Deterministic Policy Gradient Algorithms（DPG）

作者：David Silver，Guy Lever，Nicolas Heess, Thomas Degris, Daan Wierstra, Martin Riedmiller

单位：DeepMind Technologies, London, UK；University College London, UK

论文发表会议：Proceedings of the 31st International Conference on Machine Learning

论文发表时间：2014

论文地址：https://www.deepmind.com/publications/deterministic-policy-gradient-algorithms

论文贡献：这篇论文提出了确定性的策略梯度算法，是对之前的随机性梯度策略算法的发展。

一、研究动机

在随机策略梯度算法之中，计算其目标函数梯度的期望需要在状态空间和动作空间进行积分，也就是说采样时需要尽可能地覆盖到所有的状态空间和动作空间，所以就需要尽可能多地采样。但是如果策略是一个确定性的策略，那么积分就只需要在状态空间上进行，这样就大大减小了需要采样的数目。

二、随机策略梯度算法

一个基本的setting:

整个过程被定义为一个马尔可夫决策过程（Markov decision process），一个轨迹（状态、动作、奖励）具体如下所示：

$$s_1, a_1, r_1 \ldots, s_T, a_T, r_T$$

一些基本的符号定义:

随机性策略：$\pi_\theta(s)$
状态价值函数: $V^\pi(s)$
动作价值函数：$Q^\pi(s,a)$

随机策略梯度理论（Stochastic Policy Gradient Theorem)

策略的表现目标函数（performance objective）：

$$J(\pi_\theta) = \int_S \rho^\pi(s) \int_A \pi_\theta(s,a) r(s,a) da ds$$

该目标函数的梯度：

$$\nabla_\theta J(\pi_\theta) = \int_S \rho^\pi(s) \int_S \nabla_\theta \pi_\theta(a|s) Q^\pi(s,a) da ds$$

$$= E_{s\sim\rho^\pi, a\sim\pi_\theta}[\nabla_\theta \pi_\theta(a|s) Q^\pi(s,a)]$$

理论内容：把策略参数 θ 向策略梯度的方向调整，以实现最大化目标函数。但是在实际实现的过程中，这里的期望是通过采样轨迹来实现的。同时还要关注的一个问题是这里的动作价值函数$Q^{\pi}(s,a)$ 要如何得到。一个思路是直接用采样中的实际累计奖励 r_t^{γ}来估计 $Q^{\pi}(s_t,a_t)$,这也就是REINFORCE算法。

随机演员-评论员算法（Stochastic Actor-Critic Algorithms)

算法内容：我们称动作价值函数拟合器$Q^{w}(s,a)$为评论员（Critic），策略π_{θ}为演员（Actor），通过交替更新评论员和演员来实现最大化目标函数。更新的过程可以理解为$Q^{w}(s,a)$逐步拟合真实的动作价值函数$Q^{\pi}(s,a)$，策略π_{θ}根据$Q^{w}(s,a)$不断地优化参数θ。

注：在这个过程中会有这样一个问题：我们引入了 $Q^{w}(s,a)$ 来估计真实的 $Q^{\pi}(s,a)$ ，在这个过程中有可能会存在偏差。所以，文中给出两个条件，只要$Q^{w}(s,a)$满足这两个条件，那么就不会引入偏差。这两个条件为（这部分不是很重要，可跳过）：

1. $$Q^{w}(s,a)=\nabla_{\theta}log\pi_{\theta}(a|s)^{T}w$$
2. $Q^{w}(s,a)$的参数w需要满足最小化下面这个均方误差（MSE）：

$$\epsilon^{2}(w)=E_{s\sim\rho^{\pi},a\sim\pi_{\theta}}[(Q^{w}(s,a)-Q^{\pi}(s,a))^{2}]$$

异策略演员-评论员算法（Off-Policy Actor-Critic)

该算法和原始的演员-评论员算法的区别在于用于采样轨迹的策略（behaviour policy: $\beta(a|s)$）和实际更新的策略（target policy: $\pi_{\theta}(a|s)$ ）不一致，即
$\beta(a|s)\neq\pi_{\theta}(a|s)$
在原始的演员-评论员算法中，每次采样来计算目标函数的时候，由于策略已经进行了更新，都需要重新采样，这样的话采样得到的轨迹的利用率就很低。所以考虑采样和实际更新的策略采用不同的策略，这样就可以提高采样的利用率。具体推导如下：
该算法中的目标函数：

$$J_{\beta}(\pi_{\theta})=\int_{S}\rho^{\beta}(s)V^{\pi}(s)ds$$

$$=\int_{S}\int_{A}\rho^{\beta}(s)\pi_{\theta}(a|s)Q^{\pi}(s,a)dads$$

该算法中的目标函数的梯度：

$$\nabla_{\theta}J_{\beta}(\pi_{\theta})\approx\int_{S}\int_{A}\rho^{\beta}(s)\nabla_{\theta}\pi_{\theta}(a|s)Q^{\pi}(s,a)dads$$

$$=\int_{S}\int_{A}\rho^{\beta}(s)\pi_{\theta}(a|s)\nabla_{\theta}log\pi_{\theta}(a|s)Q^{\pi}(s,a)dads$$

$$=\int_{S}\int_{A}\rho^{\beta}(s)\beta(a|s)\frac{\pi_{\theta}(a|s)}{\beta(a|s)}\nabla_{\theta}log\pi_{\theta}(a|s)Q^{\pi}(a|s)dads$$

$$=E_{s\sim\rho^{\beta},a\sim\beta}[\frac{\pi_{\theta}(a|s)}{\beta(a|s)}\nabla_{\theta}log\pi_{\theta}(a|s)Q^{\pi}(a|s)]$$

在更新策略参数的时候，我们就用采样得到的轨迹目标函数梯度的平均（对期望的估计）更新策略的参数。这个采样过程也称为重要性采样。

三、确定性策略梯度算法（Deterministic Policy Gradient Algorithms)

确定性策略梯度算法和随机策略梯度算法最大的不同就是策略的形式（一个是带有随机性，一个是确定的），除此自外这两者的发展思路都差不多。

一些基本的符号定义：

确定性策略：$\mu_\theta(s)$
状态价值函数: $V^\mu(s)$
动作价值函数：$Q^\mu(s,a)=Q^\mu(s,\mu_\theta(s))$

确定性策略梯度理论（Deterministic Policy Gradient Theorem）

目标函数（performance objective）：

$$J(\mu_\theta)=\int_S \rho^\mu(s)r(s,\mu_\theta(s))ds$$

$$=E_{s\sim\rho^\mu}[r(s,\mu_\theta(s))]$$

目标函数的梯度：

$$\nabla_\theta J(\mu_\theta)=\int_S \rho^\mu(s)\nabla_\theta\mu_\theta(s)\nabla_a Q^\mu(s,a)|_{a=\mu_\theta(s)}ds$$

$$=E_{s\sim\rho^\mu}[\nabla_\theta\mu_\theta(s)\nabla_a Q^\mu(s,a)|_{a=\mu_\theta(s)}]$$

理论内容：
这个和随机策略梯度理论基本上一样，就是将策略的参数向目标函数的梯度方向调整，以最大化目标函数。参数更新的数学形式的表达即为：

$$\theta_{k+1}=\theta_k+\alpha E_{s\sim\rho^{\mu_k}}[\nabla_\theta\mu_\theta(s)\nabla_a Q^{\mu_k}(s,a)|_{a=\mu_\theta(s)}]$$

注意到，这里的目标函数也是期望的形式，但是和随机策略梯度算法中的目标函数不同之处在于，这里的期望是只对状态空间进行积分，而随机策略梯度算法中的目标函数是对状态空间以及动作空间进行积分。这也就意味着在通过采样估计目标函数的时候，确定性策略梯度算法需要的采样数更少。在论文中作者还证明了确定性策略梯度理论是随机策略梯度理论的一种极限情况，这一点就不列在这里了，有需要的可以去看原论文。

确定性演员-评论员算法（Deterministic Actor-Critic Algorithms）

（这里最核心的是关注 θ 是如何更新的）

同策略演员-评论员算法（On-Policy Deterministic Actor-Critic）

一般来说，确定性的策略在采样的时候很难有多样性，除非是在环境噪声很大的情况下。这里介绍同策略演员-评论员算法只是为了更好地理解确定性演员-评论员算法。
该算法分为两步：更新动作价值函数拟合器$Q^w(s,a)$的参数，使其更好地拟合真实的动作价值函数 $Q^\mu(s,a)$；更新确定性策略 $\mu_\theta(s)$ 的参数 θ，进行梯度上升。
数学形式如下：

$$\delta_t=r_t+\gamma Q^w(s_{t+1},a_{t+1})-Q^w(s_t,a_t)$$

$$w_{t+1}=w_t+\alpha_\theta\delta_t\nabla_w Q^w(s_t,a_t)$$

$$\theta_{t+1}=\theta_t+\alpha_\theta\nabla_\theta\mu_\theta(s_t)\nabla_a Q^{\mu_k}(s_t,a_t)|_{a=\mu_\theta(s)}$$

异策略演员-评论员算法（Off-Policy Deterministic Actor-Critic）

在确定性策略梯度算法中应用异策略演员-评论员算法的原因不同于随机策略梯度算法，这里是为了使得采样得到的轨迹更加具有多样性。这里直接给出异策略演员-评论员算法中的目标函数以及目标函数梯度的形式：

$$J_\beta(\mu_\theta)=\int_S \rho^\beta(s)V^\mu(s)ds$$

$$\nabla_\theta J_\beta(\mu_\theta)\approx\int_S \rho^\beta(s)\nabla_\theta\mu_\theta(a|s)Q^\mu(s,a)ds$$

$$=E_{s\sim\rho^\beta}[\nabla_\theta\mu_\theta(s)\nabla_a Q^{\mu_k}(s,a)|_{a=\mu_\theta(s)}]$$

算法更新的数学形式为：

$$\delta_t=r_t+\gamma Q^w(s_{t+1},a_{t+1})-Q^w(s_t,a_t)$$

$$w_{t+1}=w_t+\alpha_\theta\delta_t\nabla_w Q^w(s_t,a_t)$$

$$\theta_{t+1}=\theta_t+\alpha_\theta\nabla_\theta\mu_\theta(s_t)\nabla_a Q^{\mu_k}(s_t,a_t)|_{a=\mu_\theta(s)}$$

注意到，这里的算法更新的形式和同策略演员-评论员算法是完全一样的，这是因为我们更新参数的时候并没有用到策略梯度的期望，而是用单个采样轨迹的目标函数的梯度进行更新的，所以同策略和异策略算法在表现形式上完全一样，不同之处在于用于更新的采样轨迹，异策略算法中的采样轨迹更具有多样性。但是我们在实际应用的时候，我认为还是应该用期望来更新策略参数，而不是单个样本的梯度。

四、算法总结

这篇论文介绍的核心的算法就是确定性策略的异策略演员-评论员算法，算法的大概流程就是：

1. 基于探索策略（ $\beta(s)$ ），对轨迹进行采样
2. 利用Sarsa算法或者Q-learning对 $Q^w(s,\mu_\theta(s))$ 的参数w进行更新，使其更好地拟合 $Q^\mu(s,\mu_\theta(s))$
3. 利用梯度上升对确定性策略 $\mu_\theta(s)$ 的参数 θ 进行更新
4. 循环这个过程

注：关于如何更新 $Q^w(s,\mu_\theta(s,a))$ ，这里不做具体介绍。在实际应用算法的时候我们需要考虑如何对 $Q^w(s,a)$ ， $\mu_\theta(s)$ 进行建模，可以用特征工程+线性回归的方法进行建模，也可以考虑用神经网络对其进行建模，但是无论用哪种方式，要考虑建模的时候是否会引入偏差，原论文中对于 $Q^w(s,a)$ 应该满足何种形式才不会引入偏差进行了详细的讨论，我没有在这篇文章中列出来，有需要的可以去找原论文来看。这篇文章写于2014年，那时候深度学习还没有在RL中广泛的应用，所以原论文有些内容并不是以深度学习广泛应用为大前提来写的，我们要认识到这一点。下文提到的COPDAC(compatible off-policy deterministic actorcritic)就理解为对 $Q^w(s,a)$ ， $\mu_\theta(s)$ 进行了特殊的建模就好了，至于怎么建模的，其实不太重要，因为现在一般都用神经网络来建模了。

五、实验

这部分设置了三个情景来验证确定性策略梯度算法的效果。

连续的老虎机（Continuous Bandit）

实验设置

分别在动作空间维数m=10，25，50的setting下应用了stochastic actor-critic (SAC-B)、deterministic actor-critic (COPDAC-B)两种算法。老虎机可以视为一种弱化的强化学习，即一个回合只有一个时间步。这里的cost是衡量目前的策略相比于最优策略，差了多少，所以cost越低越好。

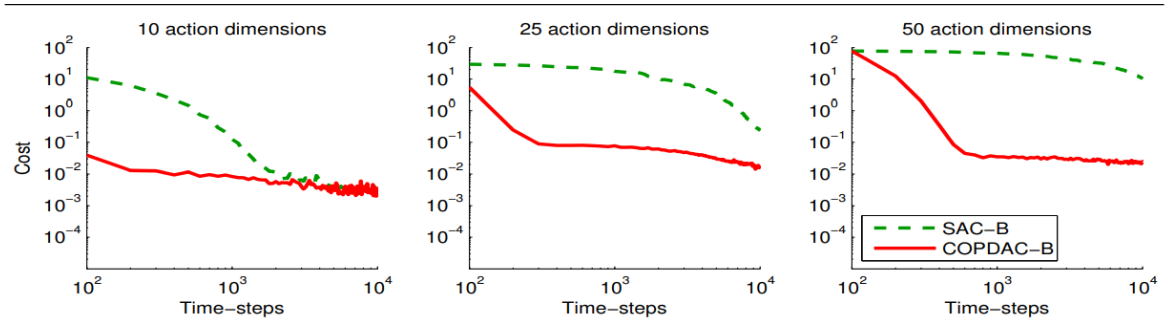


Figure 1. Comparison of stochastic actor-critic (SAC-B) and deterministic actor-critic (COPDAC-B) on the continuous bandit task.

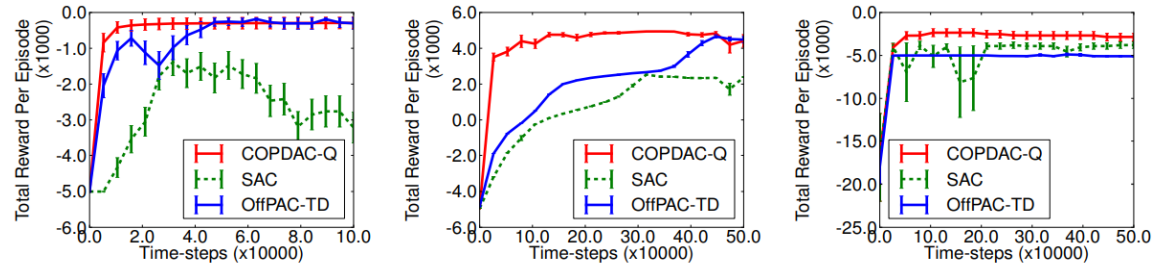
实验结果

可以看出，COPDAC-B的效果远远好于SAC-B，而且动作空间的维数越大，其优势越明显。

连续的强化学习（Continuous Reinforcement Learning）

实验设置

三个游戏环境分别为：mountain car、pendulum、2D puddle world，其动作空间都是连续的。分别在这三个游戏环境中应用stochastic on-policy actor-critic (SAC)、 stochastic off-policy actor-critic (OffPAC)、deterministic off-policy actor-critic (COPDAC)三种算法。



(a) Mountain Car (b) Pendulum (c) 2D Puddle World

Figure 2. Comparison of stochastic on-policy actor-critic (SAC), stochastic off-policy actor-critic (OffPAC), and deterministic off-policy actor-critic (COPDAC) on continuous-action reinforcement learning. Each point is the average test performance of the mean policy.

实验结果

可以看出来，COPDAC-Q在三种环境中均表现最好。而且COPDAC-Q收敛的更快，这是因为他需要的采样数更少。

章鱼手控制（Octopus Arm）

实验设置

Octopus Arm是一个具有多个自由度的机械手，这个实验的目标是这个机械手触碰到目标。实验过程中的奖励和机械手与目标之间的距离改变成正比（即离目标越近，奖励越多），当触碰到目标或者进行了300步时，一个过程结束。这里用两个感知机来建模$Q^w(s,a)$和$\mu_\theta(s)$。实验结果如下：

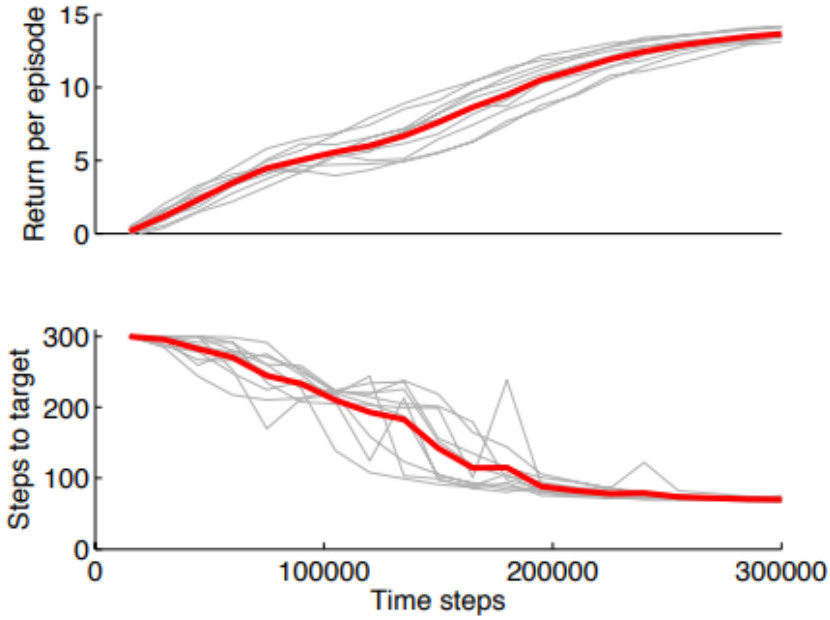


Figure 3. Ten runs of COPDAC on a 6-segment octopus arm with 20 action dimensions and 50 state dimensions; each point represents the return per episode (above) and the number of time-steps for the arm to reach the target (below).

实验结果

可以看到，算法学到了较好的策略，触碰到目标需要的步数越来越少，一个完整过程所得到的回报越来越多。

参考文献

1. Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D. & Riedmiller, M.. (2014). Deterministic Policy Gradient Algorithms. Available from https://proceedings.mlr.press/v32/silver14.html.
2. 论文十问-快速理解论文主旨的框架
3. 蘑菇书EasyRL

==========

作者：赵世天

所属院校：华东师范大学

研究方向：计算机视觉、深度学习

联系邮箱：shitian_zhao@163.com

知乎主页：https://www.zhihu.com/people/bie-lai-wu-yang-39-60

==========

关于我：本科在读，对机器学习感兴趣，现在在做一些多模态和域泛化方面的研究。有时候会在知乎上写一些有意思的文章，欢迎大家去知乎找我玩。