强化学习奖励函数塑形简介（The reward shaping of RL）

有道理
机器学习

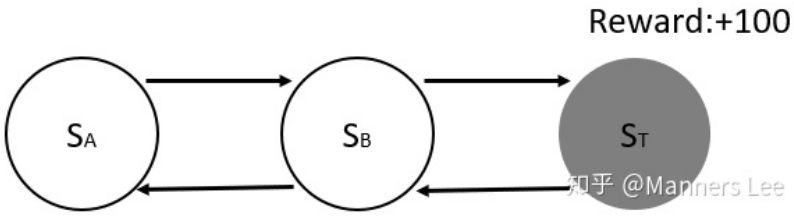
1. RL背景

强化学习解决定义在马尔科夫过程（Makov Decision Processing, MDP）下的连续决策问题。其中经典算法Q-learning使用如下方程更新 $Q^{\pi}(s,a)$ 值：策略 π 在状态s下采取行为a后的累计回报数学期望（Cumulated reward）.

$$Q(s,a) \leftarrow Q(s,a) + \alpha(r + \max_{a'} Q(s',a') - Q(s,a))$$

2. RL面临的挑战： 奖励稀疏性（sparse reward ）

大部分任务的state-action空间中，奖励信号都为0. 我们称之为奖励函数的稀疏（sparsity of reward）。 稀疏的奖励函数，导致算法收敛缓慢。 Agent需要和环境多次交互采并学习大量样本才能，收敛到最优解.



如上图MDP， Agent 从状态 s_A 出发到 s_T 结束并获得奖励+100. 在第一轮学习中， agent使用等概率探索策略， 能够到达目标获得奖励的概率为1/4。如果MDP的状态-行为空间更大更大，agent首次到达目标的概率非常低。 非终点状态奖励都为零。Agent首次到达目标概率：

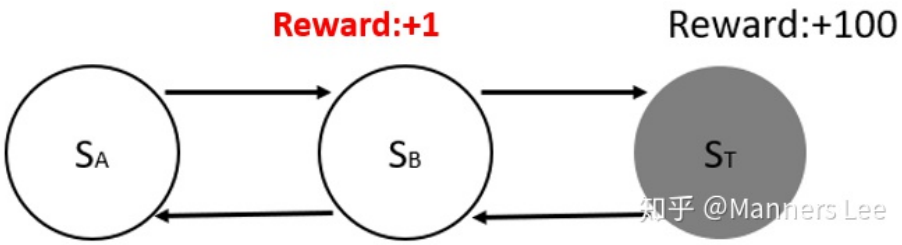
$$P = \frac{1}{|A|^M}$$

其中 M为到达目标步数，|A| agent可以采用的为行为数量

3. 解决方案： 函数塑形（reward shaping）

3.1 直觉解决方案： 额外奖励法

一个直觉的方法解决奖励稀疏性问题是当agent向目标迈进一步时，给于agent 回报函数（reward）之外的奖励。 R'(s,a,s') = R(s,a,s')+F(s'). 其中R'(s,a,s') 是改变后的新回报函数。 这个过程称之为函数塑形（reward shaping）。



3.2 改变Reward可能改变问题的最优解。

比如上图MDP的最优解方案是在 s_A 和 s_B 中间来回走动，不停的得到+1的奖励。 俗称“刷

改变奖励函数导致

3.3 势能函数，解决“刷分”

势能函数，记为 $\Phi(s)$ 定义了一个状态的势能. 这个概念借鉴了物理学的势能概念。 当agent从一个高势能状态转移到第低势能转移时，它将获得额外地奖励（类似物理中的动力）。 反过来，如果agent从低势能状态到高势能状态，它将失去奖励（得到一个和上面相同但负的奖励）。 这个机制和物理中的能量守恒类似。

使用两个状态的势能差值作为额外奖励可以保证不改变MDP的最优解。 详细证明参考[1].

$$R'(s,a,s') = R(s,a,s') + F(s')$$

$F(s') = \Phi(s') - \gamma\Phi(s)$ 其中 γ 是折扣因子

类似Reward function势能函数F也可以定义为F(s,a)和F(s,a,s')。

[1] 证明势能函数的本质，就是Q函数的初始化状态。 一般我们使用全0 Q-value作为初始Q-value，通过更新Q-value最终收敛到最优值. 势能函数改变Q-value的初始状态。好的势能函数一定程度上接近最优Q-value， 可以减少一些早期学习从而加速学习过程。 试想一下，如果势能函数完全等于最优Q-value， 当前状态仅仅学习一步满足Bellman方程。

$Q(s,a) = R(s) + F(s) + Q^{Init}(s',a')$ 其中 $Q^{Init}(s',a')$ 为全0 Q-value

4. 应用举例

函数塑形可以作为一个插入人类知识的入口。 在许多任务中， 人类总结了大量的次优（大概对，不是100%对， 基本上符合经验)启发式的规则。 记为 $a_h = \Phi(s)$,其中 a_h 是根据启发规则在状态s下得到的启发行为。 可以用agent所选择action是否和启发规则一致作为势能函数。 如果agent的行动<s,a>和启发规则一致，则认为该<s,a>具体高势能。 参考文献[3] 使用和人类示例数据的高斯距离作为势能函数，加快RL学习同事保险证收敛到最优解。

5. 结论

为了解决基于值函数的RL在稀疏奖励空间学习慢的问题， 当agent靠近最终目标或者完成一个子任务时，将被给予额外的奖励。 然而改变奖励函数可能改变MDP的最优解。 使用势能函数差作为塑形函数可以保证不改变MDP的最优解。 定义势能函数的时，插入人对任务的领域知识，增加状态-行为空间奖励信号，有助于加快RL agent学习。

6. 参考文献

[1] Ng, Andrew Y., Daishi Harada, and Stuart Russell. "Policy invariance under reward transformations: Theory and application to reward shaping."ICML. Vol. 99. 1999.

[2] Wiewiora, Eric. "Potential-based shaping and Q-value initialization are equivalent." Journal of Artificial Intelligence Research 19 (2003): 205-208.

[3] Brys, Tim, et al. "Reinforcement Learning from Demonstration through Shaping."IJCAI. 2015.

编辑于 2019-02-13

强化学习 (Reinforcement Learning)　机器学习　agent-based model

文章被以下专栏收录

推荐阅读

强化学习基础篇: 价值迭代 (Value Iteration)

冯伟

《强化学习》第一讲 简介

本讲是对于强化学习整体的一个简单介绍，描述了强化学习是什么，解决什么问题，大概用什么样的方式来解决问题。介绍了强化学习中常用的概念。这些概念非常重要，贯穿于整个强化学习始终，但...

叶强

多智能体 ——M

ECKai

9 条评论 ⇌ 切换为时间排序

评论由作者筛选后显示

Faker 2019-07-02

您好 本人刚刚入门强化学习 现在不是很明白reword的数值是自己来设计从而计算Q还是能通过样本估计出来

赞

有道理 (作者) 回复 Faker 2019-07-03

Reward是人设计出来的。比如说，设计麻将agent，专门给领导点炮。是不是应该改改reward从新训练一下?

1

有道理 (作者) 回复 Faker 2019-08-03

Reward 函数是人设计的。 Q是计算出来的。比方说，你叫奖励agent 输牌给领导

赞

陈与论 07-23

势能函数这个词不太好，翻译成潜能函数会直观一些

赞

有道理 (作者) 回复 陈与论 07-23

据说，这个idea来至于物理上的，机械能守恒。 有点在Q函数上，高低跳动，动能-势能守恒的那个意思。

赞

一木一土 09-14

有个地方请教您一下，如果自行设计的回报不收敛，该怎么办？另外不收敛多少个epochs就能判定此刻的设计是false的?

赞

有道理 (作者) 回复 一木一土 09-14

1，你指的应该是不收敛到你希望的policy。尽量保持reward 简单。2，我一般可视化q值或者当然策略，人为经验调剂。同上，我总是选择最简化的reward

赞

有道理 (作者) 09-14

经典RL是没有epoch的。episode 累积reward不收敛？上下震动正常的，1 exploit exploration 带来随机性，2 环境的随机性(转移函数概率)，3有些文献定义reward 函数有随机性，这个不常见。 三个因素乘起来是单步的分布，所步骤乘起来是episode 的分布。再看episode 的reward 方差常常就很大。

赞

有道理 (作者) 09-14

DQN memory reply的收敛性目前没有证明。大家都是靠多train 几次，拿到最好的那个。这一点一直收到争议，也是很多人转向policy gradient method 的原因。