强化学习中“策略梯度定理”的规范表达、推导与讨论

Beaman
上海交通大学自动化系在读硕士，研究方向：强化学习+ 机器人

1. 引言

我写下这篇文章的主要动机是，当我最近在复习强化学习中一个非常重要的概念“策略梯度定理”时，发现在不同的教材、论文和博客教程中，给出了多种一眼看去截然不同的表达方式，这让我产生了深深地困惑。下面，我将首先列出我目前看到的几种策略梯度定理的表达方式：

第1种形式，Sutton 第一版《Reinforcement Learning: An Introduction》

$$\nabla_\theta J(\pi_\theta)=\sum_s d^\pi(s)\sum_a \nabla_\theta \pi_\theta(a|s)Q^\pi(s,a)=\mathbb{E}_\pi\left[\gamma^t\nabla_\theta \log\pi_\theta(A_t|S_t)\,Q^\pi(S_t,A_t)\right] \quad (1)$$

其中 $d^\pi(s)=\sum_{t=0}^\infty \gamma^t \Pr(s_0\to s,t,\pi)=\sum_{t=0}^\infty\gamma^t\Pr\{S_t=s|s_0,\pi\}$ 叫做discounted state distribution。

第2种形式，David sliver 2014年《Deterministic Policy Gradient Algorithm》论文[1]

$$\begin{aligned}\nabla_\theta J(\pi_\theta)&=\int_S\rho^\pi(s)\int_A\nabla_\theta\pi_\theta(a|\,s)\,Q^\pi(s,a)dads\\&=\mathbb{E}_{s\sim\rho^\pi,a\sim\pi_\theta}\left[\nabla_\theta\log\pi_\theta(a|s)Q^\pi(s,a)\right]\end{aligned} \quad (2)$$

其中， $\rho^\pi(s)$ 与上述 $d^\pi(s)$ 相同，都是discounted state distribution。

第3种形式，肖志清《强化学习：原理与Python实现》教材

$$\nabla_\theta J(\pi_\theta)=\mathbb{E}\left[\sum_{t=0}^\infty\gamma^t\nabla_\theta\log\pi_\theta(A_t|S_t)\,Q^\pi(S_t,A_t)\right] \quad (3)$$

第4种形式：Sutton 2018年第二版《Reinforcement Learning: An Introduction》[2]

$$\nabla_\theta J(\pi_\theta)\propto\sum_S\mu^\pi(s)\sum_a\nabla_\theta\pi_\theta(a|s)Q^\pi(s,a)=\mathbb{E}_\pi\left[\nabla_\theta\log\pi_\theta(A_t|S_t)\,Q^\pi(S_t,A_t)\right] \quad (4)$$

其中， $\mu^\pi(s)=\lim_{t\to\infty}\Pr(S_t=s\mid s_0,\pi_\theta)$ 是stationary distribution (undiscounted state distribution)。

第5种形式，Open AI spinning up教程[3]

$$\nabla_\theta J(\pi_\theta)=\mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^T\nabla_\theta\log\pi_\theta(a_t|s_t)\,Q^\pi(s_t,a_t)\right] \quad (5)$$

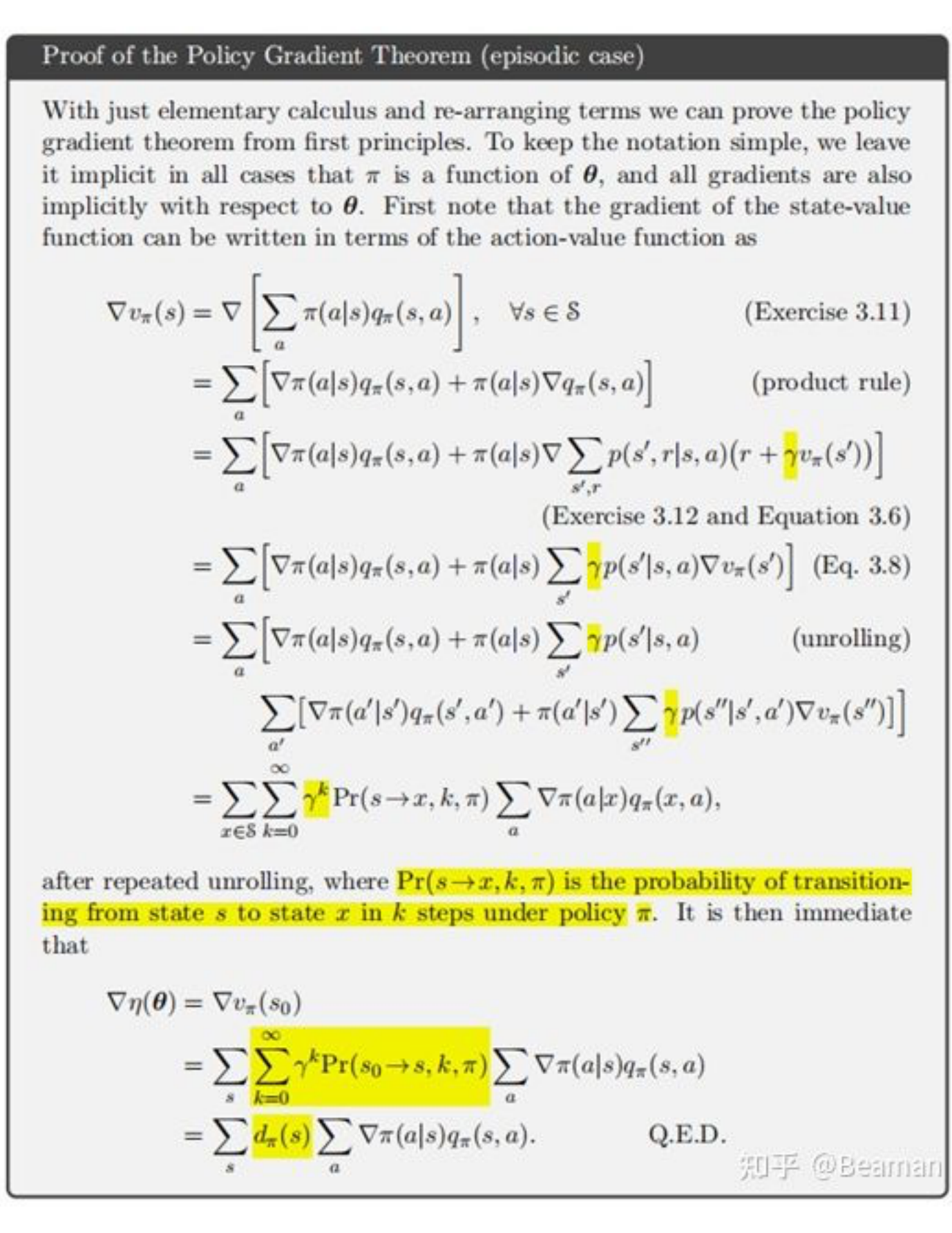
看到这么多种策略梯度定理的表达形式，你是不是也有一点眼花缭乱了呢？上面我列出的这5种形式，其实本质上可以分为两类：形式（1）—（3）是考虑了折扣因子的情况；形式（4）和（5）是不考虑折扣因子的情况。接下来我将会主要推导**考虑折扣因子**的策略梯度定理。（**注**：上面的五种形式中，其实有一些地方存在“随机变量”和“随机变量具体取值”的符号定义混淆。为了保证符号的规范性和统一性，在下面的推导中，我将用大写字母 S_t,A_t 和 S,A 来表示随机变量，用小写字母 s,a 表示随机变量具体的取值）

2 .策略梯度定理的基本形式

在策略梯度定理的形式（1）和形式（2）中的第一个等号本质上是一样的，只不过（1）中写成了求和的形式，（2）中写成了积分的形式。这来自于Sutton在他1999年发表的论文《Policy Gradient Methods for Reinforcement Learning with Function Approximation》[4]中给出的策略梯度定理的基本形式：

$$\nabla_\theta J(\pi_\theta)=\textstyle\sum_s d^\pi(s)\sum_a\nabla_\theta\pi_\theta(a|s)Q^\pi(s,a) \quad (0*)$$

这里我把上述策略梯度定理的基本形式记为形式（0*），它将是我们后面所有推导的基础。这和Sutton在他的第一版书中给出的形式相同，以下是旧版教材中的推导过程。


Proof of the Policy Gradient Theorem (episodic case)

With just elementary calculus and re-arranging terms we can prove the policy gradient theorem from first principles. To keep the notation simple, we leave it implicit in all cases that π is a function of $\boldsymbol{\theta}$, and all gradients are also implicitly with respect to $\boldsymbol{\theta}$. First note that the gradient of the state-value function can be written in terms of the action-value function as

$$\begin{aligned}\nabla v_\pi(s)&=\nabla\left[\sum_a\pi(a|s)q_\pi(s,a)\right],\quad\forall s\in\mathcal{S} &&\text{(Exercise 3.11)}\\&=\sum_a\left[\nabla\pi(a|s)q_\pi(s,a)+\pi(a|s)\nabla q_\pi(s,a)\right] &&\text{(product rule)}\\&=\sum_a\left[\nabla\pi(a|s)q_\pi(s,a)+\pi(a|s)\nabla\sum_{s',r}p(s',r|s,a)(r+\gamma v_\pi(s'))\right]\\& &&\text{(Exercise 3.12 and Equation 3.6)}\\&=\sum_a\left[\nabla\pi(a|s)q_\pi(s,a)+\pi(a|s)\sum_{s'}\gamma p(s'|s,a)\nabla v_\pi(s')\right] &&\text{(Eq. 3.8)}\\&=\sum_a\left[\nabla\pi(a|s)q_\pi(s,a)+\pi(a|s)\sum_{s'}\gamma p(s'|s,a) &&\text{(unrolling)}\\&\qquad\sum_{a'}\left[\nabla\pi(a'|s')q_\pi(s',a')+\pi(a'|s')\sum_{s''}\gamma p(s''|s',a')\nabla v_\pi(s'')\right]\right]\\&=\sum_{x\in\mathcal{S}}\sum_{k=0}^\infty\gamma^k\Pr(s\to x,k,\pi)\sum_a\nabla\pi(a|x)q_\pi(x,a),\end{aligned}$$

after repeated unrolling, where $\Pr(s\to x,k,\pi)$ is the probability of transitioning from state s to state x in k steps under policy π. It is then immediate that

$$\begin{aligned}\nabla\eta(\boldsymbol{\theta})&=\nabla v_\pi(s_0)\\&=\sum_s\sum_{k=0}^\infty\gamma^k\Pr(s_0\to s,k,\pi)\sum_a\nabla\pi(a|s)q_\pi(s,a)\\&=\sum_s d_\pi(s)\sum_a\nabla\pi(a|s)q_\pi(s,a).\qquad\text{Q.E.D.}\end{aligned}$$

知乎 @Beaman

图1 第一版《Reinforcement Learning: An Introduction》中策略梯度定理的推导过程

注意，在旧版的推导过程中，由于考虑了折扣因子，因此最后会出现一个discounted state distribution

$$d^\pi(s)=\sum_{t=0}^\infty\gamma^t\Pr(s_0\to s,t,\pi)=\sum_{t=0}^\infty\gamma^t\Pr\{S_t=s|s_0,\pi\} \quad (6)$$

这两种“折扣状态分布”的写法本质上是一样的，第一种形式出现在Sutton教材中，$\Pr(s_0\to s,t,\pi)$ 可以理解成“在策略 π 的作用下，从状态 s_0 开始经过 t 步转移到状态 s 的概率”（这里我为了统一符号，把教材中的 k 换成了 t）；第二种形式出现在Sutton的1999年的论文中[4]，$\Pr\{S_t=s\mid s_0,\pi\}$ 可以理解为“在策略 π 和初始状态 s_0 的条件下，在 t 时间步的状态 $S_t=s$的概率”。这里，我们需要强调一点，所谓的“discounted state distribution”它本质上不是一个概率分布，因为我们可以证明，它对所有的状态s求和不等于1：

$$\begin{aligned}\sum_s d^\pi(s)&=\sum_s\sum_{t=0}^\infty\gamma^t\Pr\{S_t=s\mid s_0,\pi\}=\sum_{t=0}^\infty\gamma^t\sum_s\Pr\{S_t=s\mid s_0,\pi\}=\sum_{t=0}^\infty\gamma^t \quad (7)\\&=\frac{1}{1-\gamma}\neq1\end{aligned}$$

3. 策略梯度定理的两种期望形式

我们可以看到，引言中列出的 形式（1）和形式（2）的主要区别在于第二个等号后给出的的策略梯度的期望形式不同，同时这二者又和形式（3）中给出的策略梯度的期望形式不同。下面，我将首先给出结论，即我个人认为的在考虑折扣因子的情况下，策略梯度定理最标准的两种期望形式：

$$\nabla_\theta J(\pi_\theta)=\mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^\infty\gamma^t\nabla_\theta\log\pi_\theta(A_t|S_t)\,Q^\pi(S_t,A_t)\right] \quad (1*)$$

$$\nabla_\theta J(\pi_\theta)=\frac{1}{1-\gamma}\mathbb{E}_{S\sim D^\pi,A\sim\pi_\theta}\left[\nabla_\theta\log\pi_\theta(A|S)Q^\pi(S,A)\right] \quad (2*)$$

其中，（2*）中 D^π 表示一个新的概率分布，$D^\pi(s)=(1-\gamma)d^\pi(s)$ 。可能你会发现，我给出的形式（1*）和（2*）与上面给出的形式（1）（2）（3）这三种形式都不一样。别急~下面我将一步一步地从策略梯度定理的基本形式（0*）出发，推导出两种期望形式（1*）和（2*），并推导这两种期望形式之间的关系。

3.1 策略梯度定理的期望形式（1*）的推导

在这一小节中，我将从策略梯度定理的期望形式（0*）出发，给出策略梯度定理期望形式（1*）的推导：

$$\begin{aligned}\nabla_\theta J(\pi_\theta)&=\sum_s d^\pi(s)\sum_a\nabla_\theta\pi_\theta(a|s)Q^\pi(s,a)\\&=\sum_s\sum_{t=0}^\infty\gamma^t\Pr\{S_t=s|s_0,\pi\}\sum_a\nabla_\theta\pi_\theta(a|s)Q^\pi(s,a)\\&=\sum_{t=0}^\infty\gamma^t\sum_s\Pr\{S_t=s|s_0,\pi\}\sum_a\nabla_\theta\pi_\theta(a|s)Q^\pi(s,a)\\&=\sum_{t=0}^\infty\gamma^t\mathbb{E}_{\tau\sim\pi}\left[\sum_a\nabla_\theta\pi_\theta(a|S_t)\,Q^\pi(S_t,a)\right]\\&=\mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^\infty\gamma^t\sum_a\nabla_\theta\pi_\theta(a|S_t)\,Q^\pi(S_t,a)\right]\\&=\mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^\infty\gamma^t\sum_a\pi_\theta(a|S_t)\frac{\nabla_\theta\pi_\theta(a|S_t)}{\pi_\theta(a|S_t)}Q^\pi(S_t,a)\right]\\&=\mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^\infty\gamma^t\nabla_\theta\log\pi_\theta(A_t|S_t)\,Q^\pi(S_t,A_t)\right]\end{aligned}$$

事实上，这个期望形式中有两处出现了折扣因子，第一个出现在动作价值函数 Q^π 中，它本身的定义里包括了一个折扣因子 γ;第二个折扣因子出现在前面的 γ^t 中，它实际代表的是discounted state distribution中所保留下来的折扣。这里需要注意的是，期望符号 $\mathbb{E}$ 的下标 $\tau\sim\pi$ 可以理解为一条轨迹 $\tau=(S_0,A_0,S_1,A_1,\cdots)$ 的随机性是由策略 π 来控制的（因为对于一个MDP，系统的状态转移概率 $p(S_{t+1}|S_t,A_t)$ 是由环境本身确定的，我们人能够控制的只有策略 π，不同的策略会导致不同的轨迹，从而使得轨迹中的 S_t,A_t 均为随机变量，它们的随机性由策略 π 引起。）

3.2 策略梯度定理的期望形式（2*）的推导

在这一小节中，我将从策略梯度定理的期望形式（0*）出发，给出策略梯度定理期望形式（2*）的推导。这里需要注意，在上面的公式（7）中，我们已经证明了“discounted state distribution” **$d^\pi(s)$ 本质上不是一个概率分布，因为他对所有状态的求和不等于1**，而是等于 $\frac{1}{1-\gamma}$ 。但是，我们可以通过把 $d^\pi(s)$ 乘上一个 $(1-\gamma)$,从而构造出一个新的分布 $D^\pi(s)=(1-\gamma)d^\pi(s)$ 。可以证明，这个新构造的分布 $D^\pi(s)$ 对所有状态求和一定等于1。

$$\sum_s D^\pi(s)=\sum_s(1-\gamma)d^\pi(s)=(1-\gamma)\sum_s d^\pi(s)=1 \quad (8)$$

为了构造这个新的分布 $D^\pi(s)$，我们可以在策略梯度定理的基本形式（0*）中乘一个 $(1-\gamma)$ 再除一个 $(1-\gamma)$，因此我们有：

$$\begin{aligned}\nabla_\theta J(\pi_\theta)&=\sum_s d^\pi(s)\sum_a\nabla_\theta\pi_\theta(a|s)Q^\pi(s,a)\\&=\frac{1}{1-\gamma}\sum_s(1-\gamma)d^\pi(s)\sum_a\nabla_\theta\pi_\theta(a|s)Q^\pi(s,a)\\&=\frac{1}{1-\gamma}\sum_s D^\pi(s)\sum_a\nabla_\theta\pi_\theta(a|s)Q^\pi(s,a)\\&=\frac{1}{1-\gamma}\mathbb{E}_{S\sim D^\pi}\left[\sum_a\nabla_\theta\pi_\theta(a|S)Q^\pi(S,a)\right]\\&=\frac{1}{1-\gamma}\mathbb{E}_{S\sim D^\pi}\left[\sum_a\pi_\theta(a|S)\frac{\nabla_\theta\pi_\theta(a|S)}{\pi_\theta(a|S)}Q^\pi(S,a)\right]\\&=\frac{1}{1-\gamma}\mathbb{E}_{S\sim D^\pi,A\sim\pi_\theta}\left[\nabla_\theta\log\pi_\theta(A|S)Q^\pi(S,A)\right]\end{aligned}$$

至此，我们证明了策略梯度定理的第二种期望形式，期望符号 $\mathbb{E}$ 的下标中， $S\sim D^\pi$ 表示状态服从我们新构造的分布 $D^\pi(s)=(1-\gamma)d^\pi(s)$， $A\sim\pi_\theta$ 表示动作采样于策略 π 。

3.3 策略梯度定理的期望形式（1*）与（2*）之间的联系

在3.1和3.2小结中，我们分别给出了策略梯度定理的两种期望形式（1*）和（2*）的推导，下面我们将给出这二者之间的关系，即如何从期望形式（1*）推导出期望形式（2*）：

$$\begin{aligned}\nabla_\theta J(\pi_\theta)&=\mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^\infty\gamma^t\nabla_\theta\log\pi_\theta(A_t|S_t)\,Q^\pi(S_t,A_t)\right]\\&=\mathbb{E}_{S\sim D^\pi,A\sim\pi_\theta}\left[\sum_{t=0}^\infty\gamma^t\nabla_\theta\log\pi_\theta(A|S)Q^\pi(S,A)\right]\\&=\sum_{t=0}^\infty\gamma^t\mathbb{E}_{S\sim D^\pi,A\sim\pi_\theta}\left[\nabla_\theta\log\pi_\theta(A|S)Q^\pi(S,A)\right]\\&=\frac{1}{1-\gamma}\mathbb{E}_{S\sim D^\pi,A\sim\pi_\theta}\left[\nabla_\theta\log\pi_\theta(A|S)Q^\pi(S,A)\right]\end{aligned}$$

上面推导中的第一个等号到第二个等号之间，我们将对 $\tau\sim\pi$ 求期望修改为了对 $S\sim D^\pi,A\sim\pi_\theta$ 求期望，因此第一个期望内的与时间 t 相关的随机变量 S_t,A_t 变成了第二个期望中与时间 t 无关的随机变量 S,A 。进一步，我们可以把 $\sum_{t=0}^\infty\gamma^t$ 提到期望符号的外面，然后直接对它求和，就得到了 $\frac{1}{1-\gamma}$（因为期望符号 $\mathbb{E}$ 内的随机变量已经与时间 t 无关）。综上，我们成功地从期望形式（1*）出发，推导出了期望形式（2*）。

下面我们再整体看一下这两种期望形式（1*）和（2*）之间的联系。事实上，当我们在考虑折扣因子时，在最终策略梯度定理的期望形式中，会在两个地方出现折扣：第一个是**“带折扣的价值函数”**，即 Q 中会包括折扣因子 γ 。第二个是**“带折扣的状态分布”**，在形式（1*）中，由于我们的期望 $\mathbb{E}$ 的下标 $\tau\sim\pi$ 分布中不包括折扣，所以在期望内部保留了一个折扣 γ^t；在形式（2*）中，由于期望 $\mathbb{E}$ 的下标中 $S\sim D^\pi$ 中已经包括了状态的折扣，所以期望内部便不会再有折扣。

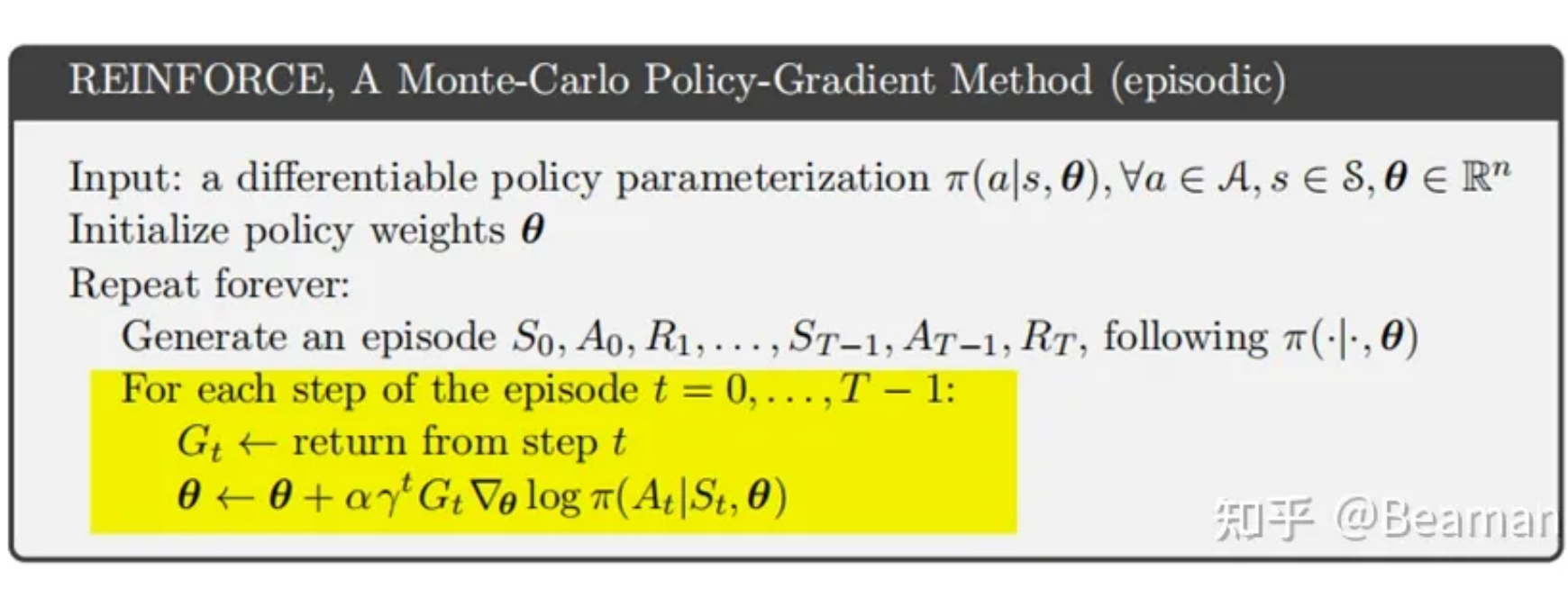
4. 形式（1*）和（2*）与形式（1）—（5）的对比

现在，我们把目前推导出的策略梯度定理的两种期望形式（1*）和（2*）与我在引言中给出的前三种策略梯度定理的形式（1）（2）（3）进行一下对比。

4.1 对于Sutton在第一版教材中给出的形式（1）

$$\nabla_\theta J(\pi_\theta)=\textstyle\sum_s d^\pi(s)\sum_a\nabla_\theta\pi_\theta(a|s)Q^\pi(s,a)=\mathbb{E}_\pi\left[\gamma^t\nabla_\theta\log\pi_\theta(A_t|S_t)\,Q^\pi(S_t,A_t)\right] \quad (1)$$

在我个人的理解中，我对第二个等号给出的期望形式是持质疑态度的。我认为Sutton的本意可能想给出的是我所推导的期望形式（1*），但是他省略了对时间 t 的求和。从具体代码实现的角度来讲，Sutton给出的形式（1）其实可以看作是一种“单步”的策略梯度，以基础的REINFORCE算法举例：


REINFORCE, A Monte-Carlo Policy-Gradient Method (episodic)

Input: a differentiable policy parameterization $\pi(a|s,\boldsymbol{\theta}),\forall a\in\mathcal{A},s\in\mathcal{S},\boldsymbol{\theta}\in\mathbb{R}^n$
Initialize policy weights $\boldsymbol{\theta}$
Repeat forever:
Generate an episode $S_0,A_0,R_1,\ldots,S_{T-1},A_{T-1},R_T$, following $\pi(\cdot|\cdot,\boldsymbol{\theta})$
For each step of the episode $t=0,\ldots,T-1$:
$G_t\leftarrow$ return from step t
$\boldsymbol{\theta}\leftarrow\boldsymbol{\theta}+\alpha\gamma^t G_t\nabla_{\boldsymbol{\theta}}\log\pi(A_t|S_t,\boldsymbol{\theta})$

知乎 @Beaman

图2 Sutton第一版《Reinforcement Learning: An Introduction》中REINFORCE算法伪代码

事实上，如果从一个回合的角度来讲，把在每一个时间步 t 上通过采样得到的梯度 $\gamma^tG_t\nabla_\theta\log\pi_\theta(A_t|S_t)$ 进行累加，就会得到 $\sum_{t=0}^\infty\gamma^tG_t\nabla_\theta\log_\theta(A_t|S_t)$ 。所以从具体实现的角度来讲，这二者没有区别；但是从数学表达的角度来讲，我个人认为我们在上面给出的期望形式（1*）更加准确。同时形式（1*）也和肖智清主编的《强化学习：原理与Python实现》这本书中给出的形式相同，只不过书中给出的期望形式 $\mathbb{E}$ 没有下标，这一点在定义上稍微有一点模糊。（肖智清的书中给出了另一种思路的策略梯度定理的证明，感兴趣的朋友可以找来书看一下）

4.2 对于David Sliver在他2014年发表的论文《Deterministic Policy Gradient Algorithm》中给出的策略梯度定理的形式（2）：

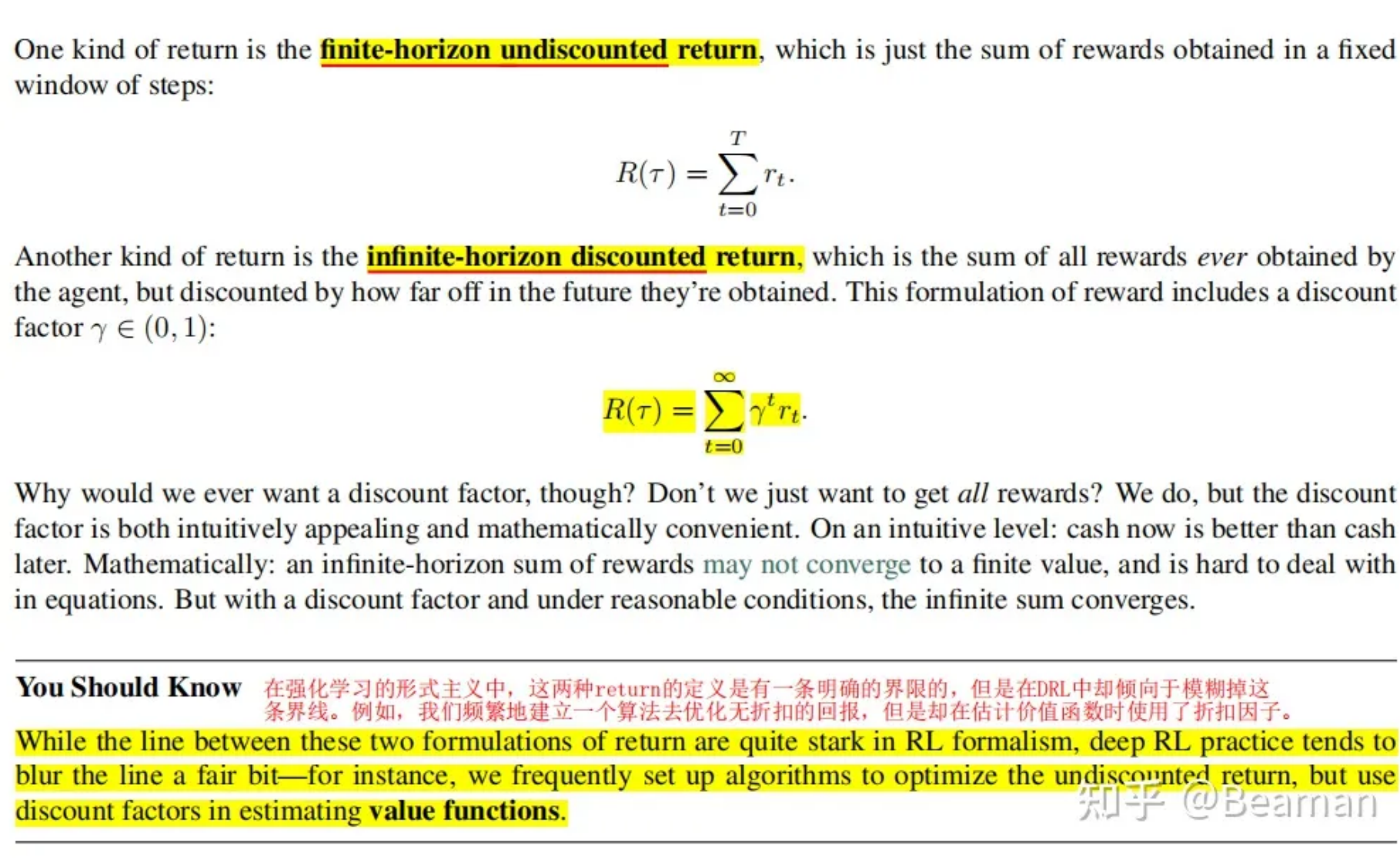
$$\begin{aligned}\nabla_\theta J(\pi_\theta)&=\int_S\rho^\pi(s)\int_A\nabla_\theta\pi_\theta(a|\,s)\,Q^\pi(s,a)dads\\&=\mathbb{E}_{s\sim\rho^\pi,a\sim\pi_\theta}\left[\nabla_\theta\log\pi_\theta(a|s)Q^\pi(s,a)\right]\end{aligned} \quad (2)$$

我认为它十分接近我们给出的期望形式（2*），但是论文中的期望形式忽略了前面的 $\frac{1}{1-\gamma}$，这可能是一处不太严谨的地方。并且形式（2）的第二个等号后期望符号 $\mathbb{E}$ 的下标中， $s\sim\rho^\pi$ 同样不够严谨，因为我们在上面已经证明过，discounted state distribution $d^\pi(s)$ (或论文中的 $\rho^\pi(s)$)本质上不是概率分布。但是从实际算法实现的角度来看，按照David Sliver给出的写法也是没问题的，因为我们实际中都是要用采样的方式来近似期望，并且我们更关注的是梯度的方向而不是大小，前面的系数 $\frac{1}{1-\gamma}$ 可以吸收到学习率中。

另外，形式（2）中关于状态和动作符号的使用也有一点“模糊”：第一个等号后是对所有状态和动作求和（积分），所以应该用小写字母 s,a；第二个等号后面是期望形式，期望内的状态和动作应该是随机变量，如果想要更准确的表达，应该加以区分，即期望形式中用大写字母 S,A 来表示随机变量。

4.3 与形式（4）和（5）的比较

此外，对于前面列出的不考虑折扣的形式（4）和（5），他们在本质上也都存在一点问题。这两种形式在推导的过程中都没有考虑折扣，那么按理来说最后的期望形式中的动作价值函数Q也不应该带折扣因子。但是在实际实现的过程中，往往是用了“带折扣的价值函数”和“不带折扣状态分布”这样的形式。这在Open AI spinning up的教程中也提到了这一点：


One kind of return is the finite-horizon undiscounted return, which is just the sum of rewards obtained in a fixed window of steps:

$$R(\tau)=\sum_{t=0}^T r_t.$$

Another kind of return is the infinite-horizon discounted return, which is the sum of all rewards *ever* obtained by the agent, but discounted by how far off in the future they're obtained. This formulation of reward includes a discount factor $\gamma\in(0,1)$:

$$R(\tau)=\sum_{t=0}^\infty\gamma^t r_t.$$

Why would we ever want a discount factor, though? Don't we just want to get *all* rewards? We do, but the discount factor is both intuitively appealing and mathematically convenient. On an intuitive level: cash now is better than cash later. Mathematically: an infinite-horizon sum of rewards may not converge to a finite value, and is hard to deal with in equations. But with a discount factor and under reasonable conditions, the infinite sum converges.

You Should Know 在强化学习的形式主义中，这两种return的定义是有一条明确的界限的，但是在DRL中却倾向于模糊掉这条界线。例如，我们频繁地建立一个算法去优化无折扣的回报，但是却在估计价值函数时使用了折扣因子。
While the line between these two formulations of return are quite stark in RL formalism, deep RL practice tends to blur the line a fair bit—for instance, we frequently set up algorithms to optimize the undiscounted return, but use discount factors in estimating **value functions**.

知乎 @Beaman

图3 Open AI spinning up的教程

关于为什么要这样做，我在另一篇知乎文章中看到了一个比较合理的解释[5]：考虑一个回合制的通关类的游戏，某一些状态只会在一个回合的最后几步才会遇到，如果同时考虑“带折扣的价值函数”和“带折扣的状态分布”，那么最后的几个状态由于 γ^t 的原因从而导致计算出的梯度非常小，也就是说，“赋予那些需要长时间才能抵达的状态很低的权重，往往会使得性能不好”。

此外，在文章《Bias in Natural Actor-Critic Algorithms》[6]中，这个作者也探讨了关于策略梯度定理的期望形式中 γ^t 的问题，指出很多主流算法都忽略了状态分布中的折扣，这会引起Bias，但在算法实现时却有实际的意义。

5. 总结

最后对本篇文章做一个简单的总结。我根据个人的理解，依据Sutton在1999年的论文[4]中给出的策略梯度定理的基本形式（0*），从两个角度推导出了策略梯度定理的两种期望形式（1*）和（2*），并推导出了二者之间的联系。据我所知，我目前没有在网上看到过类似的推导，因此写下了这篇文章。我最大的目的是希望可以提供一种策略梯度定理期望形式的规范化的表达。因为目前在各种论文、教材中对策略梯度定理的表述五花八门，并且对于一些符号的使用也不够规范和统一，这使得我自己在学习的过程中产生了很多疑惑。希望通过我的这篇文章，可以让其他正在学习强化学习的朋友少走一点弯路，加深一下对策略梯度定理的理解。同时，如果各位朋友对于我上述的任何观点或推导存在异议，欢迎与我讨论并提出建议，我将不胜感激~

参考

1. ^ Silver D, Lever G, Heess N, et al. Deterministic policy gradient algorithms[C]//International conference on machine learning. PMLR, 2014: 387-395.
2. ^ Sutton R S, Barto A G. Reinforcement learning: An introduction[M]. MIT press, 2018.
3. ^ https://spinningup.openai.com/en/latest/
4. ^ a b c Sutton R S, McAllester D, Singh S, et al. Policy gradient methods for reinforcement learning with function approximation[J]. Advances in neural information processing systems, 1999, 12.
5. ^ https://zhuanlan.zhihu.com/p/354935915?ivk_sa=1024320u
6. ^ Thomas P. Bias in natural actor-critic algorithms[C]//International conference on machine learning. PMLR, 2014: 441-448.

编辑于 2022-08-03 18:58

强化学习 (Reinforcement Learning)　深度强化学习