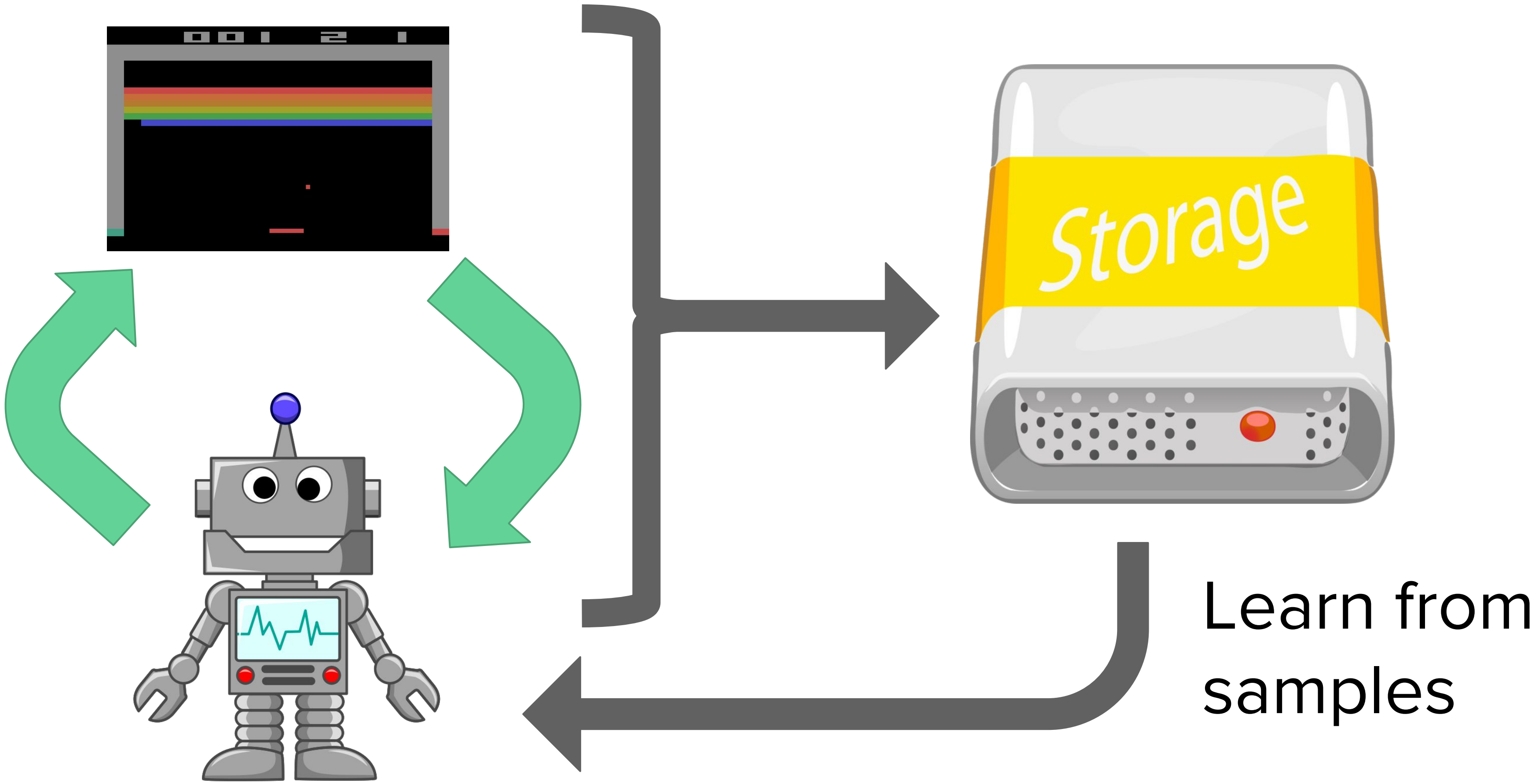
Target Network

$$y(s_t, a_t) := r_t | a_t + \gamma \max_{a'} \tilde{Q}_{\theta^-}(s_{t+1}, a')$$

$$\delta_{TD} = y(s_t, a_t) - \tilde{Q}_\theta(s_t, a_t)$$

$$\rightarrow \text{minimize } \delta_{TD}^2$$

Replay Buffer



Task?

Maximize the discounted cumulative reward in each episode by finding a good policy

$$R^{\pi} = \mathbb{E}_{\pi}\left[\sum_{t=0}^{T_e} \gamma^t r_t\right]$$

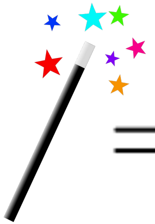
$$\pi_{\theta}(a|s_t) = Pr(a|s_t;\theta)$$

$$\max_\theta R^{\pi_\theta}$$

$$\rightarrow \theta_{k+1} = \theta_k + \alpha \nabla_\theta R^{\pi_\theta}$$

$$\nabla_\theta R^{\pi_\theta} = ?$$

Markov



$$= \mathbb{E}_\pi\left[\left(\sum_{t=0}^{T_e} \gamma^t r_t\right)\left(\sum_{t=0}^{T_e} \nabla \log \pi(a_t|s_t)\right)\right]$$

$$\mathbb{E}_\pi[(\sum_{t=0}^{T_e} \gamma^t r_t)(\sum_{t=0}^{T_e} \nabla \log \pi(a_t|s_t))]$$

Causality

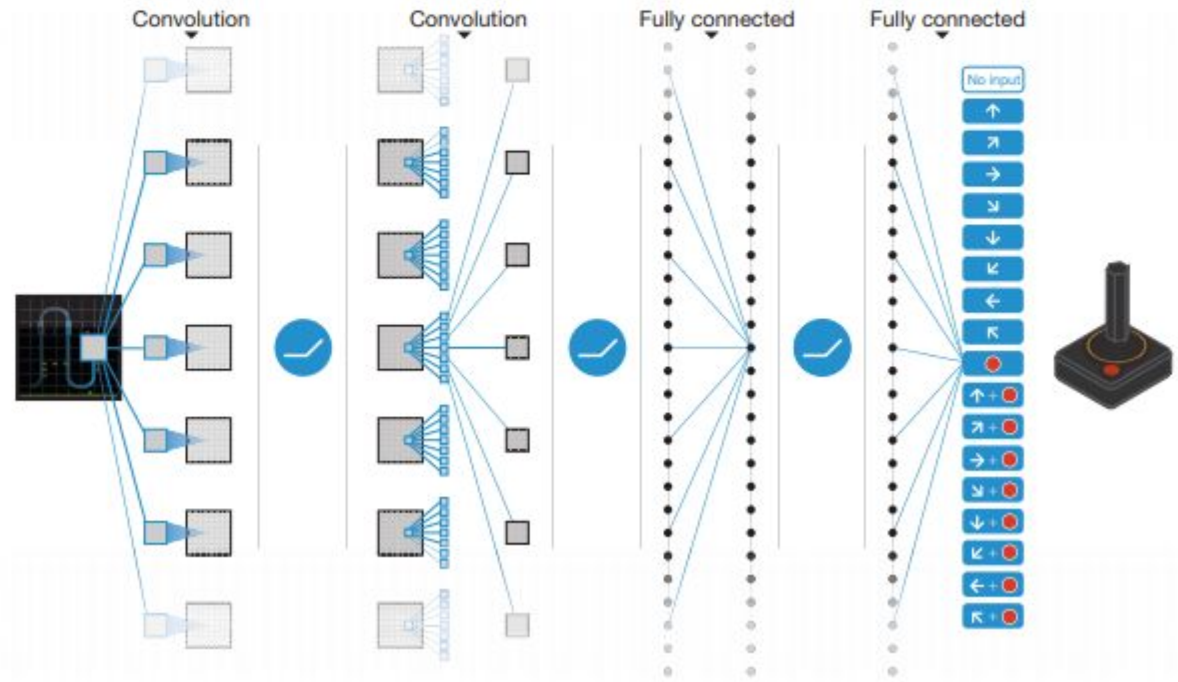
$$= \mathbb{E}_\pi[\sum_{t=0}^{T_e}(\sum_{t'=t}^{T_e} \gamma^{t'-t} r_{t'})\nabla \log \pi(a_t|s_t)]$$

$$= \mathbb{E}_\pi[\sum_{t=0}^{T_e} V^\pi(s_t)\nabla \log \pi(a_t|s_t)]$$

b independent of action

$$= \mathbb{E}_\pi[\sum_{t=0}^{T_e}(V^\pi(s_t) - b)\nabla \log \pi(a_t|s_t)]$$

Asynchronous Methods for DRL (Mnih et al., 2016)



A3C

Asynchronous Methods for DRL (Mnih et al., 2016)

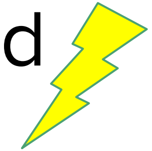
$$\max_\theta \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T_e}(V^{\pi_\theta}(s_t)-b)\log\pi_\theta(a_t|s_t)\right]$$

$$b=\tilde{V}_\phi(s_t)$$

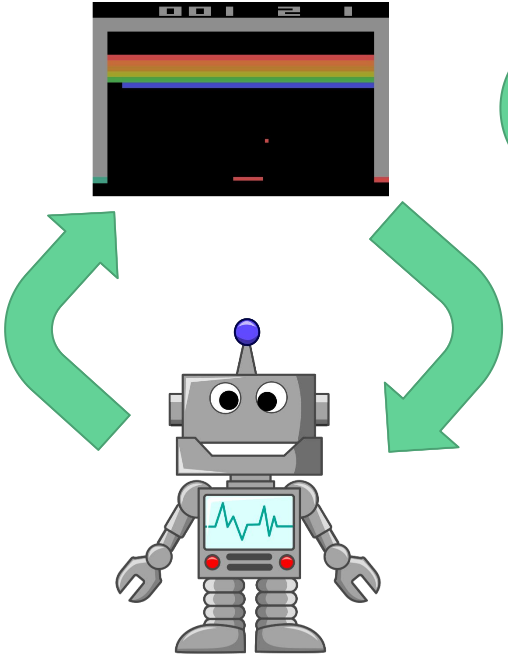
$$V^{\pi_\theta}(s_t)\approx\sum_{t'=t}^{t+n}\gamma^{t'-t}r_{t'}+\gamma^n\tilde{V}_\phi(s_{t+n})$$

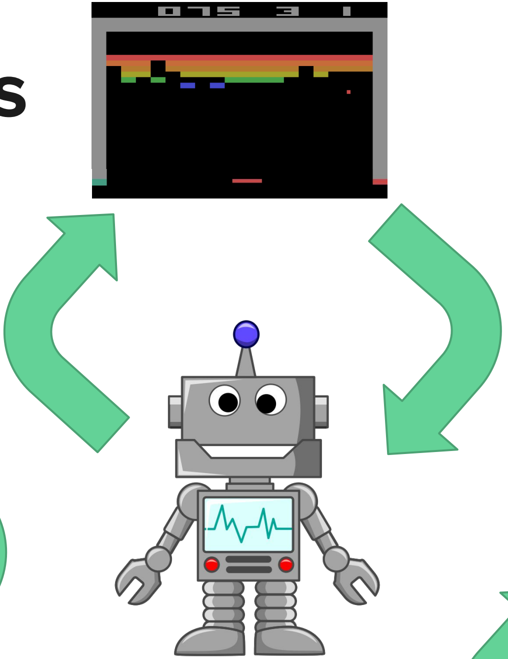
A3C

Deep Learning Works for... RL?

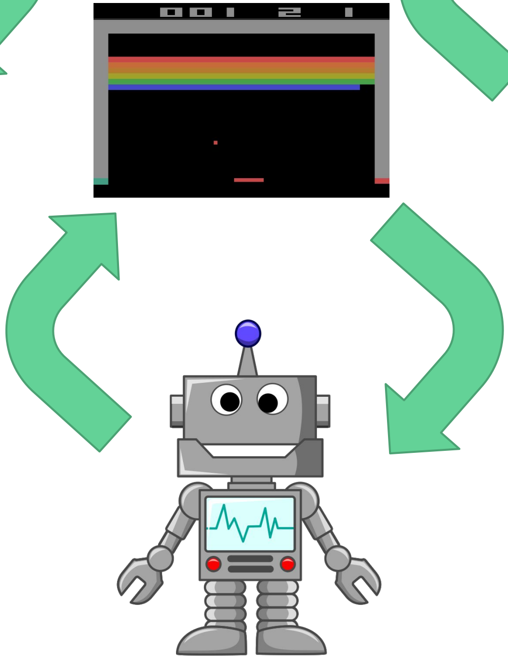
- ...~~large data~~ sets...
  many interactions
- ...with ~~labeled~~ data points... ➜ multi-step target
  self-labeled
- ...which are iid 
  ➜ multiple actors
  ➜ entropy regularization

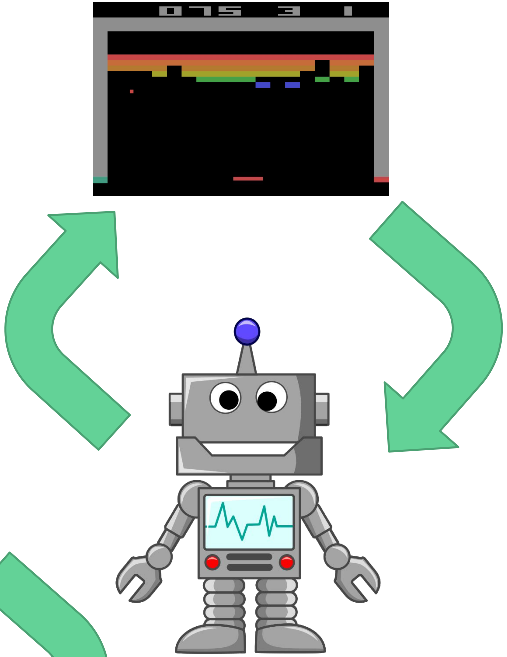
Multiple Actors





Learn from all rollouts





Entropy Regularization

… act as random as possible

$$\max_\theta \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T_e} (V^{\pi_\theta}(\boldsymbol{s}_t) - b) \log \pi_\theta(a_t|\boldsymbol{s}_t) \right.$$
$$\left. - \lambda \pi_\theta(a_t|\boldsymbol{s}_t) \log \pi_\theta(a_t|\boldsymbol{s}_t)\right]$$

DQN vs A3C

DQN	A3C
sample efficient	sample inefficient
slow to train	fast to train
(almost) deterministic	stochastic
only 1 network	2 (1.5) networks

Coding Challenge

https://github.com/OliverRichter/Coding_Challenge

DRL in the bigger picture

- Contextual Multi-Armed Bandits
- Model Predictive Control
- Optimal Control

Policy Gradient Derivation

$$
\begin{aligned}
\nabla \mathbb{E}_\pi[R(\tau)] &= \nabla \int R(\tau)\pi(\tau)d\tau \\
&= \int R(\tau)\nabla\pi(\tau)d\tau \\
&= \int R(\tau)\pi(\tau)\frac{\nabla\pi(\tau)}{\pi(\tau)}d\tau \\
&= \int R(\tau)\pi(\tau)\nabla\log\pi(\tau)d\tau \\
&= \mathbb{E}_\pi[R(\tau)\nabla\log\pi(\tau)]
\end{aligned}
$$

Markov

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

$$\rightarrow \nabla_\theta \log \pi_\theta(\tau) = \sum_{t=0}^{T_e}\nabla_\theta \log \pi_\theta(a_t|s_t)$$

$$\mathbb{E}_\pi[R(\tau)\nabla \log \pi(\tau)]$$

$$= \mathbb{E}_\pi[(\sum_{t=0}^{T_e}\gamma^t r_t)(\sum_{t=0}^{T_e}\nabla \log \pi(a_t|s_t))]$$