# 基于DDPG算法的股票交易

1. 原理介绍

概括来说，RL要解决的问题是：让agent学习在一个环境中的如何行为动作(act)， 从而获得最大的奖励值总和(total reward)。

这个奖励值一般与agent定义的任务目标关联。

agent需要的主要学习内容：第一是行为策略(action policy)， 第二是规划(planning)。

其中，行为策略的学习目标是最优策略， 也就是使用这样的策略，

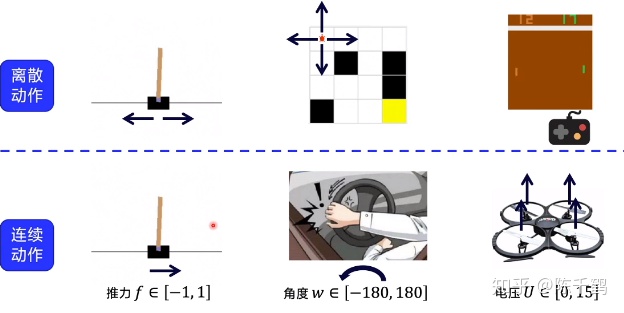
可以让agent在特定环境中的行为获得最大的奖励值，从而实现其任务目标

在RL领域，DDPG主要从：PG -> DPG -> DDPG 发展而来。

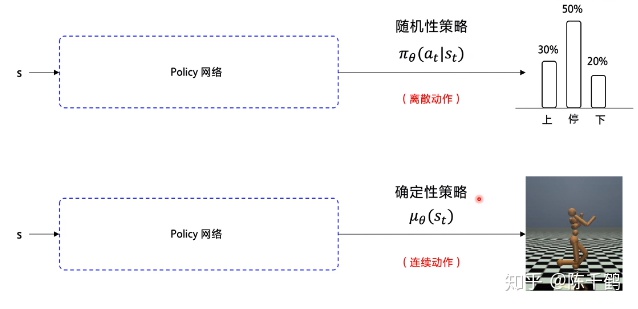
(以下来自知乎陈千鹤大佬回答)

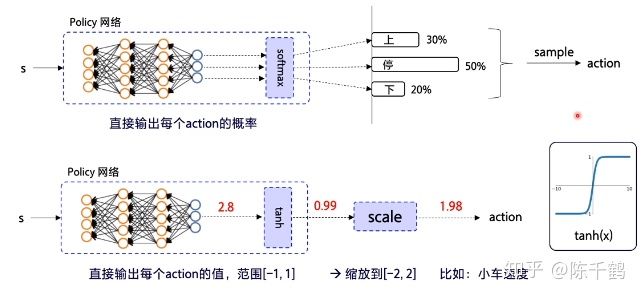
**1、连续动作**

生过中有很多动作不可数的情况，即是连续动作



为了解决连续动作的问题，我们需要使用Policy网络，输出一个确定性的策略，例如一个具体的浮点数等。常见做法就是在网络的最后增加一个tanh层。





2、DDPG简介

DDPG全称是Deep Deterministic Policy Gradient，即深度确定性策略梯度。

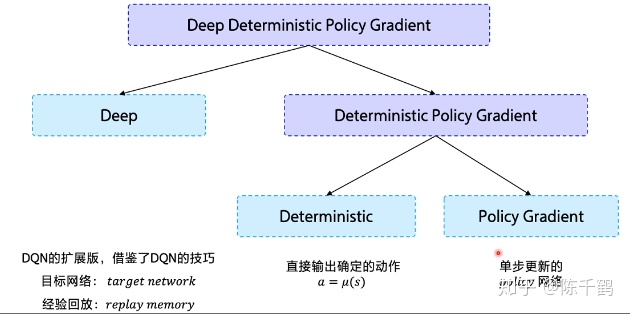
从DDPG这个名字看，它是由D（Deep）+D（Deterministic ）+ PG(Policy Gradient)组成。那么它的特点也可以这么进行拆解。

DDPG的提出动机其实是为了让DQN可以扩展到连续的动作空间。

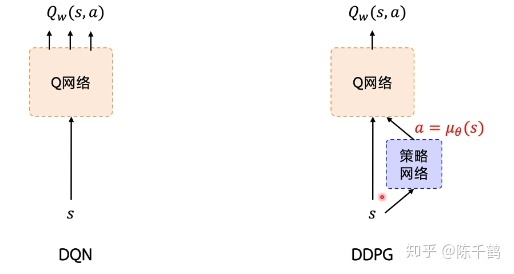
DDPG借鉴了DQN的两个技巧：经验回放 和 固定Q网络。

DDPG使用策略网络直接输出确定性动作。

DDPG使用了Actor-Critic的架构。



DDPG是DQN的扩展版本，可以扩展到连续动作的控制空间。因此它在DQN的基础上增加了一层策略网络，用于输出动作值。DDPG需要一边学习Q网络，一边学习策略网路。



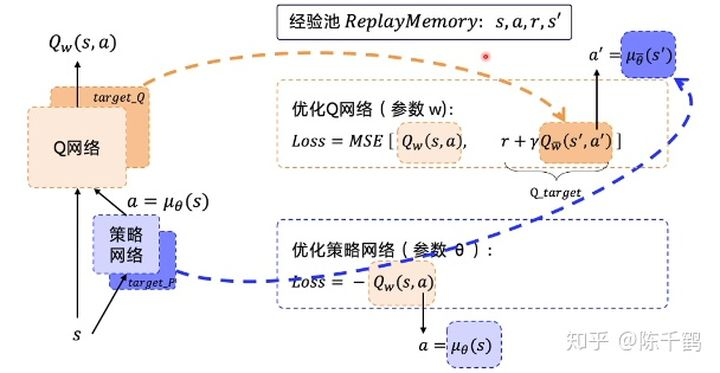
DDPG有4个网络，分别是 Actor当前网络、Actor目标网络、Critic当前网络、Critic目标网络。

Actor当前网络：负责策略网络参数θθ的迭代更新，负责根据当前状态SS选择当前动作AA，用于和环境交互生成S′,RS′,R。

Actor目标网络：负责根据经验回放池中采样的下一状态S′S′选择最优下一动作A′A′。网络参数θ′θ′定期从θθ复制。

Critic当前网络：负责价值网络参数ww的迭代更新，负责计算负责计算当前Q值Q(S,A,w)Q(S,A,w)。目标Q值yi=R+γQ′(S′,A′,w′)yi=R+γQ′(S′,A′,w′)

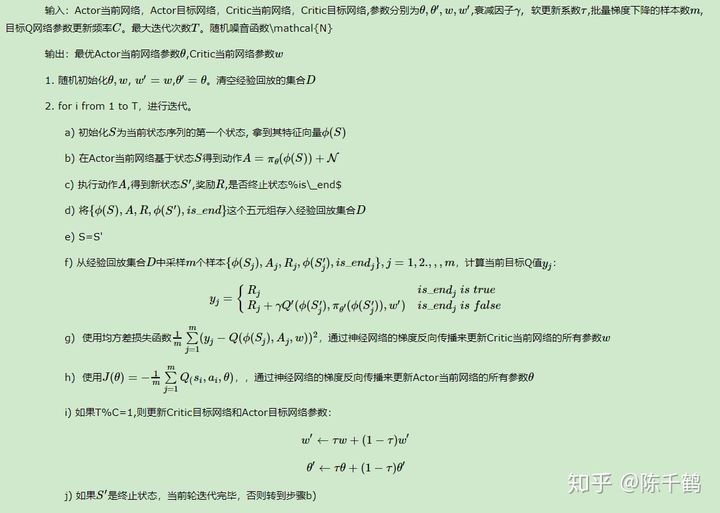
Critic目标网络：负责计算目标Q值中的Q′(S′,A′,w′)Q′(S′,A′,w′)部分。网络参数w′w′定期从ww复制。



DDPG从当前网络到目标网络的复制和我们之前讲到了DQN不一样。回想DQN，我们是直接把将当前Q网络的参数复制到目标Q网络，即w′=ww′=w, DDPG这里没有使用这种硬更新，而是使用了软更新，即每次参数只更新一点点，即： $$ w′←τw+(1−τ)w′\

θ′←τθ+(1−τ)θ′ $$

3、算法流程



使用PARL框架实现整个分为了三个部分：

