策略梯度

**价值逼近器特点:**学习价值函数逼近器;策略由价值逼近器提取;适用于有限动作集的MDPs问题。缺点:策略是确定性的(greedy or -greedy).无法表示随机策略;价值函数逼近器误差往往会导致贪心策略和最优策略之间更大的误差;难以应付大规模动作或连续动作空间.

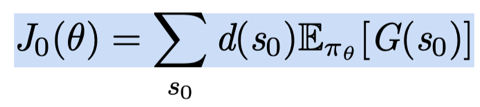
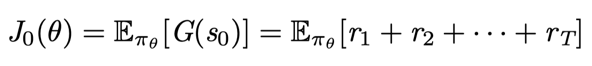
**确定性策略缺点:**只有在价值函数等于最优价值函数时，贪心策略是最优策略。而逼近器与真实函数始终存在逼近误差。贪心策略和最优策略之间的价值误差被该公式放大。

**策略RL**好处:更好的收敛性;有效解决大规模动作集或连续动作空间问题;能够学习随机策略。缺点:收敛到局部最优;策略评估费力且方差大.

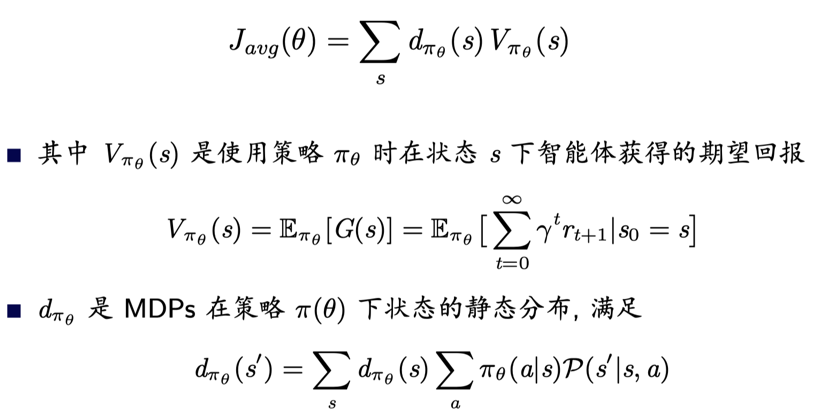
基于策略的强化学习

**策略逼近器有不同的形式**。对有限动作集:表示给定状态下选择某一动作的概率:p(a|s)=pi(a|s,).对连续的动作空间:表示给定状态下选择动作的概率分布或给定状态下的确定性动作a=pi(s,).

**确定逼近器后后我们要定义策略的优化目标:** 不同的场景对应不同的优化目标。episodic场景:每个轨迹都是有限固定步长,从同一初始状态s0或分布s0~d出发,优化目标是整个轨迹的奖励和:

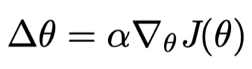
****;

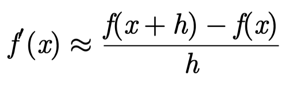
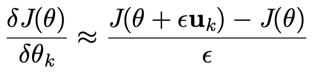
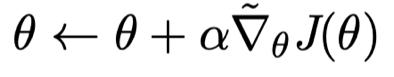
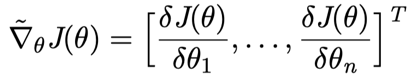
连续运行场景,优化目标是在状态空间上的平均回报:

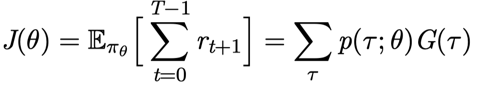
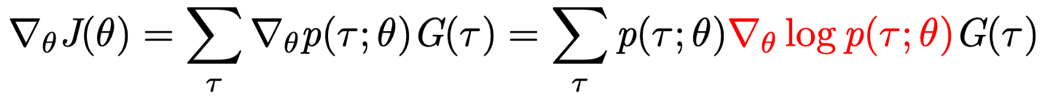
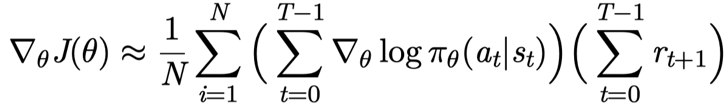
;

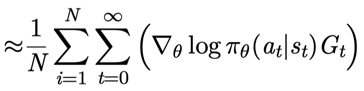
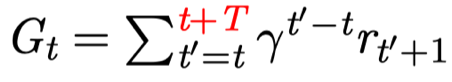
有的连续运行场景，会选择每步的平均奖励(本课不讲)。

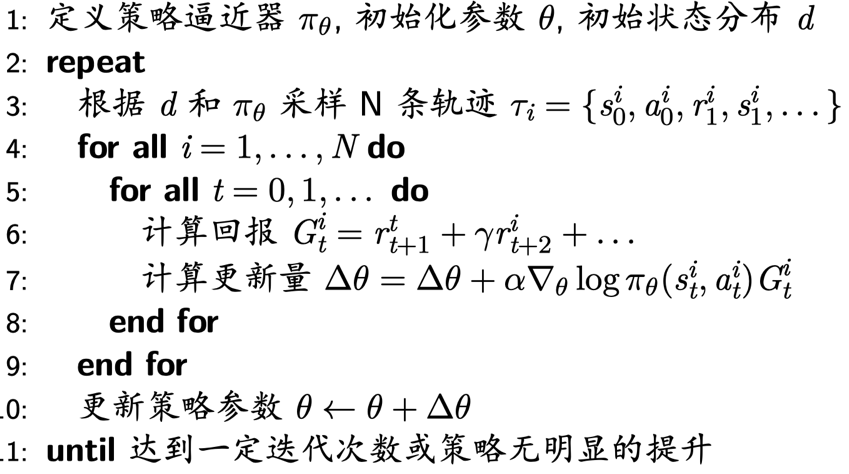
**之后如何解决优化问题?**1无梯度优化算法(爬山法)2基于梯度优化方法(梯度下降GD)。无的优点:适用于任何形式的策略逼近器甚至不可微;容易并行,加速学习.缺陷:计算量大数据利用率低;把问题看成黑箱,没考虑MDPs的时间连贯特性。梯度则数据利用率高;利用了时间连贯特性。本课只考虑梯度策略优化算法。

**策略梯度算法根据梯度上升方向调整**,找到J())的局部最大点:****;是更新步长.找到稳定的策略更新量是策略梯度RL的算法关键。

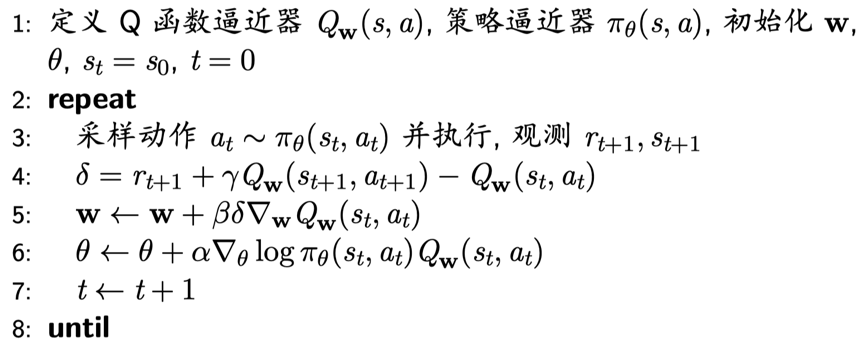
**有限差分法(FD)求:。**对策略参数的每个维度k=1,…,n增加一个微小扰动,用FD近似策略目标梯度.对于episodic问题可以用m次轨迹回报的均值作为目标J())的无偏估计.根据n次的结果构建n维梯度向量,更新参数。缺点:计算量大,每个策略梯度要对n+1个策略评估,每次评估需要生成m条轨迹;方差大用整条轨迹回报近似目标.优点:简单,数值方法不需要分析任何梯度;有时很有效(对低维策略).

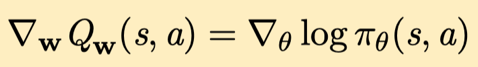
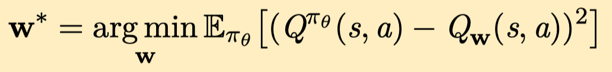
**解析法求策略梯度(无偏但方差巨大，不实用): **;求导.采样N条轨迹，用样本近似策略梯度:.

**REINFORCE算法（蒙特卡洛策略梯度算法）:**利用时间连贯性改变解析法的:t时刻之前的奖励与t时刻的策略无关，所以不影响t时刻的梯度更新;同时将上述扩展到连续运行场景。.进一步,实际算法不可能产生无穷长的轨迹,截断计算.缺点:高方差.

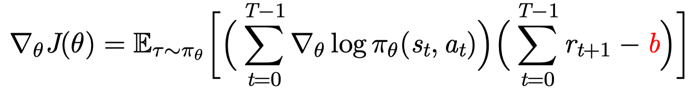
****

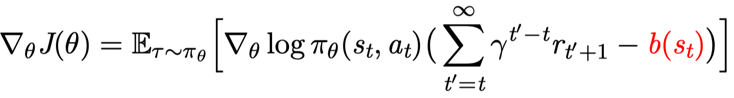
**Actor-Critic:**定义一个Q函数逼近器为Critic,策略逼近器为Actor.Critic更新价值函数的权重从而近似当前策略的价值;Actor则利用Q(降低方差)更新策略梯度。**因为Actor**的策略梯度是基于Critic来定，所以让大一些,Critic学的快一些.



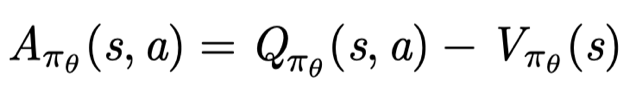
(感觉不会考)如果AC算法Q和pi满足两个条件，则基于Q逼近器的策略梯度是准确的:Q和pi兼容；Q的权重w等于真实的最小均方误差解.

**策略梯度引入基准:**在回报上减去一个baseline不改变梯度.

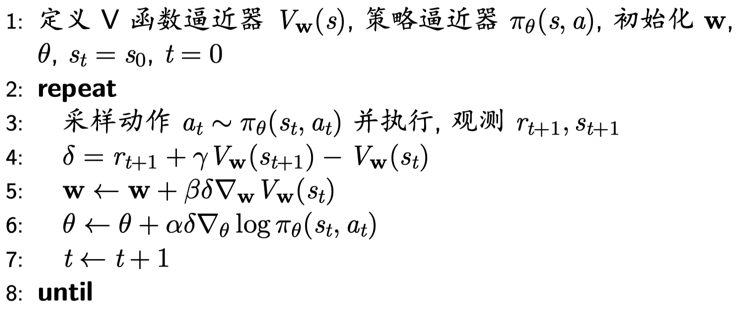
episodic:****.

连续问题:****

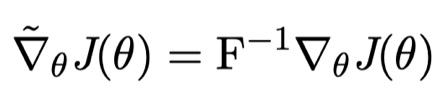
通过对b值的选择，可以方差降到最低。对于episodic,b=.连续问题b()=

**优势函数:****.**不改变梯度，降低方差.可以构造他们的逼近器，再来计算A.也可以只构造V的逼近器，用近似Q,因为.

Advantage Actor-Critic(TD)



**自然梯度(跳过了Natural Actor-Critic):**最速上升方向不再根据欧式距离决定，而根据J的策略概率分布决定.KL散度在概率分布空间上描述两个分布的距离.

**

确定型AC:以上算法主要针对有限动作集MDPs问题即a~.本算法的策略则是确定性的a=.动作的好坏又Q(s,)反应.

