

Abstract

这是一个悲伤的故事，Q-learning它病了，病的不轻，这也是最近才发现的事，在特定的条件下，Q-learning就会对动作值进行过量估计。以前没人在意这事，但其实这种病是普遍存在的，我们还不知道这病会不会危及性能，也不知道能否有对应的疫苗，一筹莫展。在这篇文章中，我们首先很正规的回答了这些疑问，然后我们以王者的姿态对使用了Q-learning的DQN进行了严厉批评，毕竟我们的团队现在加了一个爸爸的前缀，我们明确指出DQN当时确实对Atari游戏中的动作值进行了过量估计。我们接下来为大家揭示Double Q-learning算法幕后的秘密，怕太难？不，我们会从离散环境的设定讲起；怕不强？它能够适用于大规模值估计算法。我们提出了对DQN的修改方法，使其能减少观察到的过量估计，并更适合于Atari游戏环境，能取得更好的成绩。

Introduction（好桑心，原文竟然把我忘记了）

1998年Sutton与Barto提出了强化学习的目标是在序列性决策问题中学习优秀策略，方法是通过优化累计奖励信号。于是各种强化学习算法就像葫芦娃一样一个接一个出现了，一个比一个强，Q-learning就是里面最受欢迎的那个葫芦娃。但是表现优秀的年轻人总有一颗高傲的心，Q-learning总喜欢把动作值给学大，究其根源还不是核心算法里有个最大化的步骤，所以时间一长就开始眼高手低了。

以前的研究总把过高的值估计归因于近似程度不够（Thrun and Schwartz，1993）和噪声（van Hasselt，2010, 2011）。本文中我们统一了这些观点，并证明当动作值不准确时也会发生高估现象，近似误差的来源会被完全忽视。然而高估现象是强化学习过程中的常客，所以一切论证都会说明高估现象确实比以前看起来更容易发生了。

一旦过量估计发生了，那么实际上已经对AI造成了性能上的影响，目前这仍然是未能解决的问题。对值的过量估计本身其实并不是什么问题，毕竟如果所有动作值都比较高，这时候具有高相关性的动作就会被保留下来，并且我们也不会看到训练出来的策略越来越差。此外，大家都知道这种高估方法是有它的好处的：比如面对未知环境时，把值估大一些会获得更好的探索性。但是如果高估不均匀，并且没有集中在我们希望了解的状态分布上，那么就会对最终策略的质量产生负面影响。（Thrun and Schwartz，1993）给出了具体的例子，在他们的实验中高估导致了次优策略。

为了测试在实践中是否具体发生过高估计，我们研究了最近的DQN算法的性能（Mnih等，2015）。DQN将Q-learning与灵活的深度神经网络相结合，并在各种大型Atari 2600游戏中达到了人类专家级别的表现。看起来这种方式是Q-learning的最佳实现方式，因为深度神经网络的确能提供灵活的函数逼近效果，具有低渐近逼近误差的可能性，并且环境的确定性可防止噪声的有害影响。 也许令人惊讶的是，我们表明即使在这个相对有利的环境中，DQN有时也会大大高估行动的价值。

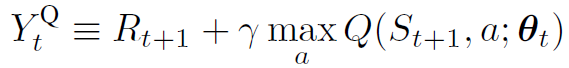
我们会展示一种新的思路，构建双层的新型Q-learning算法。这个想法是基于Hasselt在2010年提出的Double Q-learning 算法，它首次亮相于离散环境的实验，后来又被扩展到函数估计，包含深度神经网络。我们使用这个想法作为根基构建了称为Double DQN的模型。之后通过实验，我们发现这个算法不仅能提供更高速的值估计，又可以让AI在许多游戏中夺取更多的奖励。实验还显示了DQN中的过量估计会导致次优策略的现象，也说明了我们的模型能够在Atari环境超越DQN并取得最优成绩。

Background

为了解决序列决策问题，我们可以让AI为每个动作进行最优估值，这种最优的定义就基于AI根据策略做出动作时得到的平均未来奖励。然后老套的剧情又来了，我不想再把强化学习、Q-learning以及DQN的过程讲一遍了，具体可以参照《1.DQN》







上面3个公式分别是强化学习Q值计算、强化学习参数更新、强化学习目标值计算，而下面的公式就是DQN的目标值计算，我们可以看到仅仅是参数的一个上标变化，以表示其需要经常拷贝。



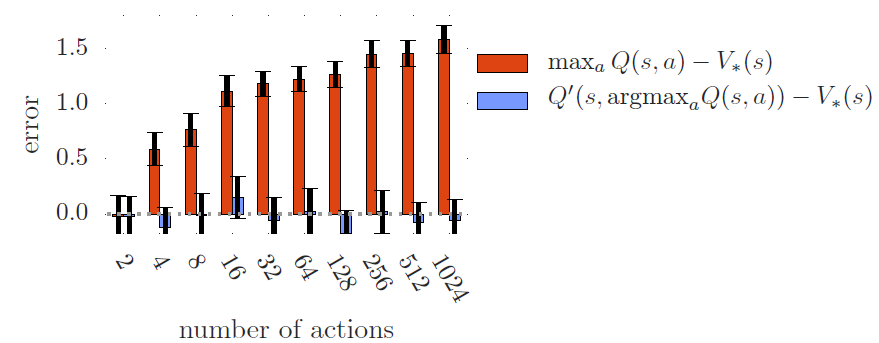
Double Q-learning

在介绍Double DQN之前我们先从Double Q-learning来引入，注意到下式，其中argmax的选择还是根据当前参数 ，然而紧接着又会有一层，以参数 来进行更新外部Q值，相当于以前是一个控制阀来控制算法动力，现在是两个，这样算法自己就能玩出更多的姿态。

**Overoptimism due to estimation errors**

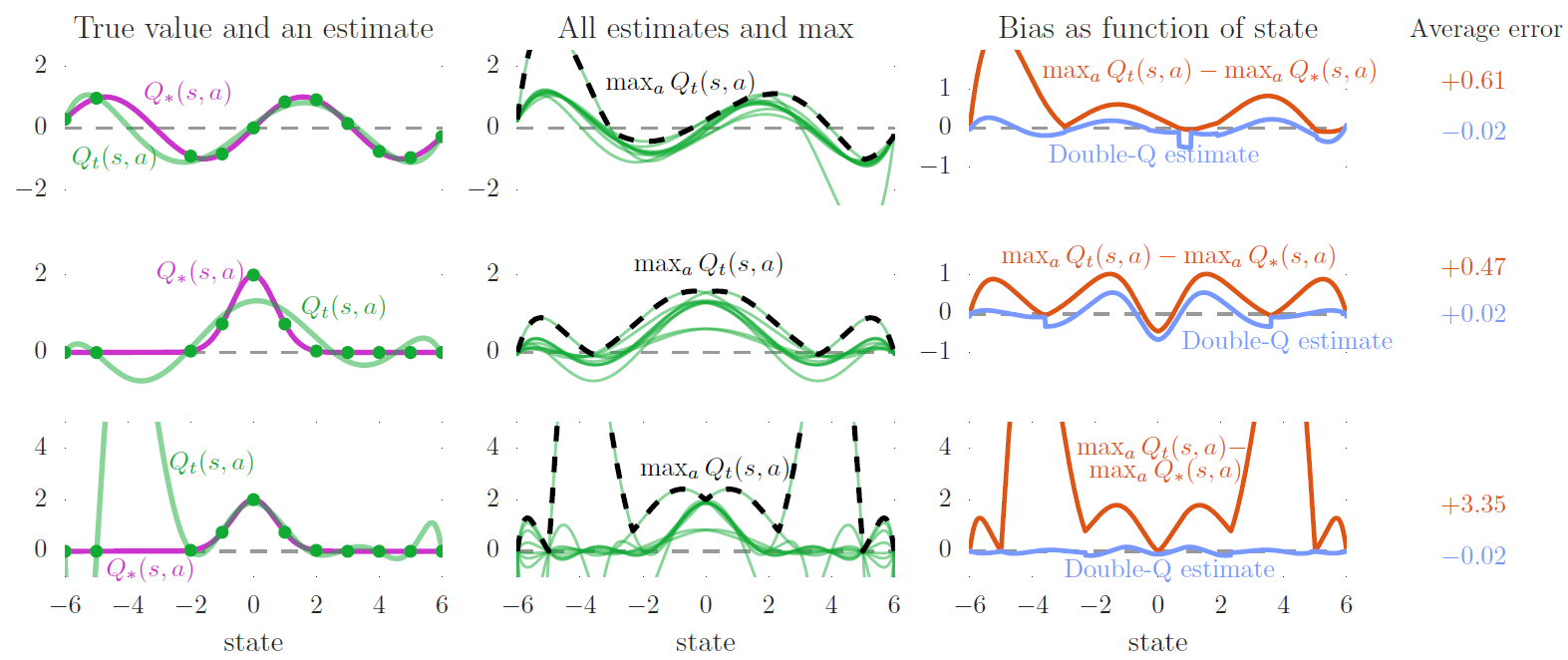
关于Q-learning的过量估计最早记载于1993年Thrun and Schwartz的文章，这篇文章阐述了如果动作值包含一定分布的随机噪声时（分布于），那么每个目标函数都会过量估计至 之上，其中m就是动作数量，具体过程参见原为Theorem 1。他们还设计了一个确定性的实验，证明过量估计在能够把自己淡化为一个难以察觉的问题的基础上，渐进的导致了次优问题，这些在使用函数估计时都会发生。接着van Hasselt在2010年发表了一个论点，即使在离散环境下，只要环境中存在噪声，那么一定会产生过量估计现象，并且他当时介绍了解决方案，就是Double Q-learning。

这种一定会过量的问题看起来仿佛就是有人为其增加了偏置一般，本文也会证明其实任何类型的估计误差都能够造成这种向上的偏置。这些噪声可能来自于环境、函数逼近过程、结构稳定性或其他来源。



*图1：橙色柱状是将动作值增加随机标准分布的误差后所形成的单Q偏置，蓝色则展示了双层Q带来的好处，双层就无法被偏置了*。

接下来我们又做了实验，这次我们弄了一个所有状态均包含10个离散动作的环境，其实验结果及解释如图2。



*图2：我们可以看到state是连续的，观察范围是（-6, 6）的连续值，每个状态都有10个动作。最左边的图显示了紫色是真值而绿色是估值，我们可以看到从上到下分三个图是因为右端描述了三种平均误差的添加方式，当误差添加范围较小时，左图中的估值可以很好的拟合真值，而当误差范围扩大时情况就会发生变化；中图是对比了所有10个动作的估值曲线，我们可以看到黑色线段是最大估值，通常都高于其他曲线；右图则说明蓝色Double-Q估值紧贴0轴，偏置较小，在误差影响下的相对浮动也较小。*

Double DQN

搞明白了Double Q-learning，就很容易说明Double DQN了，首先就是算法上，其实就是将用于估计的目标函数更换了一下,注意 的正确解释是：在贪心策略下计算出来的目标网络权重。而这个版本的Double DQN是最小变化形态，也就是初代战机，发明它的初心就是在利用到Double Q-learning 优势的条件下，尽可能多的保留DQN的完整性。



Others

关于其他内容，包含实验结果、评价方法、人类玩家起步等，都已经在前面的文章中介绍过了，这篇文章就不再做多余的介绍，希望你能看的开心，勇于实践！

