强化学习2022
第1节

涉及知识点：
强化学习技术概览、探索与利用

强化学习简介

张伟楠 – 上海交通大学
http://wnzhang.net

张伟楠 – 上海交通大学

电院John中心、计算机系、APEX实验室副教授 wnzhang@sjtu.edu.cn

讲者简介: 2016年UCL博士，2011年交大ACM班本科，主要研究深度学习、强化学习及其在数据挖掘场景中的落地。

课程大纲

强化学习基础部分

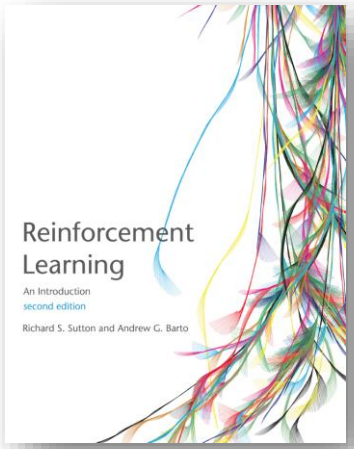
1. 强化学习、探索与利用
2. MDP和动态规划
3. 值函数估计
4. 无模型控制方法
5. 规划与学习
6. 参数化的值函数和策略
7. 深度强化学习价值方法
8. 深度强化学习策略方法

强化学习前沿部分

9. 基于模型的深度强化学习
10. 模仿学习
11. 离线强化学习
12. 参数化动作空间
13. 目标导向的强化学习
14. 多智能体强化学习
15. 强化学习大模型
16. 技术交流与回顾

课程参考材料

- 本课程没有官方教材，以下内容作为参考资料
- 参考书



Rich Sutton

Andew Barto

http://incompleteideas.net/book/RLbook2020.pdf

- 参考课程
  - UCL David Silver RL Course: https://www.davidsilver.uk/teaching/
  - Berkeley Sergey Levine Deep RL Course: http://rail.eecs.berkeley.edu/deeprlcourse/
  - OpenAI DRL Camp: https://sites.google.com/view/deep-rl-bootcamp/lectures
  - RL China Camp: http://rlchina.org/

课程评分机制

- 没有笔试
- 5个notebook代码小作业（50%）
  - 每个小作业包括代码填空和问答题，可能会包含探索性小课题
  - 掌握强化学习基础知识的原理，并通过编程获得实践经验
- 1个大作业（第6周给出，30%）
  - 在几个大方向上选择一个方向（去年：Imitation Learning、MARL、GoRL）
  - 以探索创新为主，并不要求性能打榜，最终产出4+页的mini paper（及附件和代码）
- 答辩展示（第16周，10%）：展示大作业的研究问题、采用的模型、实验结果
- 出勤（10%）

本课程辅助学习平台1: 伯禹ElitesAI强化学习课程

注册后免费获取强化学习课程
https://www.boyuai.com/rl

本课程辅助学习平台2:《动手学强化学习》

DQN 代码实践

接下来，我们就开始正式进入 DQN 算法的代码实践环节，我们采用的测试环境 CartPole-v0 的状态空间相对简单，只有 4 个变量，因此我们的网络结构设计也相对简单。采用一层 128 个神经元的全连接并以 ReLU 作为激活函数。当遇到更复杂的诸如以图像作为输入的环境时，我们可以考虑采用深度卷积网络。

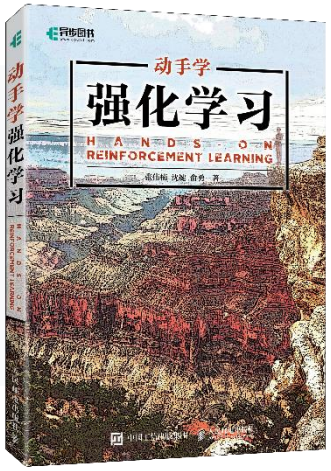
```
import random
import gym
import numpy as np
import collections
from tqdm import tqdm
import torch
import torch.nn.functional as F
import matplotlib.pyplot as plt
import rl_utils
```

我们首先定义经验回放池的类，主要包括加入数据、采样数据两大函数。

```
class ReplayBuffer:
    ''' 经验回放池 '''
    def __init__(self, capacity):
        self.buffer = collections.deque(maxlen=capacity) # 队列，先进先出

    def add(self, state, action, reward, next_state, done):
        self.buffer.append((state, action, reward, next_state, done)) # 将数据加入buffer

    def sample(self, batch_size): # 从buffer中采样数据，数量为batch_size
        transitions = random.sample(self.buffer, batch_size)
        state, action, reward, next_state, done = zip(*transitions)
        return np.array(state), action, reward, np.array(next_state), done
```

https://hrl.boyuai.com/ Hands-on RL 动手学强化学习 本课程的最新课件也会持续在此更新

课程大纲

强化学习基础部分

1. 强化学习、探索与利用
2. MDP和动态规划
3. 值函数估计
4. 无模型控制方法
5. 规划与学习
6. 参数化的值函数和策略
7. 深度强化学习价值方法
8. 深度强化学习策略方法

强化学习前沿部分

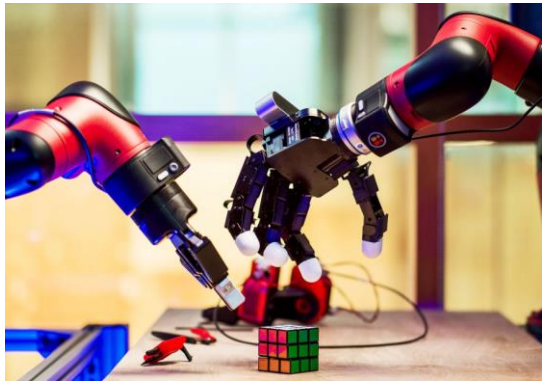
9. 基于模型的深度强化学习
10. 模仿学习
11. 离线强化学习
12. 参数化动作空间
13. 目标导向的强化学习
14. 多智能体强化学习
15. 强化学习大模型
16. 技术交流与回顾

强化学习技术概览

张伟楠 – 上海交通大学

两种人工智能任务类型

- 预测型任务
  - 根据数据预测所需输出（有监督学习）
  - 生成数据实例（无监督学习）
- 决策型任务
  - 在动态环境中采取行动（强化学习）
    - 转变到新的状态
    - 获得即时奖励
    - 随着时间的推移最大化累计奖励
    - Learning from interaction in a trial-and-error manner



决策和预测的不同

- 决策下达到环境中，直接改变环境
  - 未来发展随之改变
  - “做决策，担责任”
- 预测仅产生信号，不考虑环境的改变
  - 预测的信号是否用、怎么用，不需要考虑

智慧医疗

- 决策亲自改变世界
  - 医生或者AI直接给病人下达治疗方案
- 预测辅助别人改变世界
  - AI告诉医生病人可能的得病预测，医生综合各方面判断最后给病人下达治疗方案

序贯决策（Sequential Decision Making）

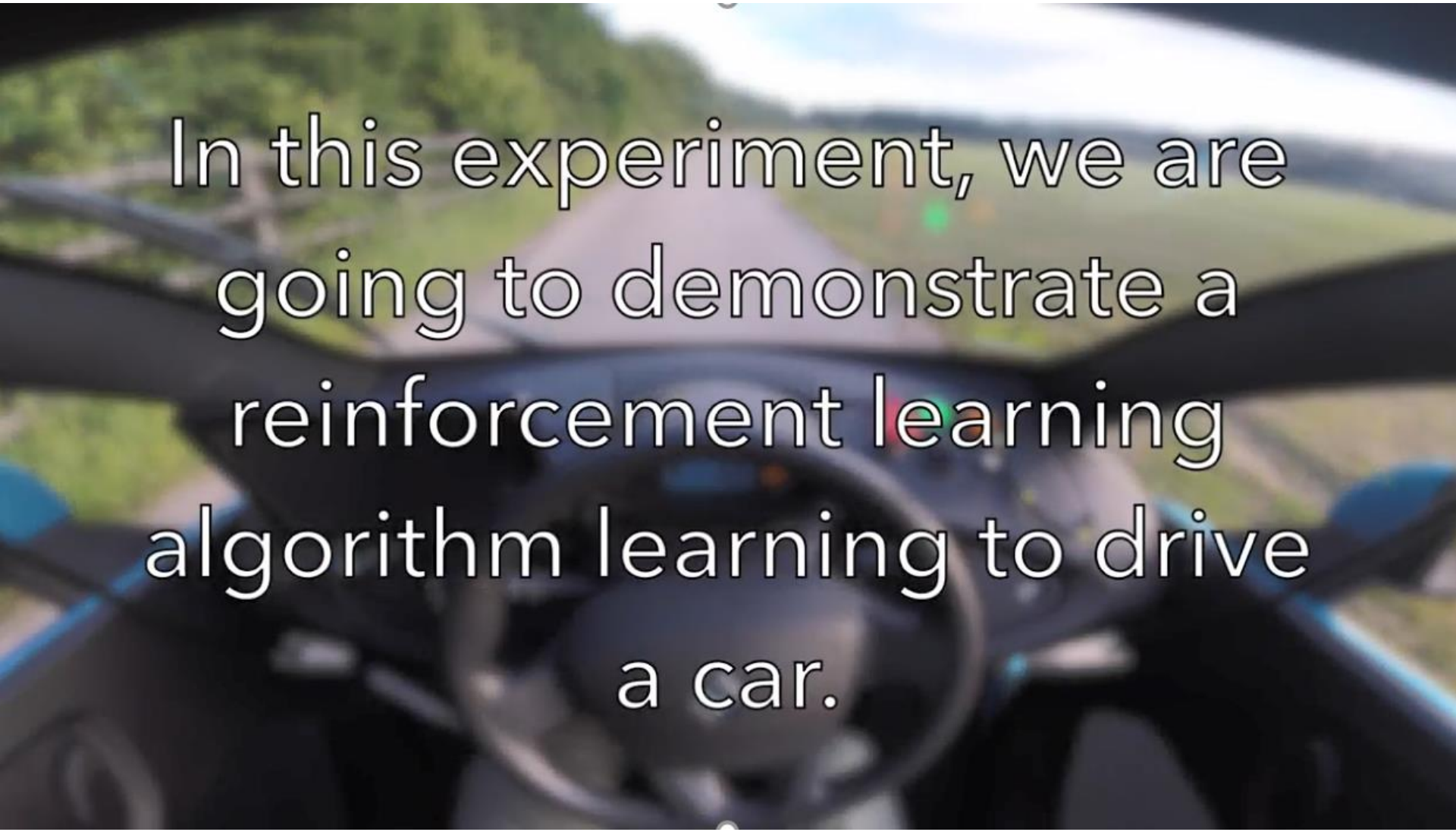
序贯决策

- 决策者序贯地做出一个个决策，并接续看到新的观测，直到最终任务结束。

Sequential decision making describes a situation where the decision maker (DM) makes successive observations of a process before a final decision is made.

绝大多数序贯决策问题，可以用强化学习来解

强化学习应用案例：无人驾驶小车



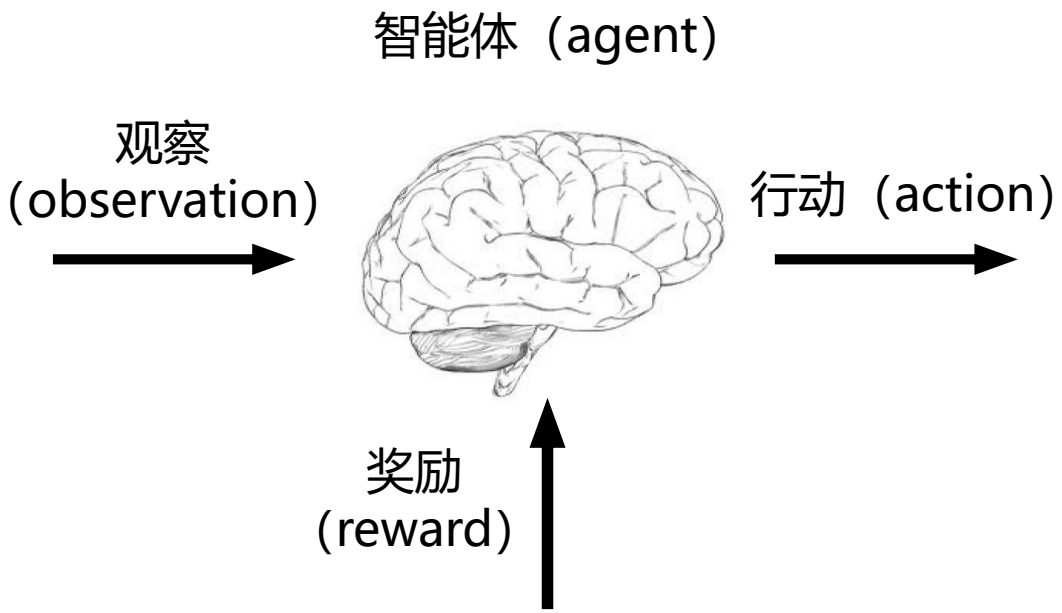
Alex Kendall et.al, Learning to Drive in a Day. ICRA 2019: 8248-8254
https://www.youtube.com/watch?v=eRwTbRtnT1I

主要内容

- 面向决策任务的人工智能
- 强化学习的基础概念和研究前沿
- 强化学习的落地现状与挑战

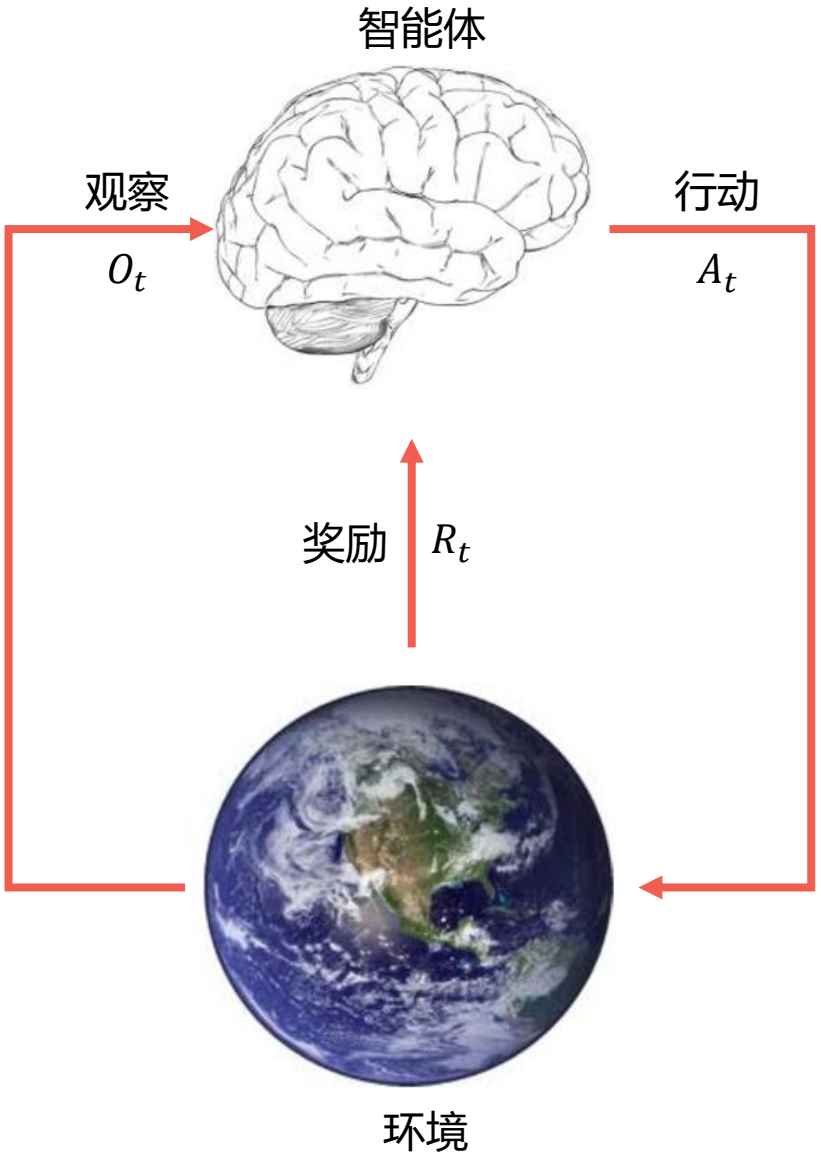
强化学习定义

- 通过从交互中学习来实现目标的计算方法



- 三个方面：
  - 感知：在某种程度上感知环境的状态
  - 行动：可以采取行动来影响状态或者达到目标
  - 目标：随着时间推移最大化累积奖励

强化学习交互过程



- 在每一步 t，智能体：
  - 获得观察 O_t
  - 获得奖励 R_t
  - 执行行动 A_t
- 环境：
  - 获得行动 A_t
  - 给出观察 O_{t+1}
  - 给出奖励 R_{t+1}
- t 在环境这一步增加

在与动态环境的交互中学习

有监督、无监督学习

Model ⟵

Fixed Data

强化学习

Agent ⟷

Dynamic Environment

Agent不同，交互出的数据也不同！

强化学习系统要素

- 历史（History）是观察、行动和奖励的序列

$$H_t = O_1, R_1, A_1, O_2, R_2, A_2, \ldots, O_{t-1}, R_{t-1}, A_{t-1}, O_t, R_t$$

  - 即，一直到时间t为止的所有可观测变量
  - 根据这个历史可以决定接下来会发生什么
    - 智能体选择行动
    - 环境选择观察和奖励

- 状态（state）是一种用于确定接下来会发生的事情（行动、观察、奖励）的信息
  - 状态是关于历史的函数

$$S_t = f(H_t)$$

强化学习系统要素

- 策略（Policy）是学习智能体在特定时间的行为方式
  - 是从状态到行动的映射
  - 确定性策略（Deterministic Policy）

$$a = \pi(s)$$

  - 随机策略（Stochastic Policy）

$$\pi(a|s) = P(A_t = a \mid S_t = s)$$

强化学习系统要素

- 奖励（Reward） $R(s,a)$
  - 一个定义强化学习目标的标量
  - 能立即感知到什么是“好”的
- 价值函数（Value Function）
  - 状态价值是一个标量，用于定义对于长期来说什么是“好”的
  - 价值函数是对于未来累积奖励的预测
    - 用于评估在给定的策略下，状态的好坏

$$Q_\pi(s,a) = \mathbb{E}_\pi[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots | S_t = s, A_t = a]$$
$$= \mathbb{E}_\pi[R_{t+1} + \gamma Q_\pi(s',a') | S_t = s, A_t = a]$$

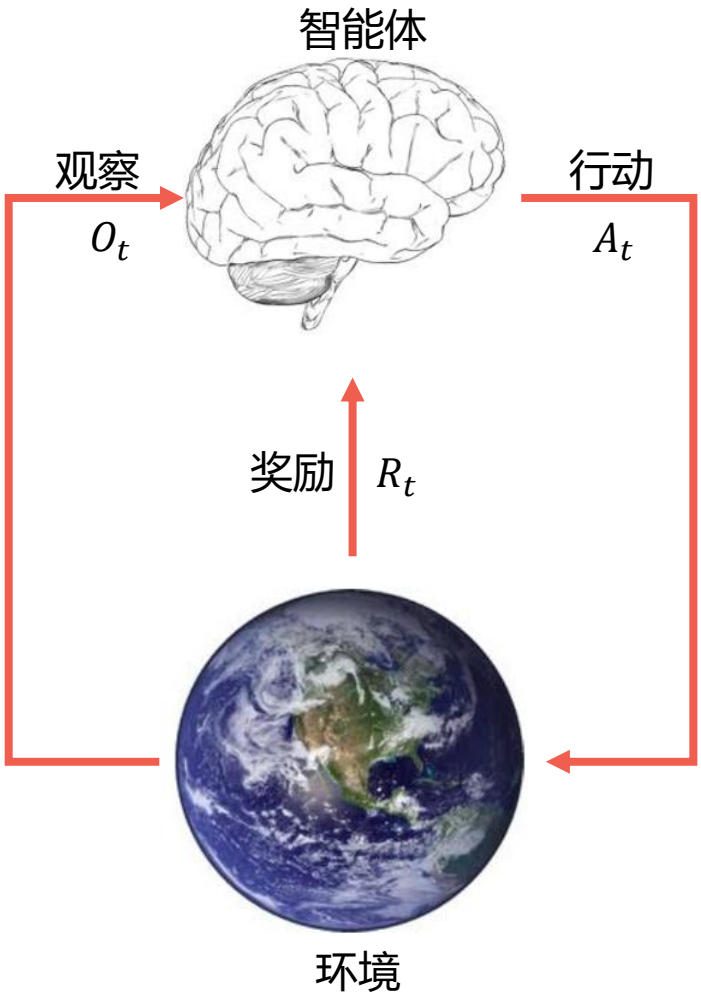
强化学习系统要素

- 环境的模型（Model）用于模拟环境的行为
  - 预测下一个状态

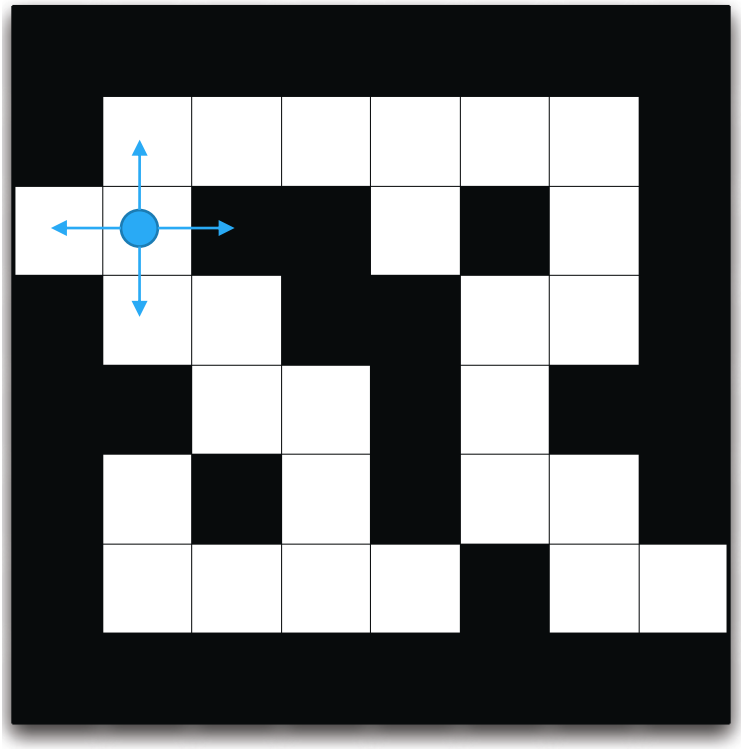
$$\mathcal{P}_{ss'}^{a} = \mathbb{P}[S_{t+1} = s' | S_t = s, A_t = a]$$

  - 预测下一个（立即）奖励

$$\mathcal{R}_{s}^{a} = \mathbb{E}[R_{t+1} | S_t = s, A_t = a]$$

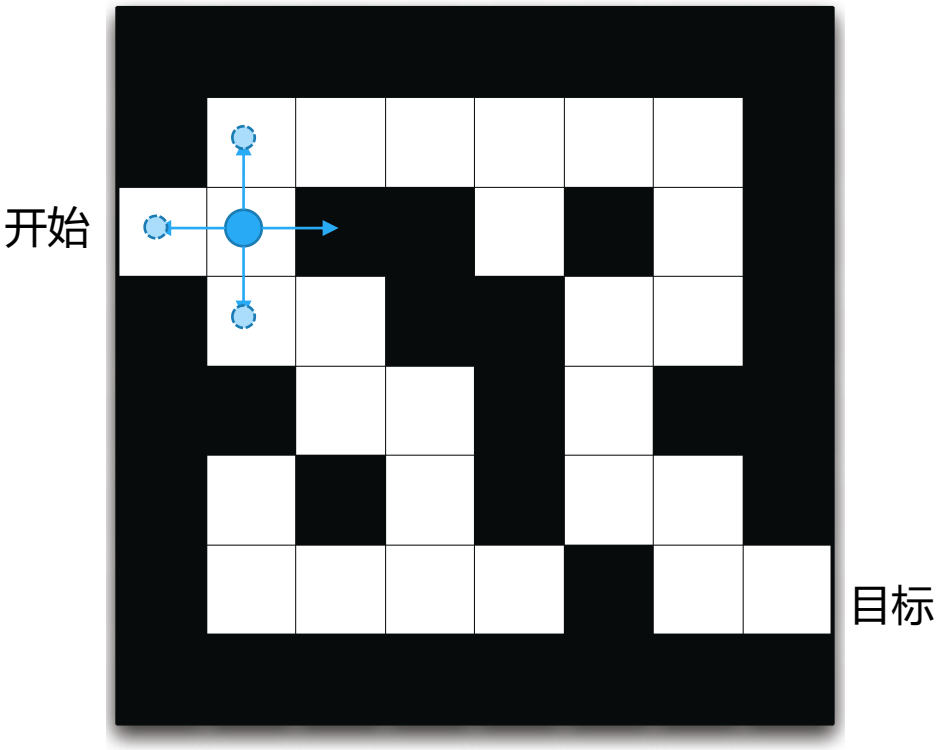


举例：迷宫



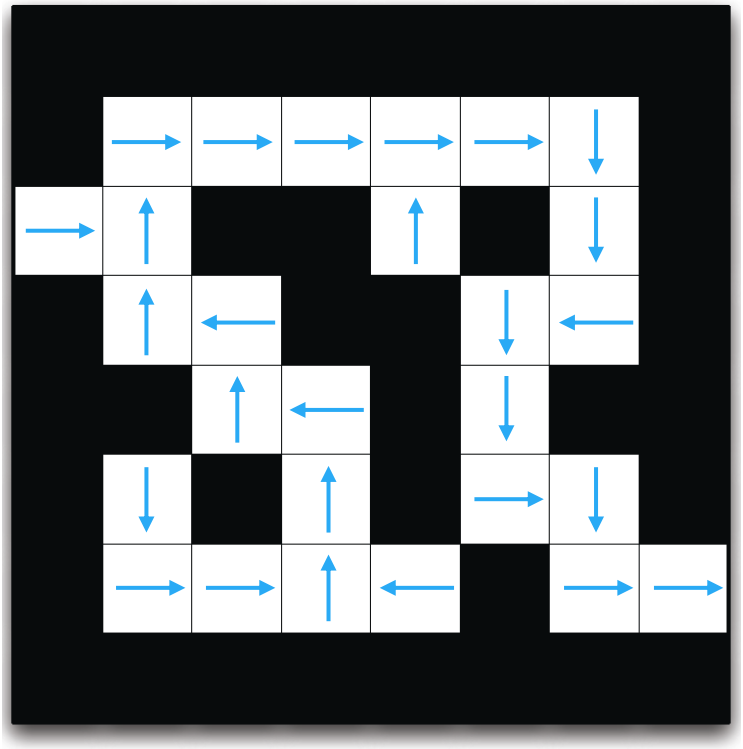
- 状态：智能体的位置
- 行动：N,E,S,W

举例：迷宫



- 状态：智能体的位置
- 行动：N,E,S,W
- 状态转移：根据行动方向朝下一格移动
  - 如果行动的方向是墙则不动

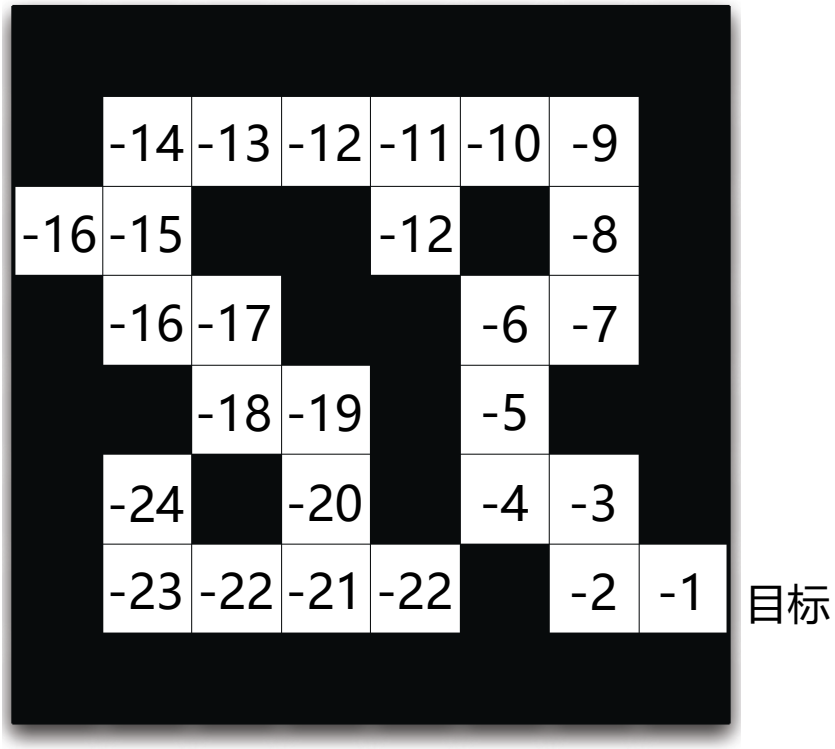
举例：迷宫



- 状态：智能体的位置
- 行动：N,E,S,W
- 状态转移：根据行动方向朝下一格移动
- 奖励：每一步为-1

- 给定一个上图所示的策略
  - 箭头表示每一个状态s下的策略$\pi(s)$

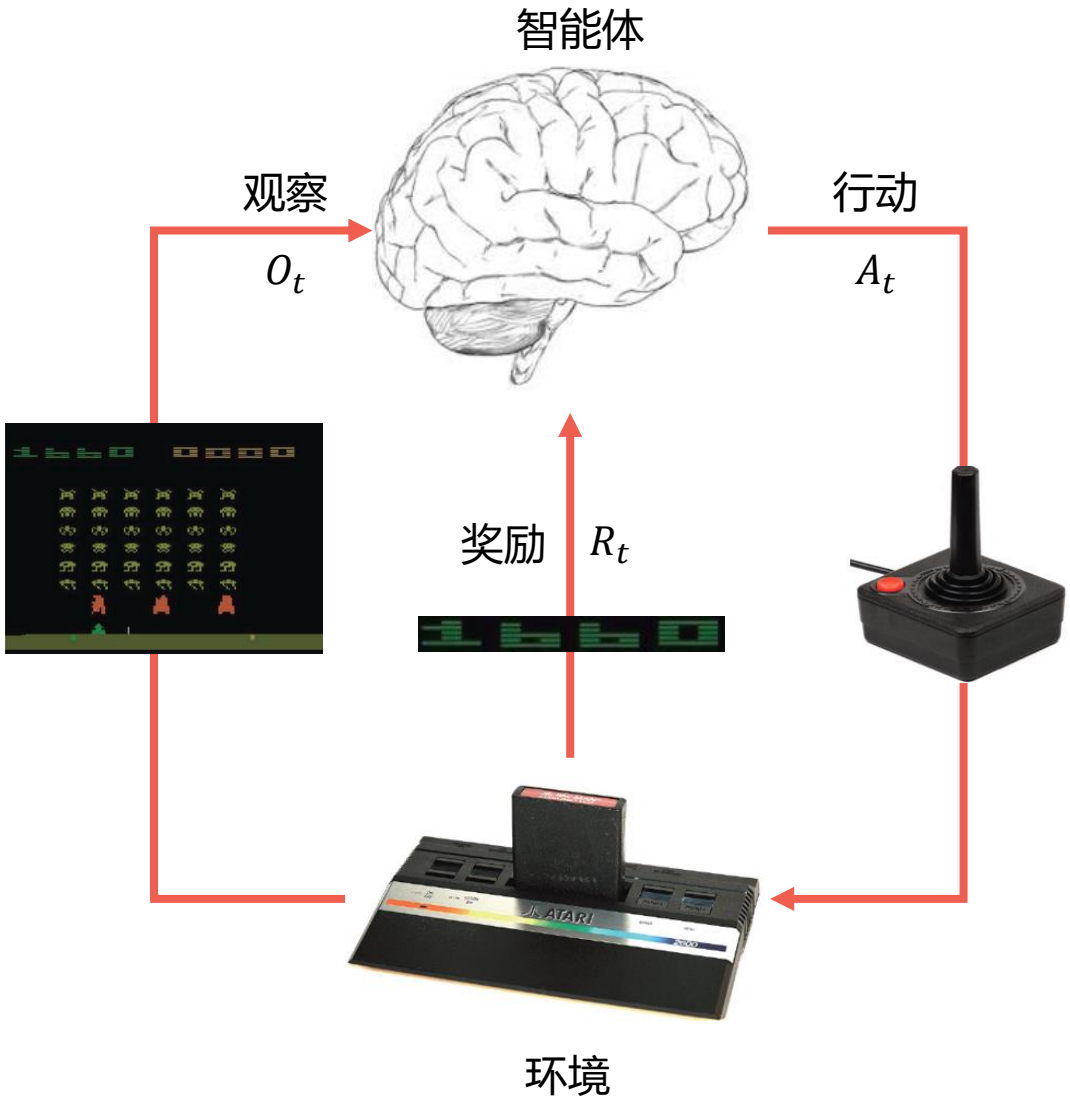
举例：迷宫



- 状态：智能体的位置
- 行动：N,E,S,W
- 状态转移：根据行动方向朝下一格移动
- 奖励：每一步为-1

- 数字表示每一个状态s下的状态价值$v_\pi(s)$

举例：Atari游戏



- 游戏规则未知
- 从交互游戏中进行学习
- 在操纵杆上选择行动并查看分数和像素画面

强化学习的方法分类

- 基于价值：知道什么是好的什么是坏的
  - 没有策略（隐含）
  - 价值函数

$$\nabla_{\theta_i} L_i(\theta_i) = \mathbb{E}_{s,a\sim\rho(\cdot);s'\sim\mathcal{E}}\left[\left(r + \gamma \max_{a'} Q(s', a'; \theta_{i-1}) - Q(s, a; \theta_i)\right) \nabla_{\theta_i} Q(s, a; \theta_i)\right]$$

- 基于策略：知道怎么行动
  - 策略
  - 没有价值函数

- Actor-Critic：学生听老师的
  - 策略
  - 价值函数

价值和策略近似

$Q_\theta(s,a)$

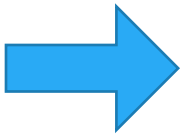
	a_1	a_2
s_1	3.2	4.1
s_2	0.8	1.2
s_3	2.3	-0.3

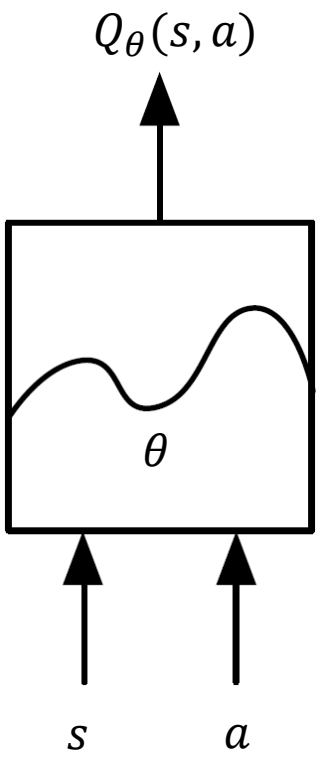
价值函数表格

$\pi_\theta(a|s)$

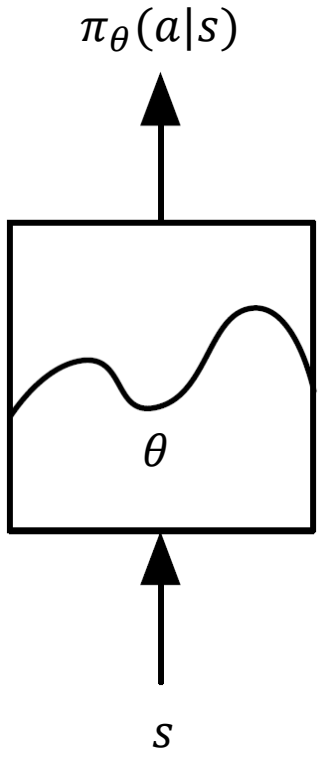
	a_1	a_2
s_1	0.2	0.8
s_2	0.4	0.6
s_3	0.9	0.1

策略表格





价值函数近似



策略近似

- 假如我们直接使用深度神经网络建立这些近似函数呢?
- 深度强化学习!

深度强化学习的崛起

- 2012年AlexNet在ImageNet比赛中大幅度领先对手获得冠军
- 2013年12月，第一篇深度强化学习论文出自NIPS 2013 Reinforcement Learning Workshop

Playing Atari with Deep Reinforcement Learning

Volodymyr Mnih **Koray Kavukcuoglu** **David Silver** **Alex Graves** **Ioannis Antonoglou**

Daan Wierstra **Martin Riedmiller**

DeepMind Technologies

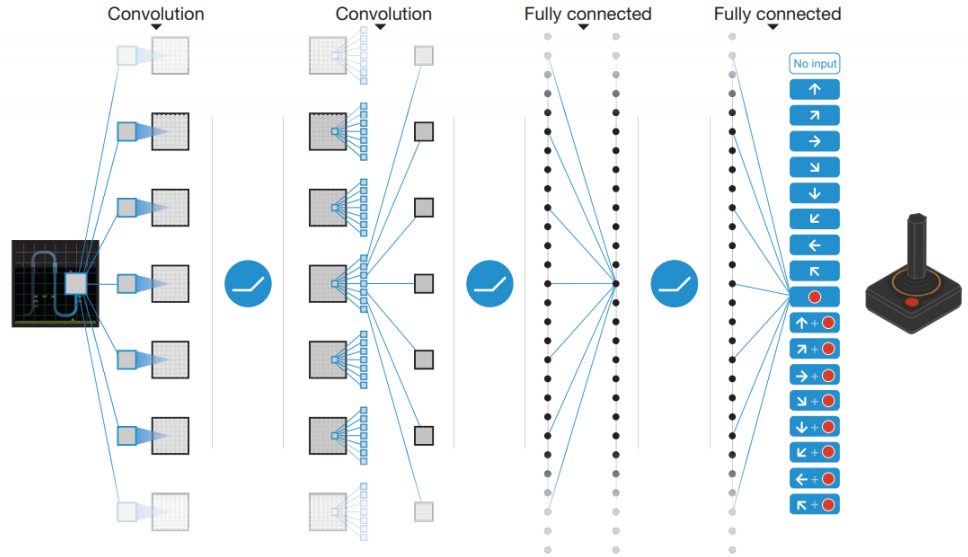
{vlad,koray,david,alex.graves,ioannis,daan,martin.riedmiller} @ deepmind.com

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

深度强化学习

深度强化学习

- 利用深度神经网络进行价值函数和策略近似
- 从而使强化学习算法能够以端到端的方式解决复杂问题

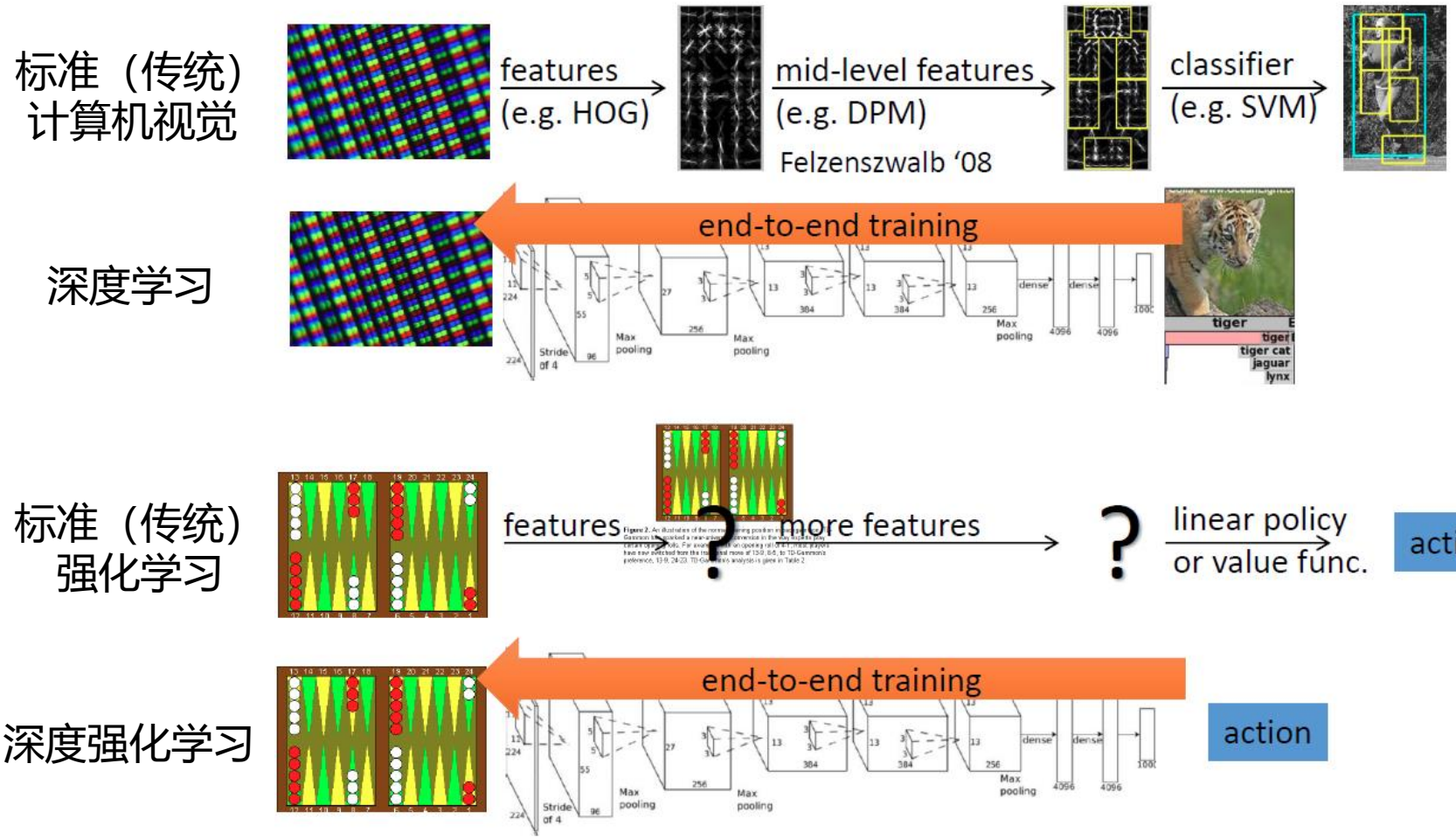


$$\nabla_{\theta_i} L_i(\theta_i) = \mathbb{E}_{s,a\sim\rho(\cdot);s'\sim\mathcal{E}}\left[\left(r + \gamma \max_{a'} Q(s', a'; \theta_{i-1}) - Q(s, a; \theta_i)\right) \nabla_{\theta_i} Q(s, a; \theta_i)\right]$$

Q函数的参数通过神经网络反向传播学习

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

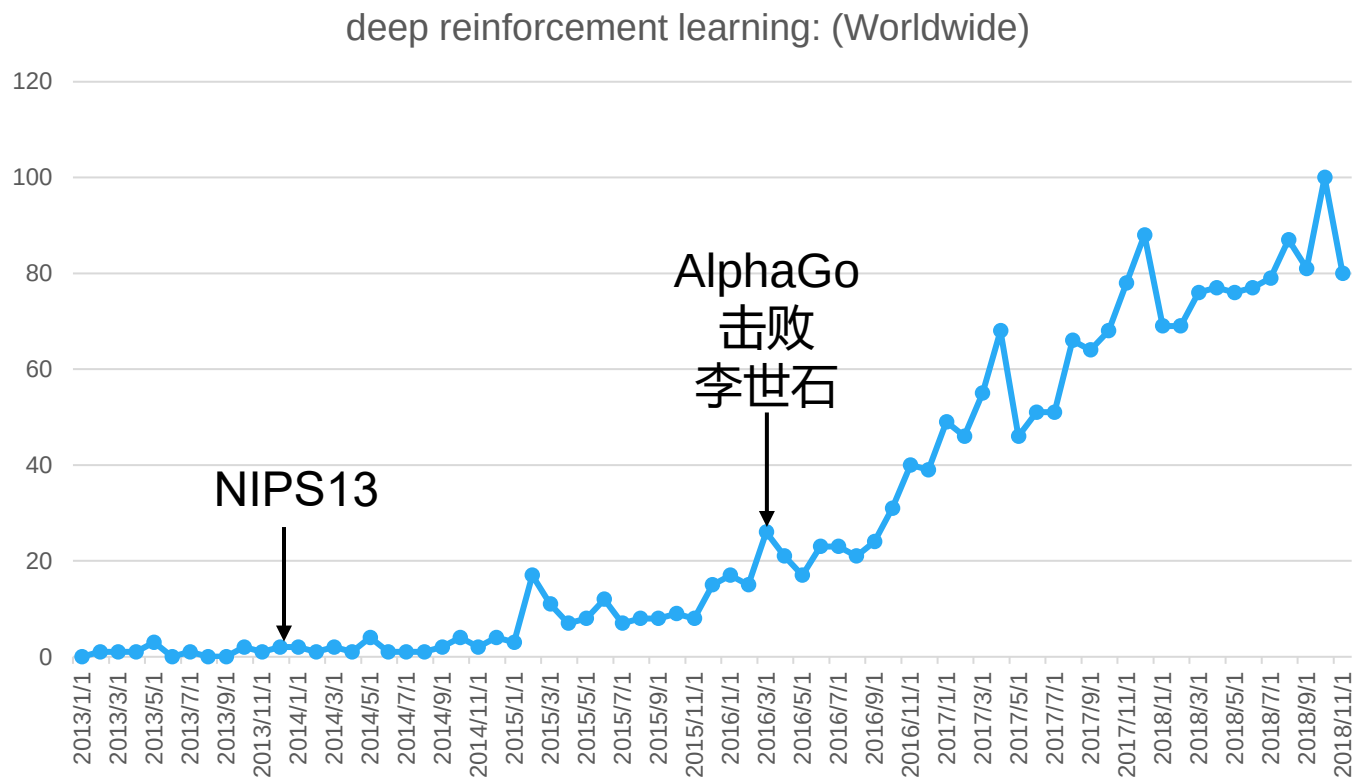
端到端强化学习



- 深度强化学习使强化学习算法能够以端到端的方式解决复杂问题
- 从一项实验室学术技术变成可以产生GDP的实际技术

Slide from Sergey Levine. http://rail.eecs.berkeley.edu/deeprlcourse/static/slides/lec-1.pdf

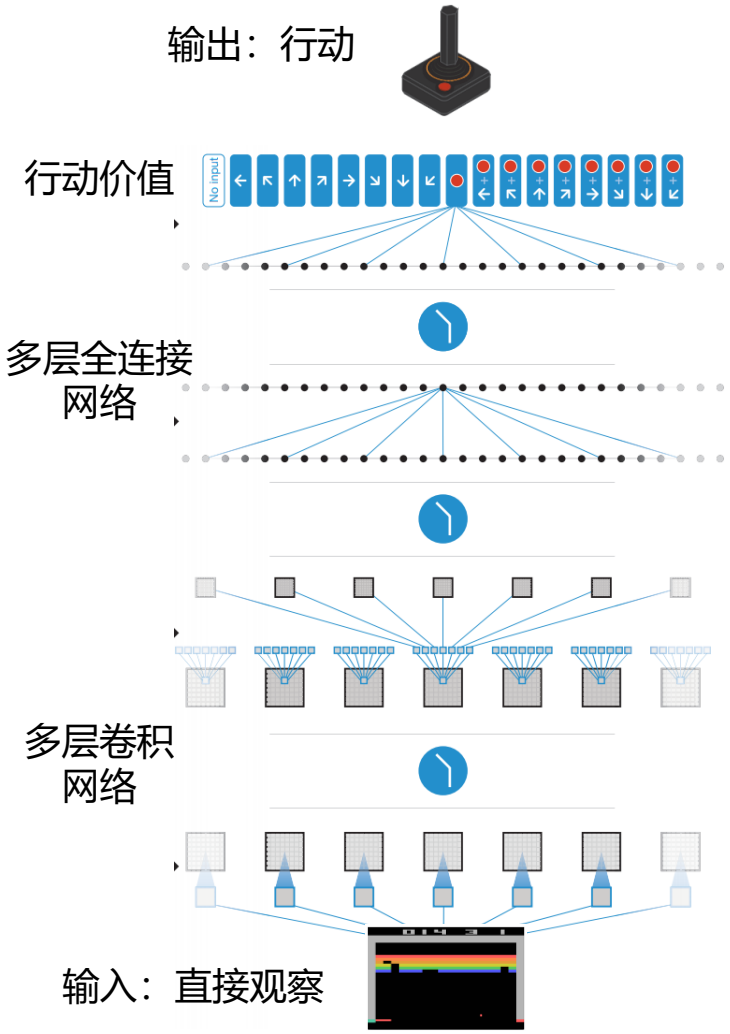
深度强化学习趋势



- Google搜索中词条“深度强化学习（deep reinforcement learning）”的趋势

深度强化学习带来的关键变化

- 将深度学习（DL）和强化学习（RL）结合在一起会发生什么？
  - 价值函数和策略变成了深度神经网络
  - 相当高维的参数空间
  - 难以稳定地训练
  - 容易过拟合
  - 需要大量的数据
  - 需要高性能计算
  - CPU（用于收集经验数据）和GPU（用于训练神经网络）之间的平衡
  - …
- 这些新的问题促进着深度强化学习算法的创新



深度Q网络（DQN）

- Q 学习算法学习一个由 θ 作为参数的函数 $Q_\theta(s,a)$
  - 更新方程 $Q_\theta(s_t,a_t) \leftarrow Q_\theta(s_t,a_t) + \alpha(r_t + \gamma \max_{a'} Q_\theta(s_{t+1},a') - Q_\theta(s_t,a_t))$

直观想法

- 使用神经网络来逼近 $Q_\theta(s,a)$，面对算法不稳定问题
  - 连续采样得到的 $\{(s_t,a_t,s_{t+1},r_t)\}$ 不满足独立分布
  - $\{(s_t,a_t,s_{t+1},r_t)\}$ 为状态-动作-下一状态-回报输入
  - $Q_\theta(s,a)$ 的频繁更新

解决办法

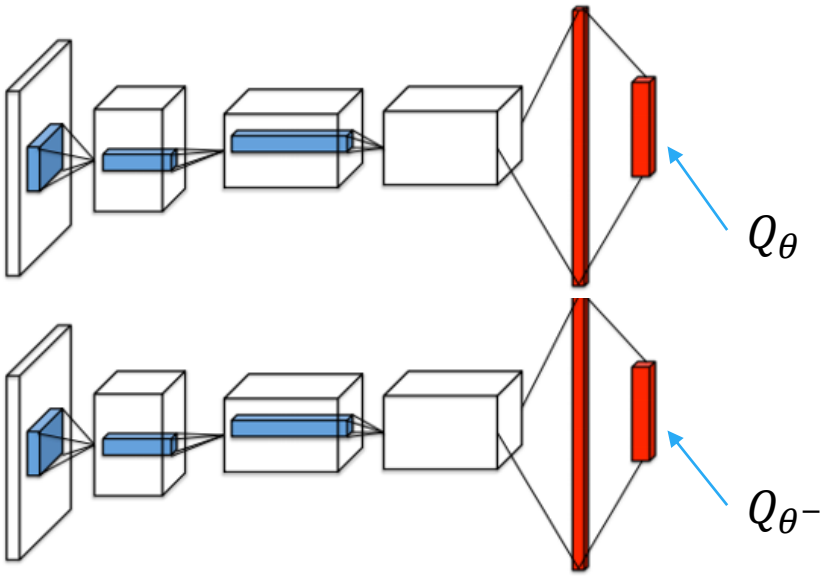
- 经验回放：均匀采样和优先经验回放
- 使用双网络结构：评估网络（evaluation network）和目标网络（target network）

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

目标网络

- 目标网络 $Q_{\theta^-}(s,a)$
  - 使用较旧的参数，记为 θ^-，每隔 C 步和训练网络的参数同步一次。
  - 第i次迭代的损失函数为

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

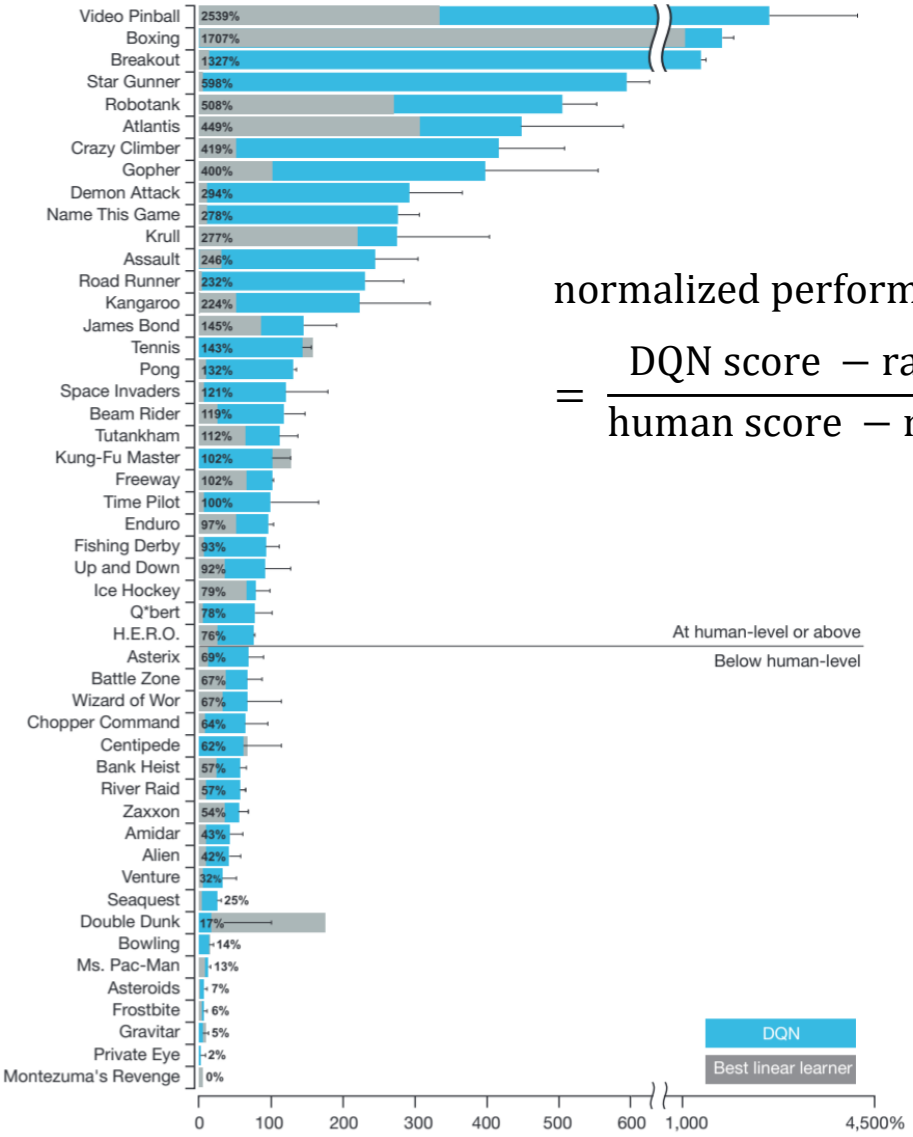


评估网络（evaluation network）

目标网络（target network）是评估网络几轮前的拷贝，不更新

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

在 Atari 环境中的实验结果



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

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)

深度强化学习的研究前沿

基于模拟模型的强化学习

- 模拟器的无比重要性

目标策动的层次化强化学习

- 长程任务的中间目标是桥梁的基石

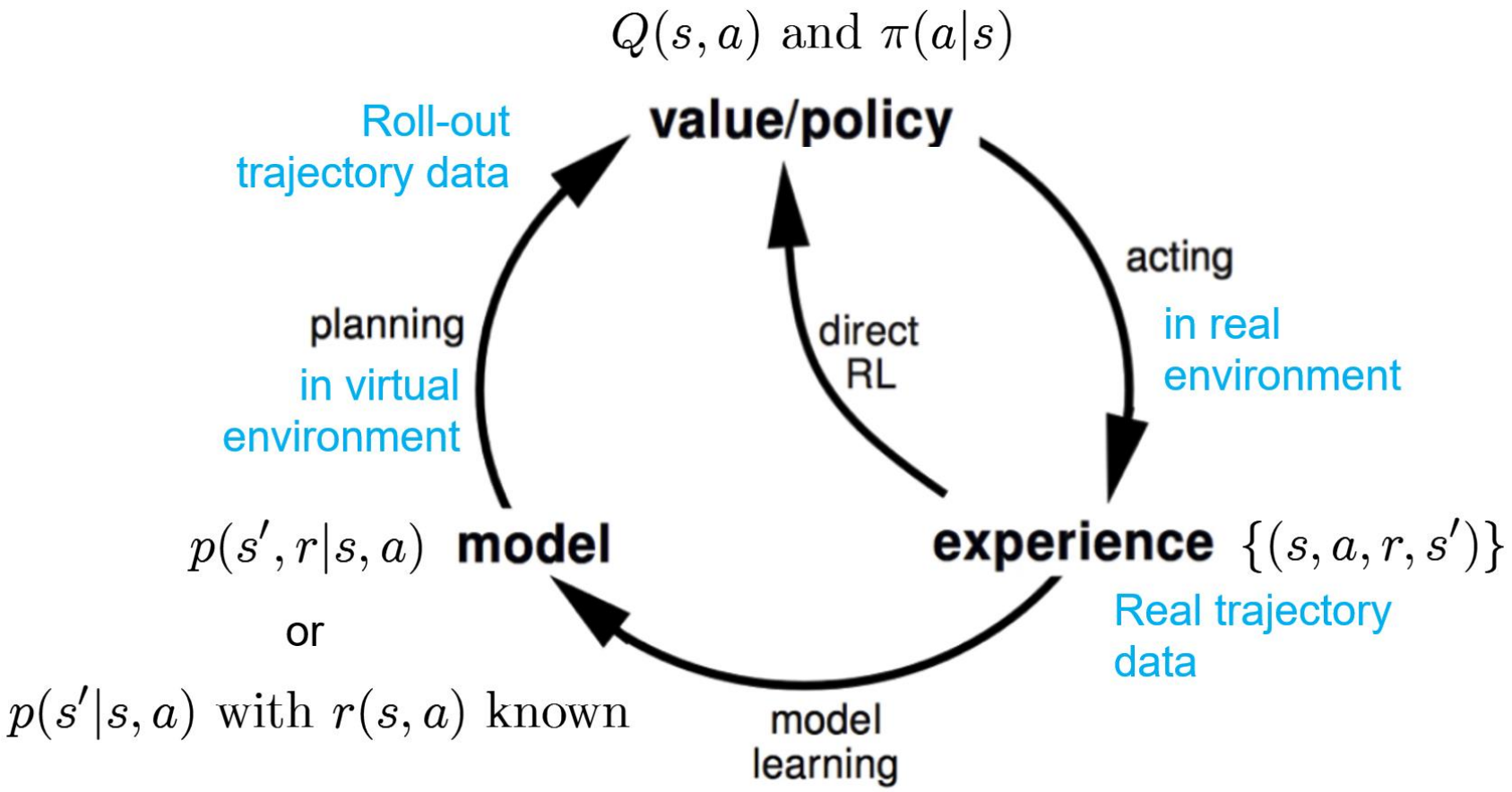
模仿学习

- 无奖励信号下跟随专家做策略学习

多智能体强化学习

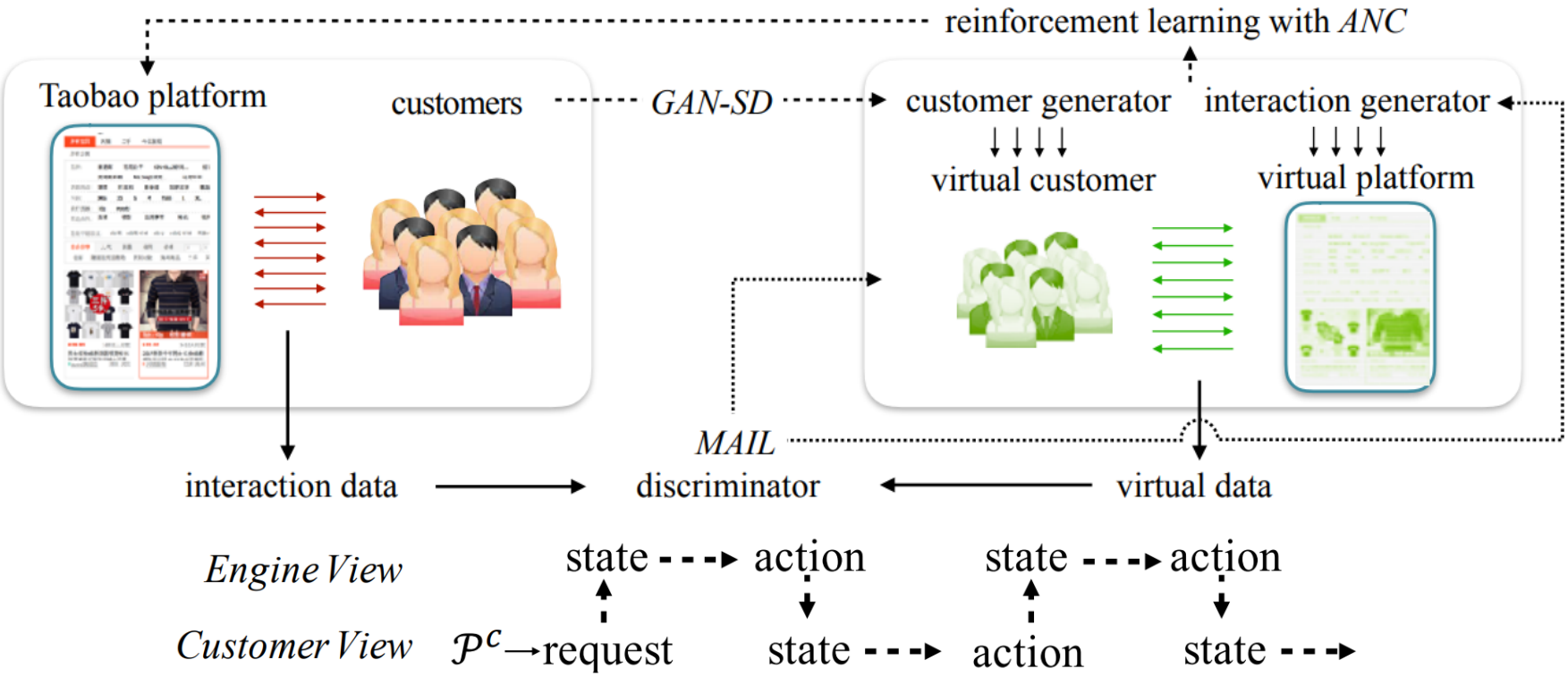
- 分散式、去中心化的人工智能

基于模拟模型的强化学习(Model-based RL)



- 建立环境模拟器，在模拟器中训练强化学习策略，减少对真实环境的影响，也可以生成更多特定场景数据

基于模拟模型的强化学习(Model-based RL)

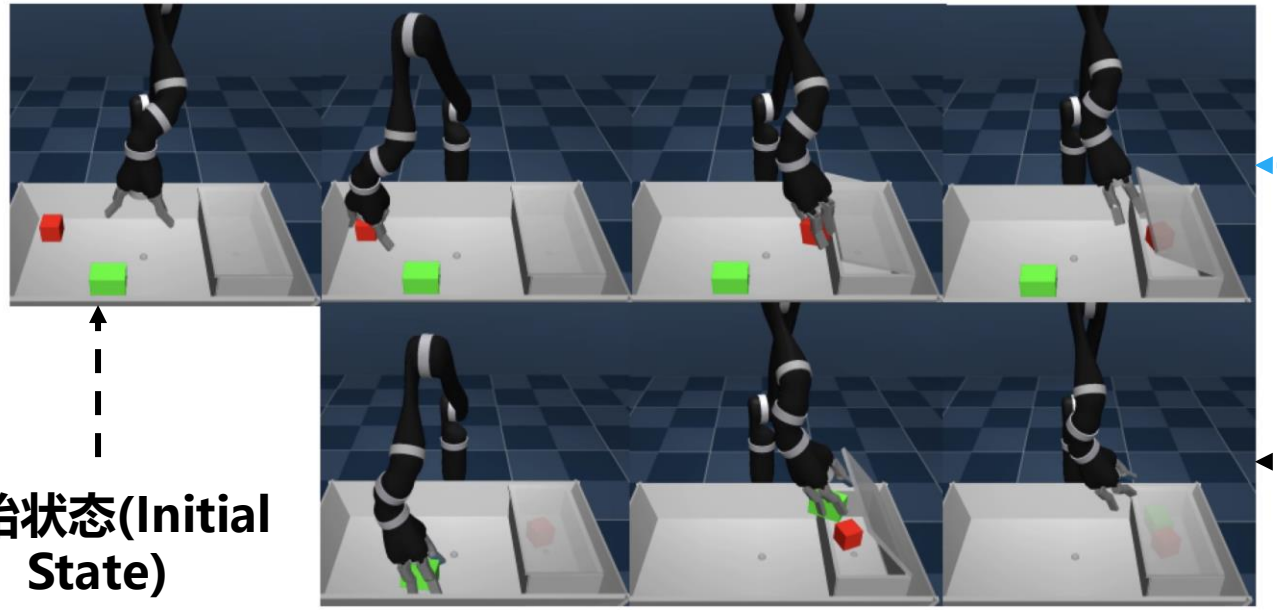


	Engine view	Customer view
State	customer feature and request	customer feature, engine action and PV info
Action	parametermized as a vector in $\mathbb{R}^d$	buy, turn page or leave

- 建立用户在电商平台行为模拟器，模拟不同商品推荐策略下用户的浏览、点击、购买等行为，进而优化电商推荐策略

Shi, Jing-Cheng, Yang Yu, et al. "Virtual-taobao: Virtualizing real-world online retail environment for reinforcement learning." AAAI 2019.

目标策动的强化学习(Goal-oriented RL)



中间状态
(Sub-Goal State)

初始状态(Initial State)

目标状态
(Goal State)

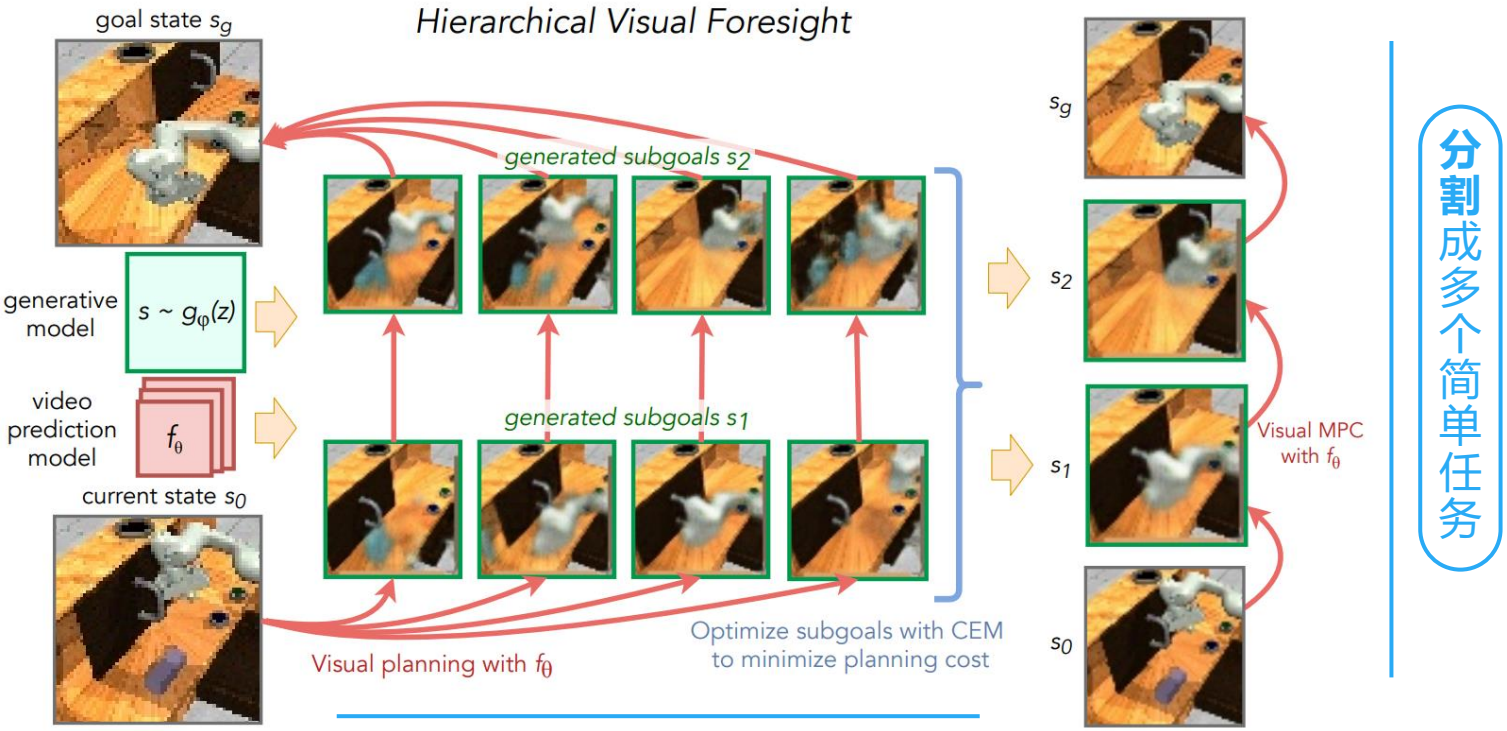
长期限任务挑战 (Long-Horizon)

- 累计建模误差(**Compounding** Model Error).
- 稀疏反馈(**Sparse** Cost).

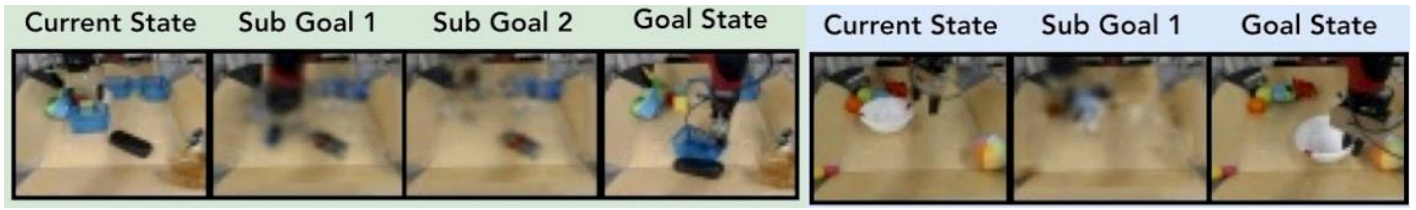
生成中间状态，将长期限任务**分割**成多个简单的短期限任务

目标策动的强化学习(Goal-oriented RL)

模拟器场景



实际机器人场景



Suraj Nair, Chelsea Finn. Hierarchical Foresight: Self-Supervised Learning of Long-Horizon Tasks via Visual Subgoal Generation. *ICLR*, 2020.

模仿学习(Imitation Learning)

Computer Games

reward

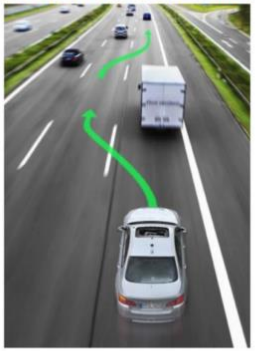
Mnih et al. '15

Real World Scenarios

robotics

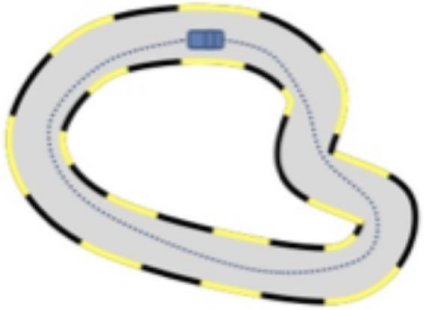
dialog

autonomous driving

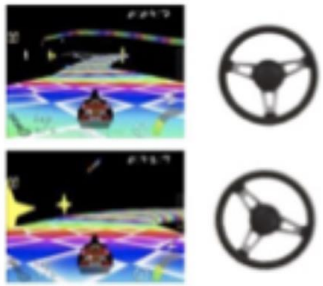


what is the reward?
often use a proxy

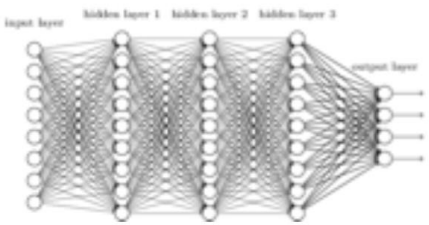
Expert Demonstrations



(*s, a*) pairs



Imitation Learning



Agent Experience

(*s, a, r, s', a'*) tuples

Reinforcement Learning

模仿学习(Imitation Learning)

Waymo has made simulation one of the pillars of its autonomous vehicle development program. But **Latent Logic** could help Waymo make its simulation more realistic by using a form of machine learning called imitation learning.

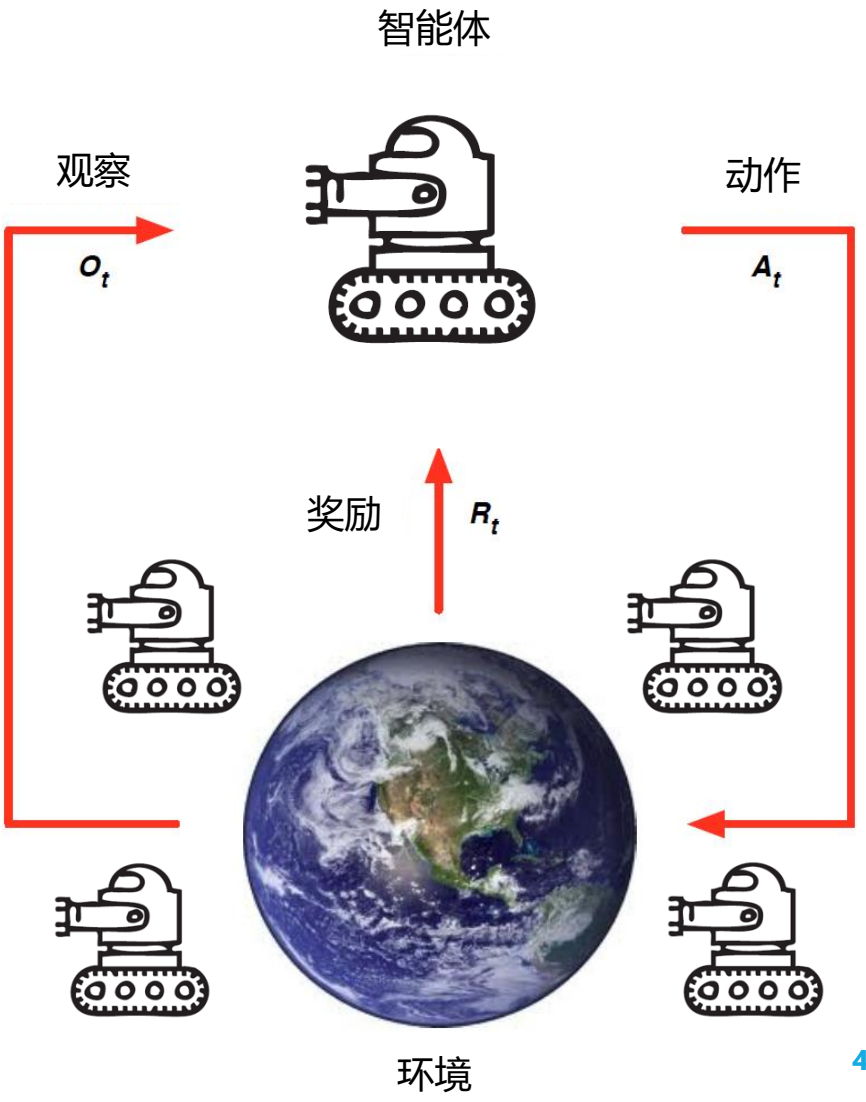
Imitation learning models human behavior of motorists, cyclists and pedestrians. The idea is that by modeling the mistakes and imperfect driving of humans, the simulation will become more realistic and theoretically improve Waymo's behavior prediction and planning.

https://techcrunch.com/2019/12/12/waymo-buys-latent-logic-drives-deeper-into-simulation-and-europe/

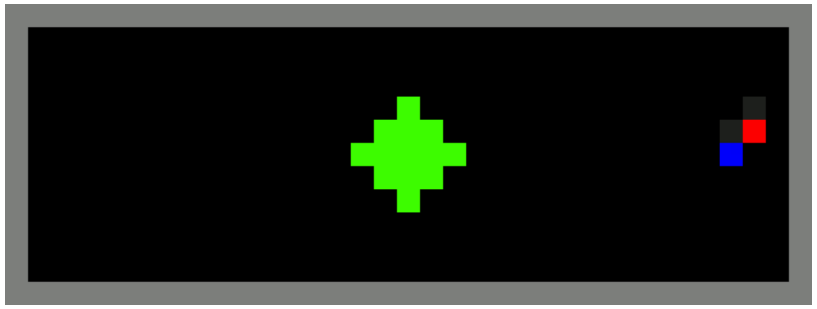
多智能体强化学习(Multi-agent RL)

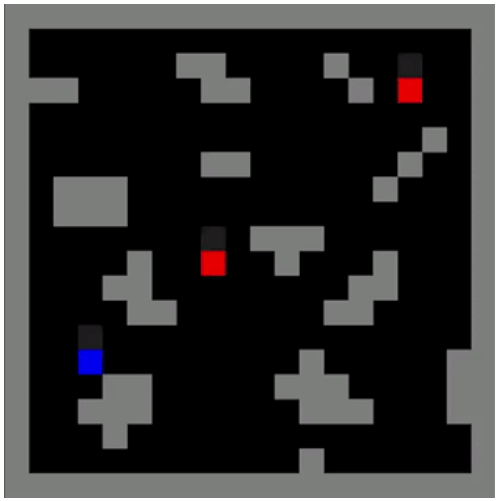
在与环境的交互过程中学习

- 环境包含有不断进行学习和更新的其他智能体
- 在任何一个智能体的视角下，环境是非稳态的（non-stationary）
  - 环境迁移的分布会发生改变

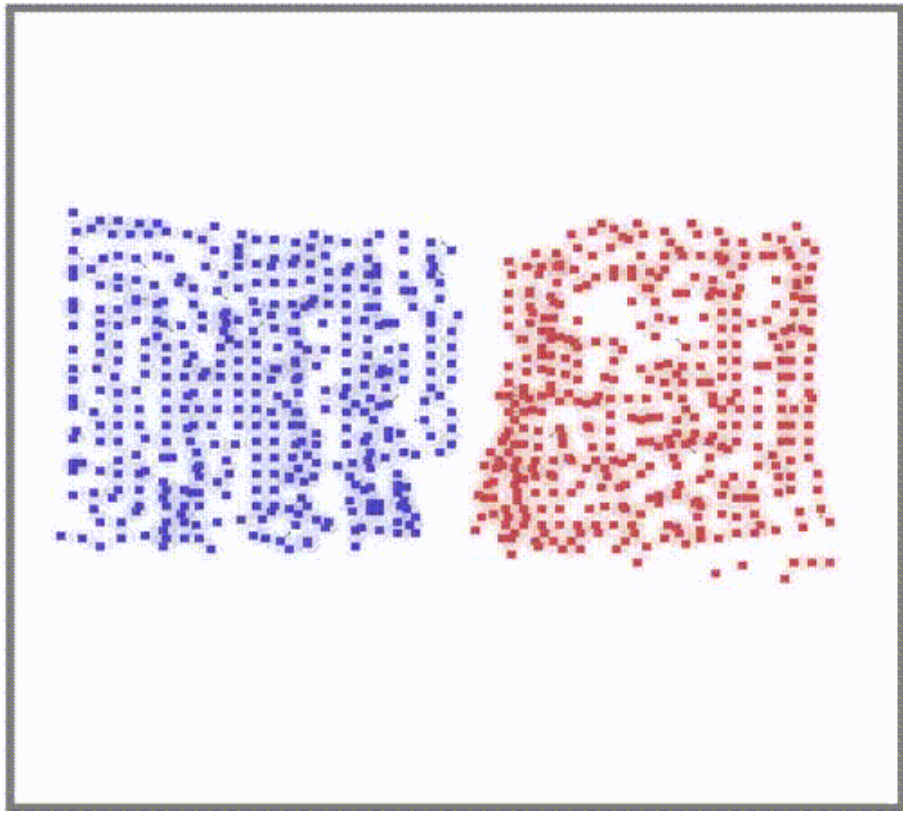


多智能体强化学习(Multi-agent RL)





双智能体对抗与合作



大规模智能体战斗模拟

Leibo, Joel Z., et al. Multi-agent Reinforcement Learning in Sequential Social Dilemmas. AAMAS 2017.
Lianmin Zheng, et al. Magent: a many-agent reinforcement learning platform for artificial collective intelligence. NIPS17 & AAAI18.

强化学习的落地场景

- 无人驾驶
- 游戏AI
- 交通灯调度
- 网约车派单
- 组合优化
- 推荐搜索系统
- 数据中心节能优化
- 对话系统
- 机器人控制
- 路由选路
- 工业互联网场景
- ...

总结强化学习技术发展与落地挑战

强化学习做什么

- 序列型决策任务
- 让AI做完一切事情，而不仅仅是一个辅助的角色

强化学习的技术发展

- 2013年12月的NIPS workshop论文开启了深度强化学习时代
- 目前深度强化学习方法已经可以解决部分序列决策任务，但距离真正普及还有很长的路要走

强化学习的落地挑战

- 决策权力交给AI，人对AI有更高的要求
- 强化学习技术人才短缺，决策场景千变万化，并不统一
- 当前强化学习算法对数据和算力的极大需求

课程大纲

强化学习基础部分

1. 强化学习、探索与利用
2. MDP和动态规划
3. 值函数估计
4. 无模型控制方法
5. 参数化的值函数和策略
6. 规划与学习
7. 深度强化学习价值方法
8. 深度强化学习策略方法

强化学习前沿部分

9. 基于模型的深度强化学习
10. 离线强化学习
11. 模仿学习
12. 参数化动作空间
13. 多智能体强化学习基础
14. 多智能体强化学习前沿
15. 强化学习的应用
16. 技术交流与回顾

探索与利用

讲师：张伟楠 – 上海交通大学

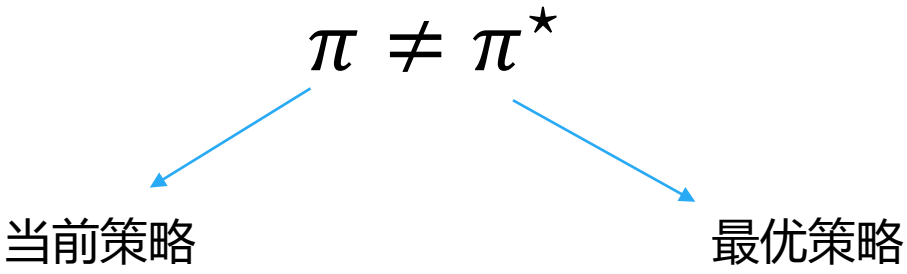
目录

Contents


01
探索与利用

序列决策任务中的一个基本问题

- 基于目前策略获取已知最优收益还是尝试不同的决策
  - **Exploitation** 执行能够获得已知最优收益的决策
  - **Exploration** 尝试更多可能的决策，不一定会是最优收益



序列决策任务中的一个基本问题

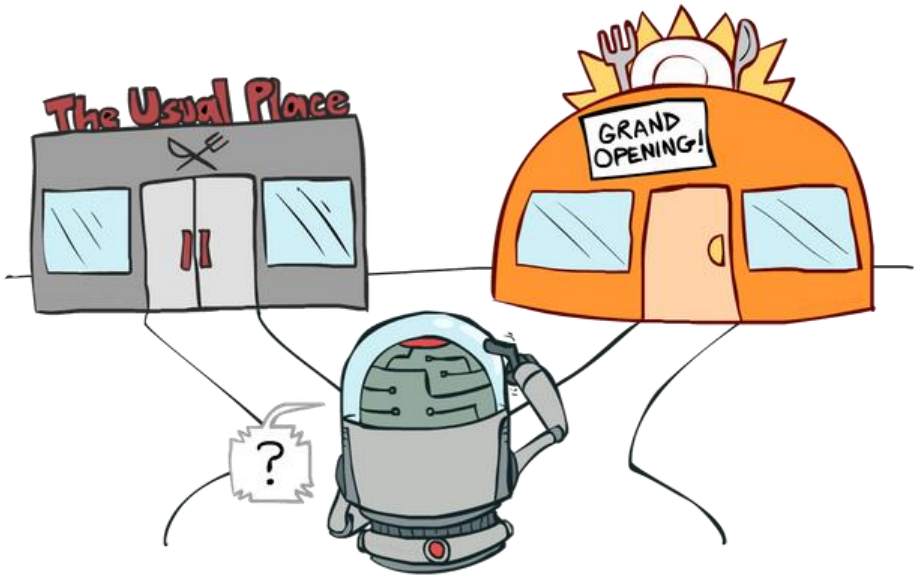
- 基于目前策略获取已知最优收益还是尝试不同的决策
  - **Exploitation** 执行能够获得已知最优收益的决策
  - **Exploration** 尝试更多可能的决策，不一定会是最优收益

$$\mathcal{E}_t = \{\pi_t^i \mid i = 1, \dots, n\} \xrightarrow{\text{探索}} \mathcal{E}_{t+1} = \{\pi_t^i \mid i = 1, \dots, n\} \cup \{\pi_e^j \mid j = 1, \dots, m\}$$

$$\exists\, V^\star(\cdot \mid \pi_t^i \sim \mathcal{E}_t) \le V^\star(\cdot \mid \pi_{t+1}^i \sim \mathcal{E}_{t+1}) \qquad \pi_{t+1}^i \sim \{\pi_e^i \mid i = 1, \dots, m\}$$

探索：可能发现更好的策略

一个例子



图片引用：https://steemit.com/technology/@mor/machine-learning-series-part-5-exploration-vs-exploitation-dilemma-in-reinforcement-learning

策略探索的一些原则

- 朴素方法（Naïve Exploration）
  - 添加策略噪声 ϵ-greedy
- 积极初始化（Optimistic Initialization）
- 基于不确定性的度量 （Uncertainty Measurement）
  - 尝试具有不确定收益的策略，可能带来更高的收益
- 概率匹配（Probability Matching）
  - 基于概率选择最佳策略
- 状态搜索（State Searching）
  - 探索后续状态可能带来更高收益的策略

02
多臂老虎机

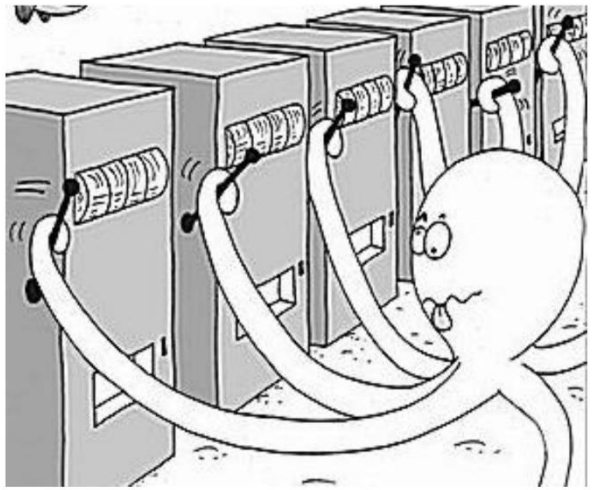
多臂老虎机

- 多臂老虎机（multi-arm bandit）问题的形式化描述

动作集合：$a^i \in \mathcal{A}, i = 1, \ldots, K$

$\langle \mathcal{A}, \mathcal{R} \rangle$

收益（反馈）函数分布：$\mathcal{R}(r \mid a^i) = \mathbb{P}(r \mid a^i)$



- 最大化累积时间的收益：$\max \sum_{t=1}^{T} r_t, r_t \sim \mathcal{R}(\cdot \mid a_t)$

问题 不确定的反馈函数，如何估计？

收益估计

- 期望收益和采样次数的关系

$$Q_n(a^i) = \frac{r_1 + r_2 + \cdots + r_{n-1}}{n-1}$$

- 缺点：每次更新的空间复杂度是 $O(n)$

增量实现

误差项：Δ_n^i

$$Q_{n+1}(a^i) := \frac{1}{n}\sum_{i=1}^{n} r_i = \frac{1}{n}\left(r_n + \frac{n-1}{n-1}\sum_{i=1}^{n-1} r_i\right) = \frac{1}{n}r_n + \frac{n-1}{n}Q_n = Q_n(a^i) + \frac{1}{n}(r_n - Q_n)$$

空间复杂度为 $O(1)$

算法：多臂老虎机

I. 初始化：$Q(a^i) \coloneqq c^i, N(a^i) = 0, i = 1, \dots, n$

II. 主循环 $t = 1:T$

1. 利用策略 π 选取某个动作 a
2. 获取收益：$r_t = Bandit(a)$
3. 更新计数器：$N(a) \coloneqq N(a) + 1$
4. 更新估值：$Q(a) \coloneqq Q(a) + \frac{1}{N(a)}[r_t - Q(a)]$

权衡探索与利用

应当选取什么样的策略 π？

Regret函数

- 决策的期望收益：$Q(a^i) = \mathbb{E}_{r\sim\mathbb{P}(r|a^i)}[r]$
- 最优收益：$Q^\star = \max_{a^i\in\mathcal{A}} Q(a^i)$

Regret

- 决策与最优决策的收益差：$R(a^i) = Q^\star - Q(a^i)$
- Total Regret 函数：$\sigma_R = \mathbb{E}_{a\sim\pi}\left[\sum_{t=1}^{T} R(a_t^i)\right]$

等价性

- $\min \sigma_R = \max \mathbb{E}_{a\sim\pi}\left[\sum_{t=1}^{T} Q(a_t^i)\right]$

随着时间推移，单步regret越来越小
那么探索一直都是必须的吗？

Regret函数

- 如果一直探索新决策：$\sigma_R \propto T \cdot R$，total regret 将线性递增，无法收敛
- 如果一直不探索新决策：$\sigma_R \propto T \cdot R$，total regret 仍将线性递增

是否存在一个方法具有次线性（sublinear）收敛保证的 regret?

下界（Lai & Robbins）

- 使用 $\Delta_a = Q^\star - Q(a)$ 和反馈函数分布相似性：$D_{KL}\big(\mathcal{R}(r \mid a) \parallel \mathcal{R}^\star(r \mid a)\big)$ 描述

$$\lim_{T\to\infty} \sigma_R \geq \log T \sum_{a|\Delta_a>0} \frac{\Delta_a}{D_{KL}\big(\mathcal{R}(r \mid a) \parallel \mathcal{R}^\star(r \mid a)\big)}$$

Lai, Tze Leung, and Herbert Robbins. "Asymptotically efficient adaptive allocation rules." *Advances in applied mathematics* 6.1 (1985): 4-22.

贪心策略和 ϵ-greedy 策略

- 贪心策略

$$Q(a^i) = \frac{1}{N(a^i)} \sum_{t=1}^{T} r_t \cdot 1(a_t = a^i)$$

$$a^\star = \arg\max_{a^i} Q(a^i)$$

$$\sigma_R \propto T \cdot [Q(a^i) - Q^\star]$$

线性增长的 Total regret

- ϵ-greedy 策略

$$a_t = \begin{cases} \arg\max_a \hat{Q}(a) & \text{采样概率：} 1-\epsilon \\ U(0, |\mathcal{A}|) & \text{采样概率：} \epsilon \end{cases}$$

常量 ϵ 保证 total regret 满足

$$\sigma_R \geq \frac{\epsilon}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \Delta_a$$

Total regret 仍然是线性递增的，
只是增长率比贪心策略小

衰减贪心策略

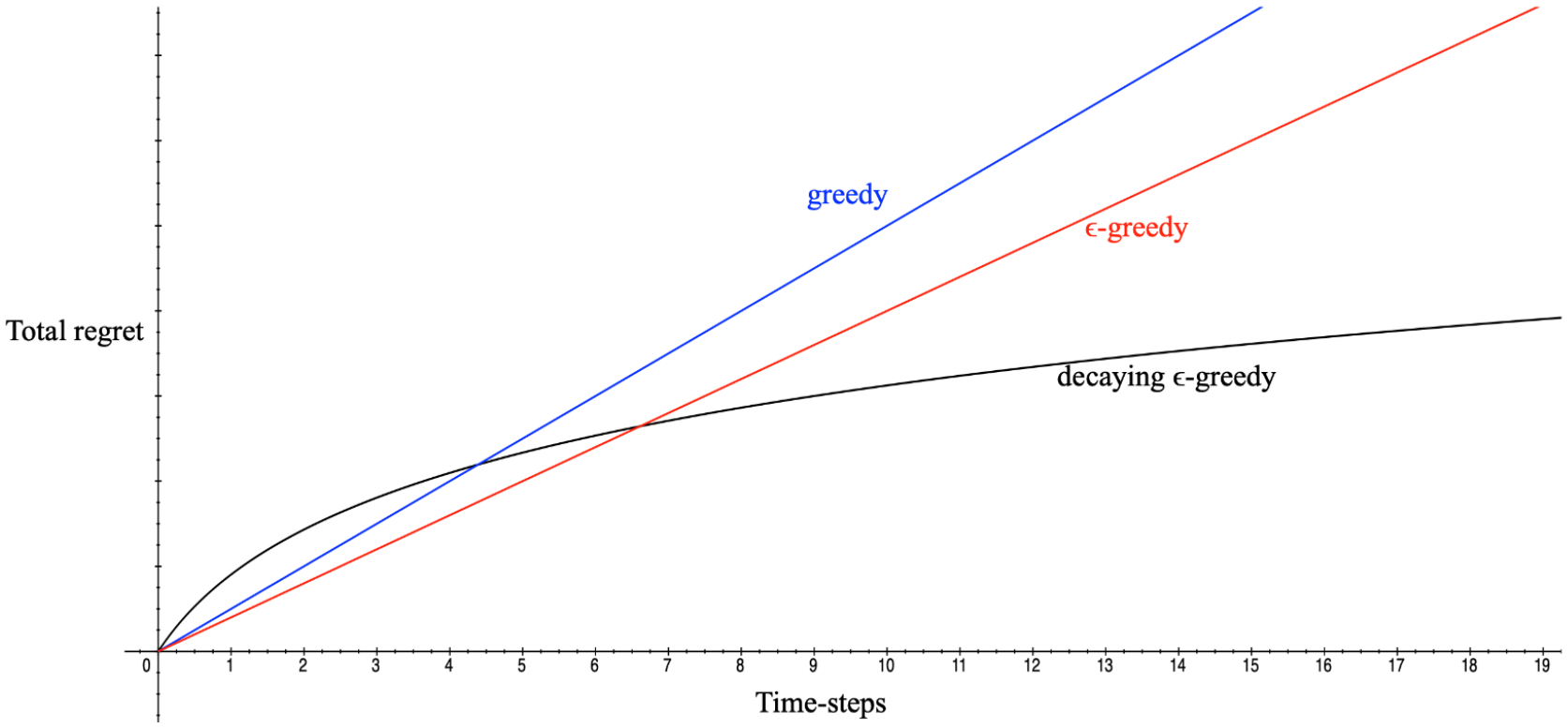
- ϵ-greedy 的变种，ϵ 随着时间衰减
- 理论上对数渐进收敛!
- 一种可能的衰减方式：

$$c \geq 0, \qquad d = \min_{a|\Delta_a>0} \Delta_a, \qquad \epsilon_t = \min\left\{1, \frac{c|\mathcal{A}|}{d^2 t}\right\}$$

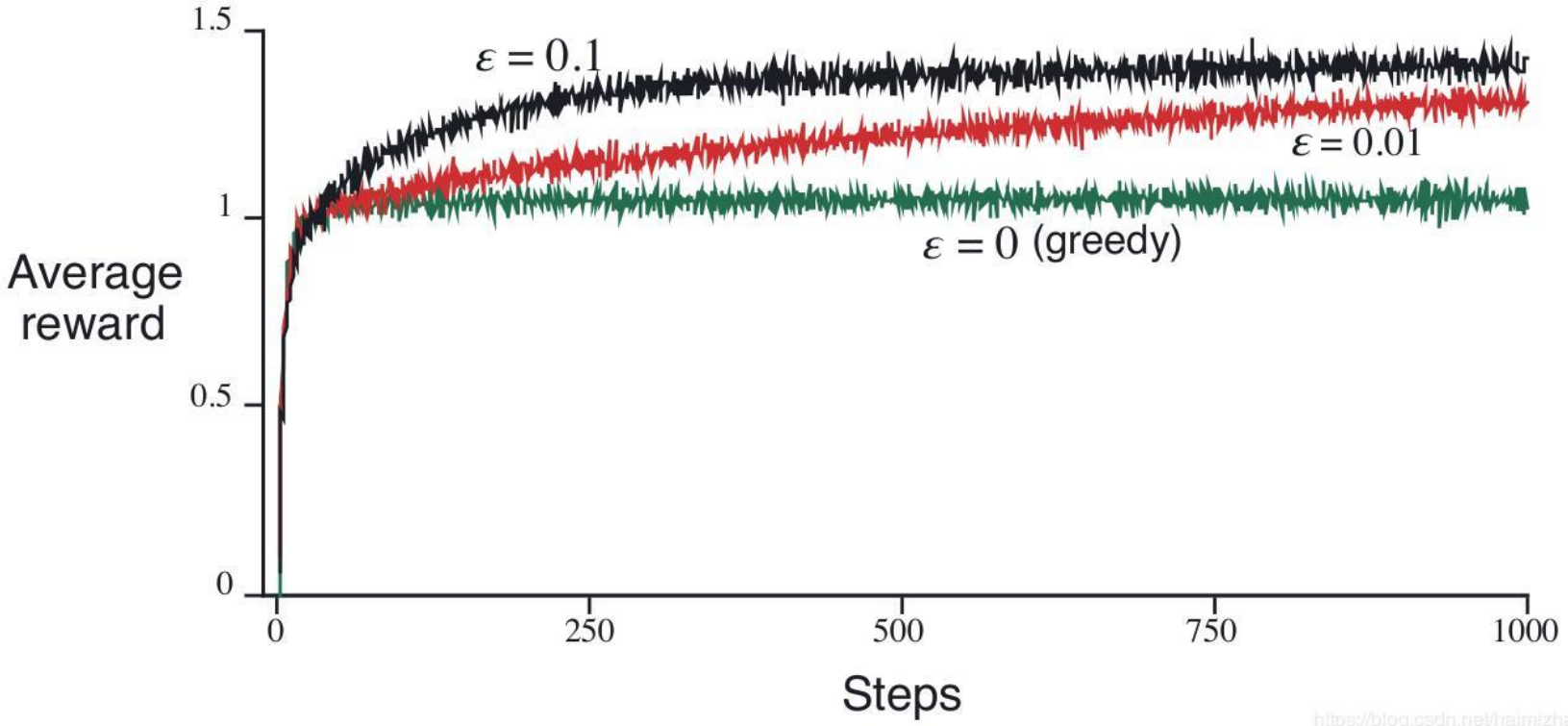
缺点

- 很难找到合适的衰减规划

不同 ϵ-greedy 策略对比：Total Regret



不同 ϵ-greedy 策略对比：平均收益



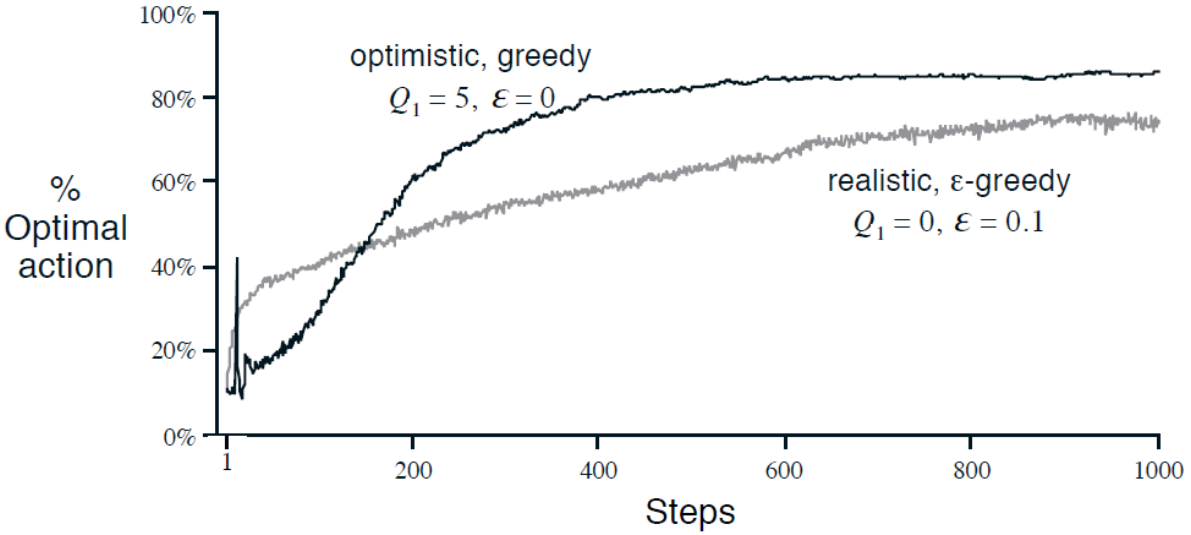
https://blog.csdn.net/haimizhao

乐观初始化

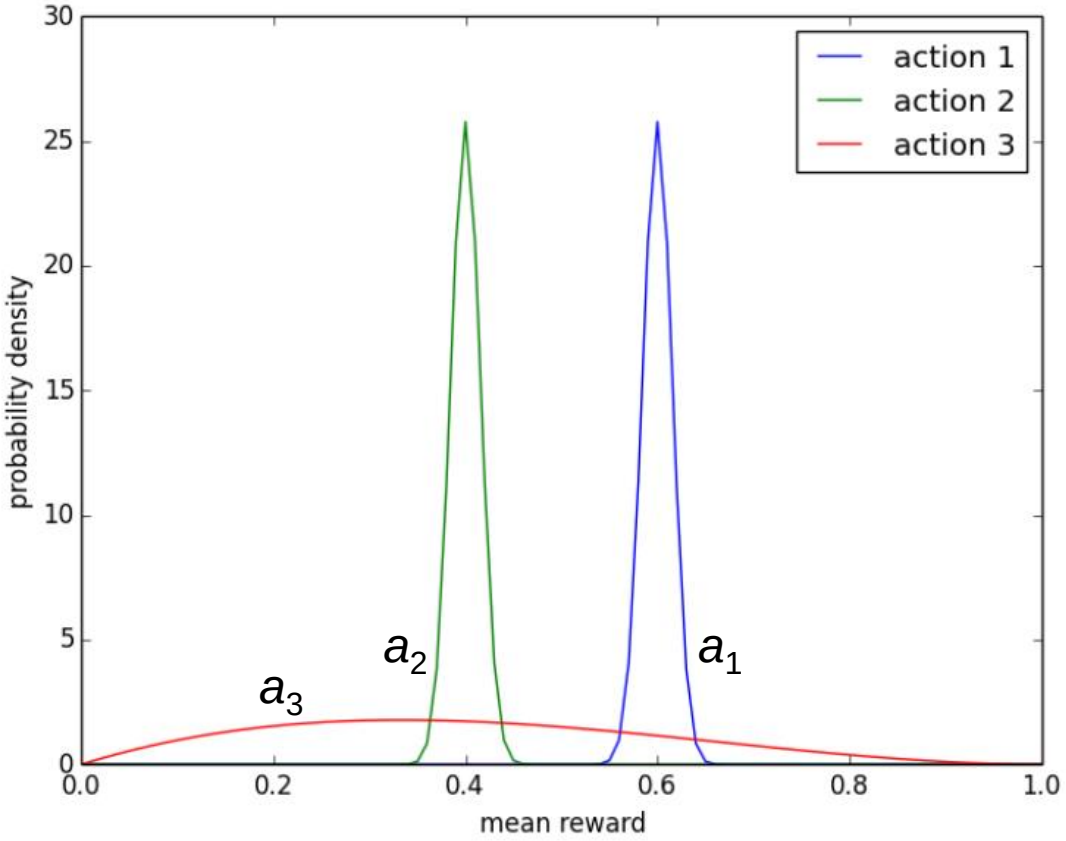
- 给 $Q(a^i)$ 一个较高的初始化值
- 增量式蒙特卡洛估计更新 $Q(a^i)$

$$Q(a^i) := Q(a^i) + \frac{1}{N(a^i)}\left(r_t - Q(a^i)\right)$$

- 有偏估计，但是随着采样增加，这个偏差带来的影响会越来越小
- 但是仍然可能陷入局部最优



显式地考虑动作的价值分布



- 考虑以上三个动作的价值分布，平衡探索和利用，选择哪个动作?
- 1. 显式地鼓励不确定性； 2. 直接根据分布采样来选择

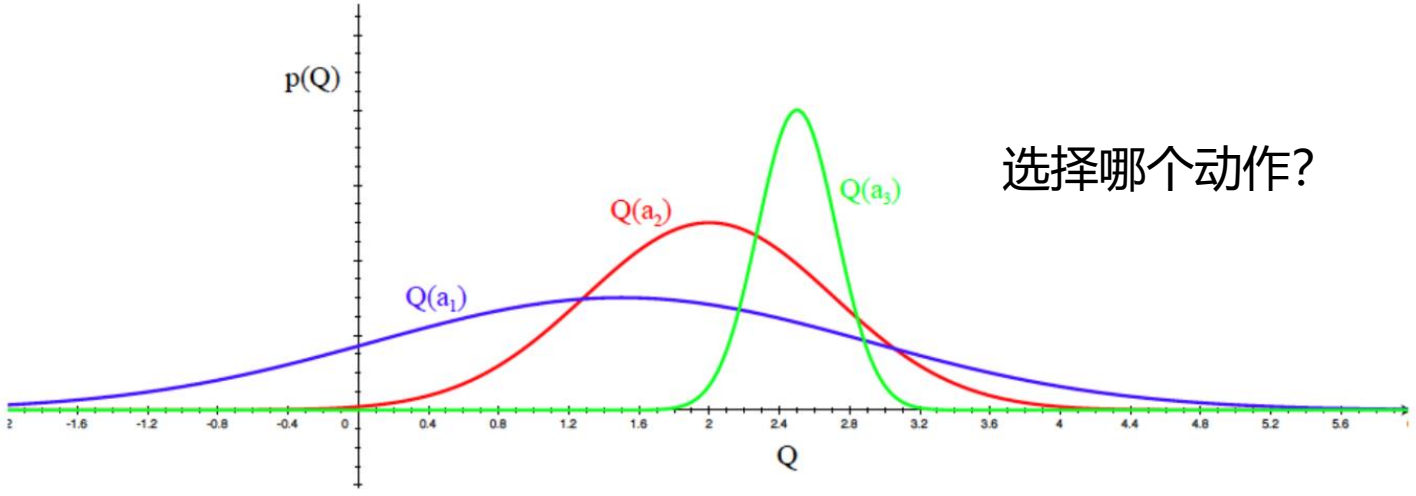
https://web.stanford.edu/~bvr/pubs/TS_Tutorial.pdf

基于不确定性测度

- 不确定性越大的 $Q(a^i)$，越具有探索的价值，有可能会是最好的策略
- 一个经验性指导：
  - $N(a)$ 大，$U(a)$ 小
  - $N(a)$ 小，$U(a)$ 大

策略 π：$a = \arg\max_{a \in \mathcal{A}} \hat{Q}(a) + \hat{U}(a)$

也称为 UCB：上置信界法（Upper Confidence Bounds）



选择哪个动作?

UCB：上置信界

Hoeffding 不等式：$\mathbb{P}[\mathbb{E}[x] > \bar{x}_t + u] \leq e^{-2tu^2}$ for $x \in [0,1]$

- 为每个动作收益估值估计一个上置信界：$\hat{U}(a)$
- 显然有：$Q_t(a) \leq \hat{Q}_t(a) + \hat{U}_t(a)$以高概率$p$成立（Hoeffding不等式）
- 依据以下原则挑选进行决策：

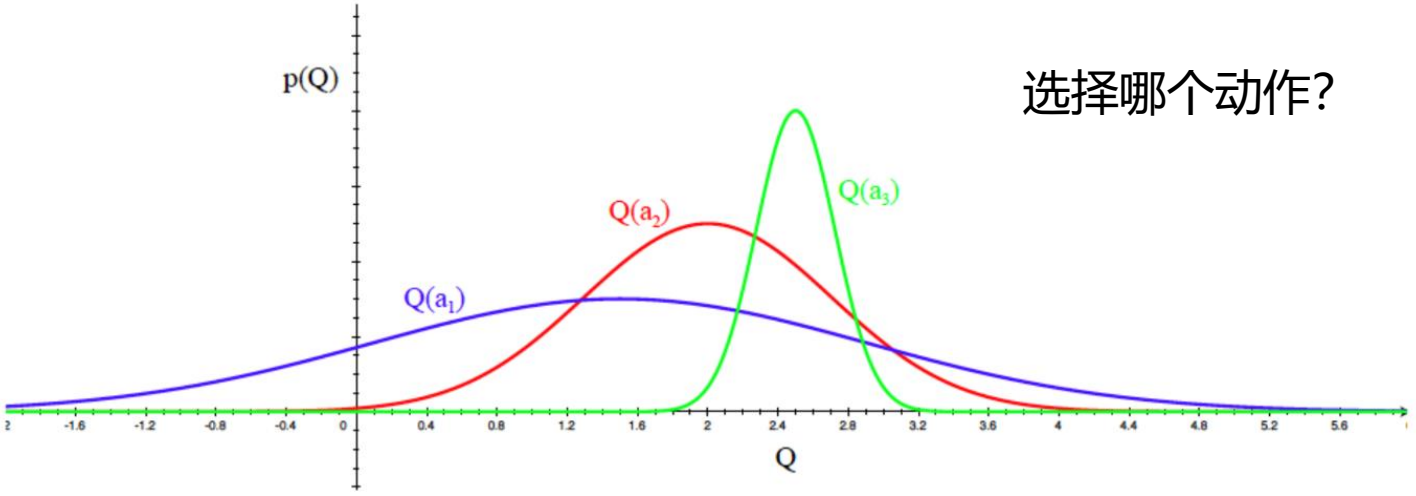
$$a = \arg\max_{a \in \mathcal{A}} \hat{Q}_t(a) + \hat{U}_t(a) \qquad e^{-2N_t(a)U_t(a)^2} = p \Rightarrow \hat{U}_t(a) = \sqrt{-\frac{\log p}{2N_t(a)}}$$

- 收敛性：

$$\lim_{t \to \infty} \sigma_R \leq 8 \log t \sum_{a|\Delta_a > 0} \Delta_a$$

Hoeffding, Wassily. "Probability inequalities for sums of bounded random variables." *The Collected Works of Wassily Hoeffding*. Springer, New York, NY, 1994. 409-426.

Thompson Sampling方法



选择哪个动作?

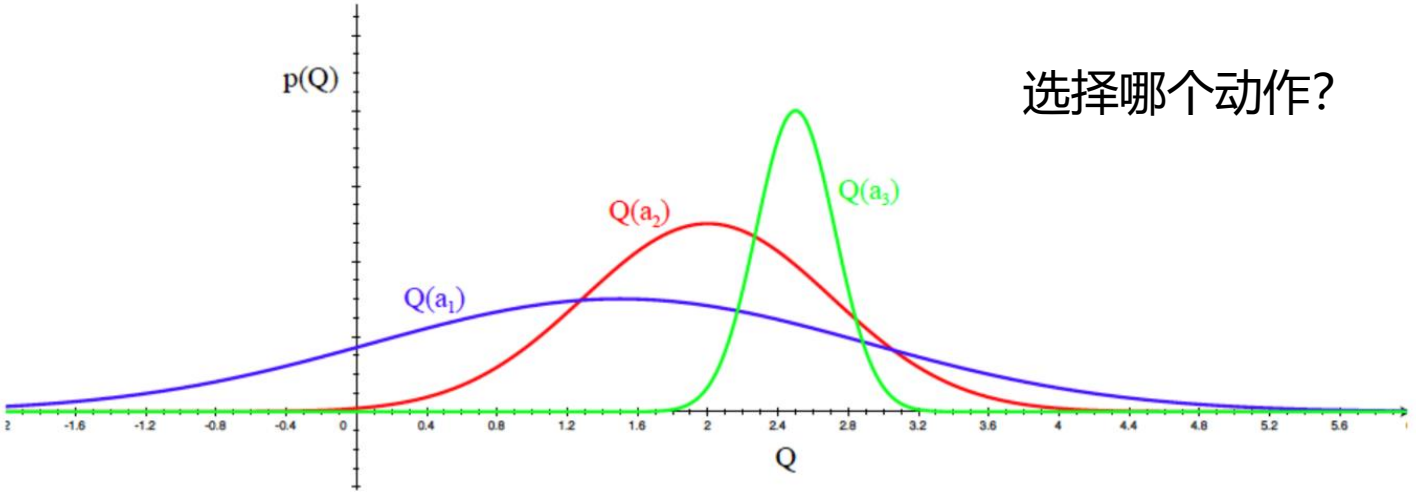
- 想法：根据每个动作成为最优的概率来选择动作

$$p(a) = \int \mathbb{I}\left[\mathbb{E}_{p(Q(a))}\left[Q(a;\theta)\right] = \max_{a' \in \mathcal{A}} \mathbb{E}_{p(Q(a'))}\left(Q(a';\theta)\right)\right] d\theta$$

- 实现：根据当前每个动作a的价值概率分布$p(Q(a))$来采样到其价值$Q(a)$，选择价值最大的动作

Chapelle, Olivier, and Lihong Li. "An empirical evaluation of thompson sampling." Advances in neural information processing systems 24 (2011): 2249-2257.

Thompson Sampling方法



选择哪个动作?

- 实现：根据当前每个动作a的价值概率分布$p(Q(a))$来采样到其价值$Q(a)$，选择价值最大的动作

Algorithm 1 Thompson sampling

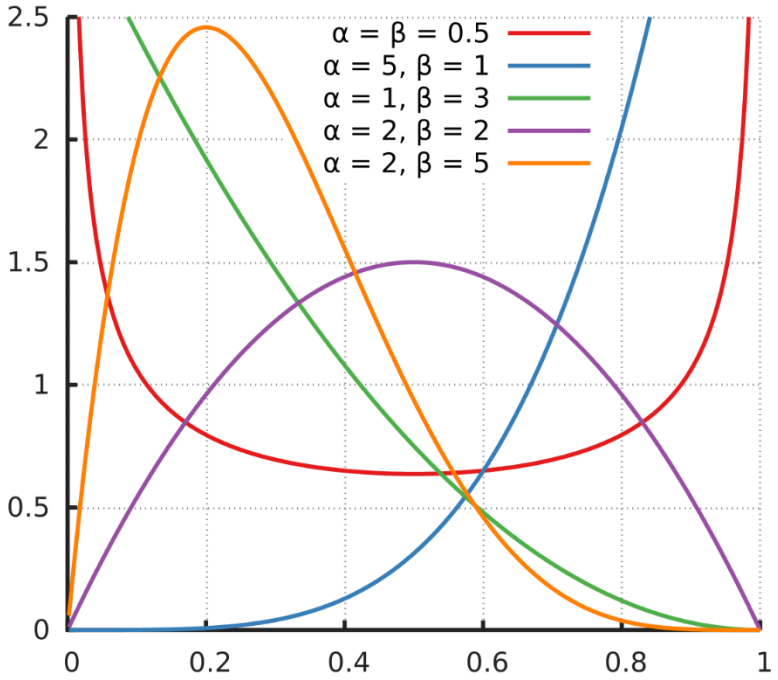
$D = \emptyset$
for $t = 1, \ldots, T$ **do**
  Receive context x_t
  Draw θ^t according to $P(\theta|D)$
  Select $a_t = \arg\max_a \mathbb{E}_r(r|x_t, a, \theta^t)$
  Observe reward r_t
  $D = D \cup (x_t, a_t, r_t)$
end for

Chapelle, Olivier, and Lihong Li. "An empirical evaluation of thompson sampling." *Advances in neural information processing systems*. 2011.

Thompson Sampling和UCB的实验对比

- 实验：K-arm多臂老虎机，用Beta分布建模每个arm的成功率，初始化成功率分布为Beta(1,1).

$$Beta(x;\alpha,\beta) \propto x^{\alpha-1}\ (1-x)^{\beta-1}$$



Algorithm 2 Thompson sampling for the Bernoulli bandit

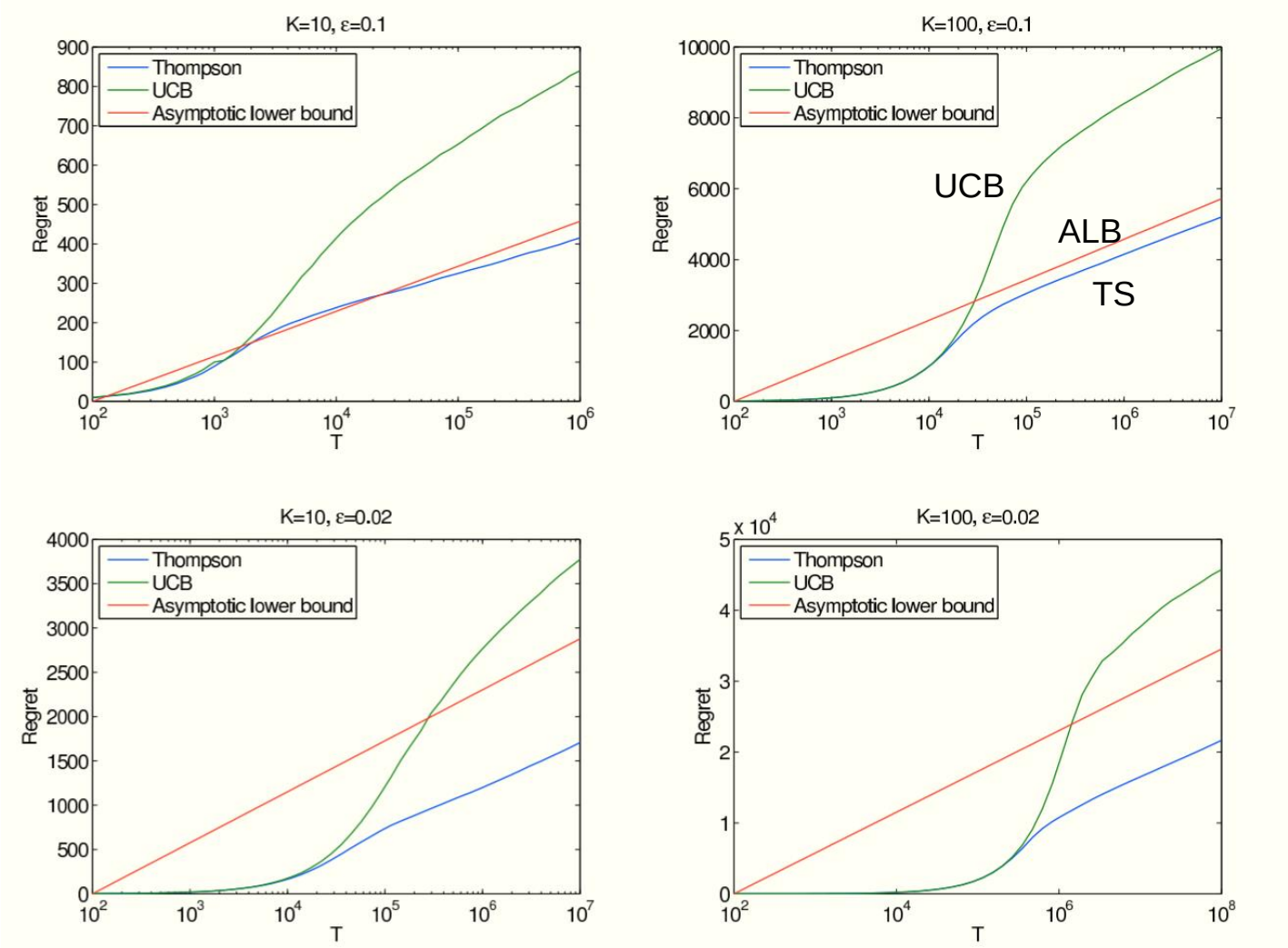
Require: α, β prior parameters of a Beta distribution
$S_i = 0, F_i = 0, \ \forall i$. {Success and failure counters}
for $t = 1, \ldots, T$ **do**
  for $i = 1, \ldots, K$ **do**
    Draw θ_i according to Beta$(S_i + \alpha, F_i + \beta)$.
  end for
  Draw arm $\hat{\imath} = \arg\max_i \theta_i$ and observe reward r
  if $r = 1$ **then**
    $S_{\hat{\imath}} = S_{\hat{\imath}} + 1$
  else
    $F_{\hat{\imath}} = F_{\hat{\imath}} + 1$
  end if
end for

Chapelle, Olivier, and Lihong Li. "An empirical evaluation of thompson sampling." *Advances in neural information processing systems*. 2011.

Beta distribution: https://towardsdatascience.com/beta-distribution-intuition-examples-and-derivation-cf00f4db57af

Thompson Sampling和UCB的实验对比

- K-arms
- The best arm has reward probability of 0.5
- K-1 other arms have the reward probability of 0.5-ε
- Asymptotic lower bound: (Lai & Robbins)



$$\lim_{T\to\infty} \sigma_R \geq \log T \sum_{a|\Delta_a>0} \frac{\Delta_a}{D_{KL}\left(\mathcal{R}(r \mid a) \parallel \mathcal{R}^{\star}(r \mid a)\right)}$$

Chapelle, Olivier, and Lihong Li. "An empirical evaluation of thompson sampling." *Advances in neural information processing systems*. 2011.

探索与利用总结

- 探索与利用是强化学习的trial-and-error中的必备技术
- 多臂老虎机可以被看成是无状态(state-less)强化学习
- 多臂老虎机是研究探索与利用技术理论的最佳环境
  - 理论的渐近最优regret为 $O(\log T)$
- ε-greedy、UCB和Thompson Sampling方法在多臂老虎机任务中十分常用，在强化学习的探索中用也十分常用，最常见的是ε-greedy

THANK YOU