强化学习基础篇（十八）TD与MC方法的对立统一

1、TD与MC在Bias与Variance的区别

前面介绍TD的过程中，我们已经提到过一些TD和MC的区别，例如在Bias与Variance角度看：

a. MC具有高方差，为无偏估计

- 具有良好的收敛性（甚至在值函数近似的场景也有良好的收敛性）
- MC对初始值得设置并不敏感
- 理解容易，易于使用

b. TD具有低方差，为有偏估计

- 其效率通常高于MC方法
- $TD(0)$可以收敛到$v_\pi(S_t)$（但是在值函数近似的场景不总是能收敛到$v_\pi(S_t)$）
- TD对初始值得设置较为敏感

2、TD与MC在序列完整性上的区别

a. TD可以在知道最终结果之前就进行学习。

- 我们可以使用TD进行在线学习，每一个时间步之后就可以进行学习。
- MC确是必须等到当前幕介绍，例如必须开车到家后知道最终的时间才能进行学习。

b. TD也可以在没有最终输出的场景下进行学习。

- 就算是序列不完整，TD也是可以学习的，而MC方法必须依赖于完整的序列。
- TD方法可以应用于连续的环境任务（没有结束点），而MC方法比必须应用于具有结束点的环境。

3、批量更新方法

假设只有有限的经验，比如10幕数据或100个时间步。在这种情况下，使用增量学习方法的一般方式是反复地呈现这些经验，直到方法最后收敛到一个答案为止。给定近似价值函数V，在访问非终止状态的每个时刻t，使用下面两式计算相应的增量但是价值函数仅根据所有增量的和改变一次。

$$V(S_t) \leftarrow V(S_t) + \alpha(G_t - V(S_t))$$
$$V(S_t) \leftarrow V(S_t) + \alpha(R_{t+1} + \gamma V(S_{t+1}) - V(S_t))$$

然后，利用新的值函数再次处理所有可用的经验，产生新的总增量，依此类推，直到价值函数收敛。我们称这种方法为批量更新，因为只有在处理了整批的训练数据后才进行更新。

在批量更新下，如果经验趋于无穷多，只要选择足够小的步长参数α，$TD(0)$就能确定地收敛到与α无关的唯一结果。常数α MC方法在相同条件下也能确定地收敛，但是会收敛到不同的结果。

当然，这是在经验趋于无穷（也即无数次试验）的情况下达到的理想情况，但是实际中我们不可能达到，那如果我们利用有限的经验来对值函数进行估计将得到什么样的结果？比方说，对于下面这K个episode：

$$s_1^1, a_1^1, s_2^1, \ldots, s_{T_1}^1$$
$$\ldots$$
$$s_1^K, a_1^K, s_2^K, \ldots, s_{T_1}^K$$

如果我们重复从这K个episode中进行采样，对于某一次采样得到的样本k应用MC或者$TD(0)$方法，会得到什么样的结论呢？先来看一个例子。

4、AB Example

假设在一个强化学习问题中有A和B两个状态，模型未知，不涉及策略和行为，只涉及状态转换和即时奖励，衰减系数为1。现有如下表所示8个完整状态序列的经历，其中除了第1个状态序列发生了状态转移外，其余7个完整的状态序列均只有一个状态构成。现要求根据现有信息计算状态A、 B的价值分别是多少？

序号	状态转移：奖励
1	A:0，B:0
2	B:1
3	B:1
4	B:1
5	B:1
6	B:1
7	B:1
8	B:0

考虑分别使用MC算法和TD算法来计算状态A、 B的价值：

对于MC算法：

在8个完整的状态序列中，只有第一个序列中包含状态A，因此A价值仅能通过第一个序列来计算：

$$V(A) \leftarrow V(A) + \frac{1}{N(A)}(G_1 - V(A))$$

所以可以得到$V(A) = 0$。

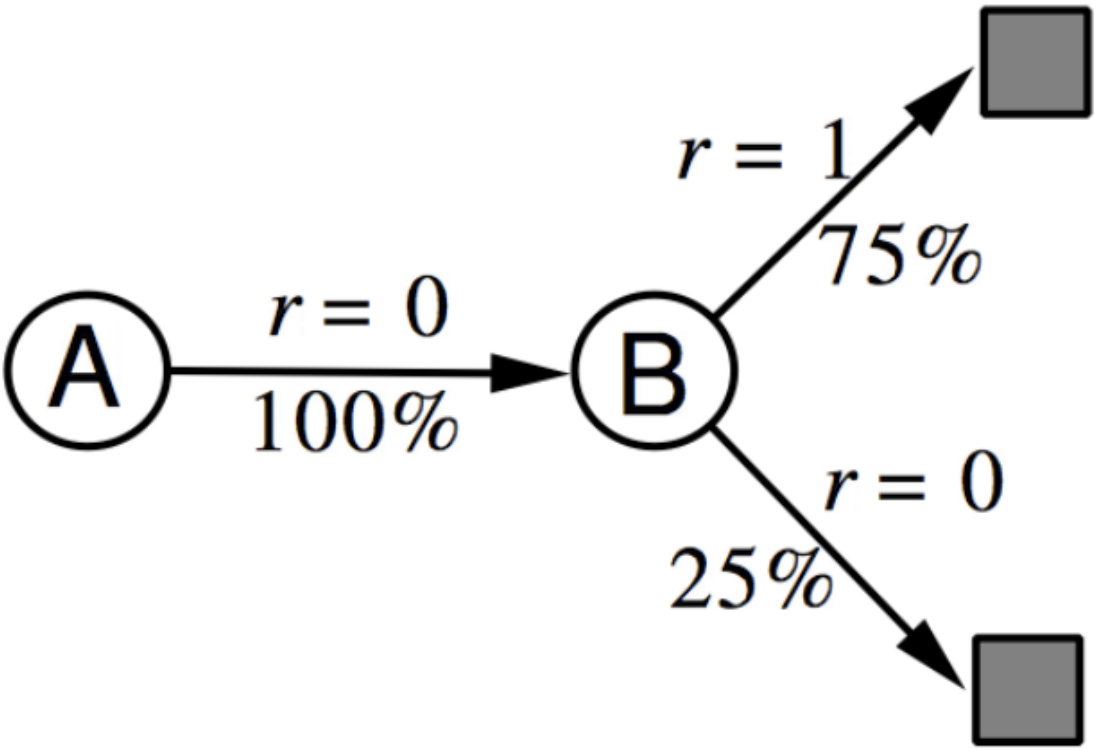
状态B的价值，则需要通过状态B在8个序列中的收获值来平均：

$$V(B) \leftarrow V(B) + \frac{1}{N(B)}(G_1 - V(B))$$

可以得到$V(B) = \frac{6}{8}$。

对于TD算法，

再来考虑应用TD算法。TD算法试图利用现有的episode经验构建一个MDP（如下图），由于存在一个episode使得状态A有后继状态B，因此状态A的价值是通过状态B的价值来计算的，同时经验表明A到B的转移概率是100%，且A状态的即时奖励是0，并且没有衰减，因此A的状态价值等于B的状态价值。



其计算过程如下：

$$\begin{aligned} V(A) &= \pi(a|A)[R_A^a + \gamma P_{AB}^a V(B)] \\ &= 1 * [0 + 1 * 1 * V(B)] \\ &= V(B) \end{aligned}$$

$$\begin{aligned} V(B) &= \pi(b_1|B)[R_B^{b1} + \gamma P_{BB'}^{b1} V(B')] + \pi(b_2|B)[R_B^{b2} + \gamma P_{BB'}^{b2} V(B')] \\ &= 0.75 * [1 + 1 * 1 * 0] + 0.25 * [0 + 1 * 1 * 0] \\ &= 0.75 \end{aligned}$$

因此在TD算法下$V(A) = V(B) = \frac{6}{8}$。

5、确定性等价估计（Certainty Equivalence estimate）

AB Example体现了通过批量$TD(0)$和批量蒙特卡洛方法计算得到的估计值之间的差别。批量蒙特卡洛方法总是找出最小化训练集上均方误差的估计，而批量$TD(0)$总是找出完全符合马尔科夫过程模型的最大似然估计参数。一个参数的最大似然估计是使得生成训练数据的概率最大的参数值。

- MC算法试图收敛至一个能够最小化状态价值与实际收获的均方差的解决方案，这一均方差用公式表示为：

  $$\sum_{k=1}^{K} \sum_{t=1}^{T_k} (G_t^k - V(s_t^k))^2$$

  其中，k表示episode的序号，K为总的episode的数量，t为一个episode内状态序号，T_k为第k个episode的总状态数，G_t^k表示第k个episode里t时刻状态S_t获得的最终回报，$V(S_t^k)$表示的是第k个episode里算法估计的t时刻状态S_t的滑子菇，

- TD算法试图收敛至一个根据已有经验构建的最大似然马尔可夫（以上例为准，A状态只会跳转到B状态，等价于内在动态过程是确定性的估计）模型$< S, A, P, R, \gamma >$的状态价值，也就是说TD算法将首先根据已有经验估计状态间的转移概率：

  $$P_{s,s'}^a = \frac{1}{N(s,a)} \sum_{k=1}^{K} \sum_{t=1}^{T_k} 1(s_t^k, a_t^k, s_{t+1}^k = s, a, s')$$

同时估计某一个状态的即时奖励：

$$R_s^a = \frac{1}{N(s,a)} \sum_{k=1}^{K} \sum_{t=1}^{T_k} 1(s_t^k, a_t^k = s, a) r_t^k$$

TD与MC的另一个差异

- TD算法使用了MDP问题的马尔可夫属性，在Markov环境下更有效；
- 但是MC算法并不利用马尔可夫属性，通常在非Markov环境下更有效。

6、统一的观点看MC，TD与DP

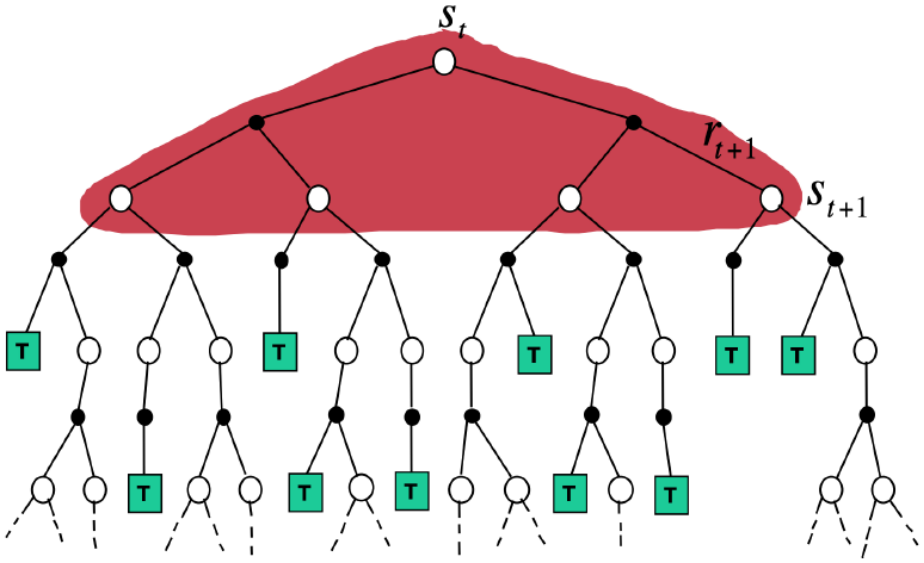
现在为止所阐述的MC学习算法、TD学习算法和DP算法都可以用来计算状态价值。它们的特点也是十分鲜明的，MC和TD是两种在不依赖模型的情况下的常用方法，这其中又以MC学习需要完整的状态序列来更新状态价值，TD学习则不需要完整的状态序列；DP算法则是基于模型的计算状态价值的方法，它通过计算一个状态S所有可能的转移状态S'及其转移概率以及对应的即时奖励来计算这个状态S的价值。

- 在是否使用bootstrapping上， MC学习并不使用bootstrapping，它使用实际产生的奖励值来计算状态价值； TD和DP则都是用后续状态的预估价值作为引导数据来计算当前状态的价值。
- 在是否采样的问题上，MC和TD不依赖模型，使用的都是个体与环境实际交互产生的采样状态序列来计算状态价值的； 而DP则依赖状态转移概率矩阵和奖励函数，全宽度计算状态价值，没有采样之说。

下图，非常直观的体现了三种算法的区别。

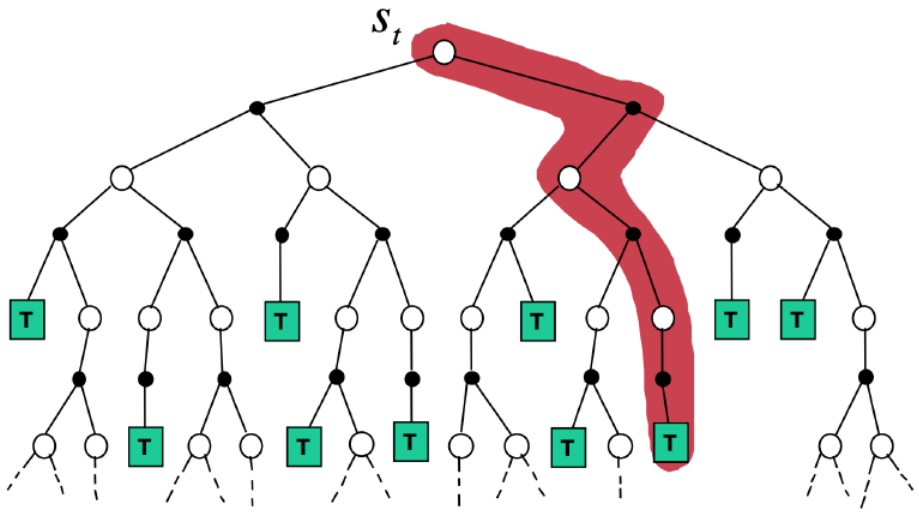
Unified View: Dynamic Programming Backup

$$v(S_t) \leftarrow \mathbb{E}_\pi[R_{t+1} + \gamma v(S_{t+1})]$$



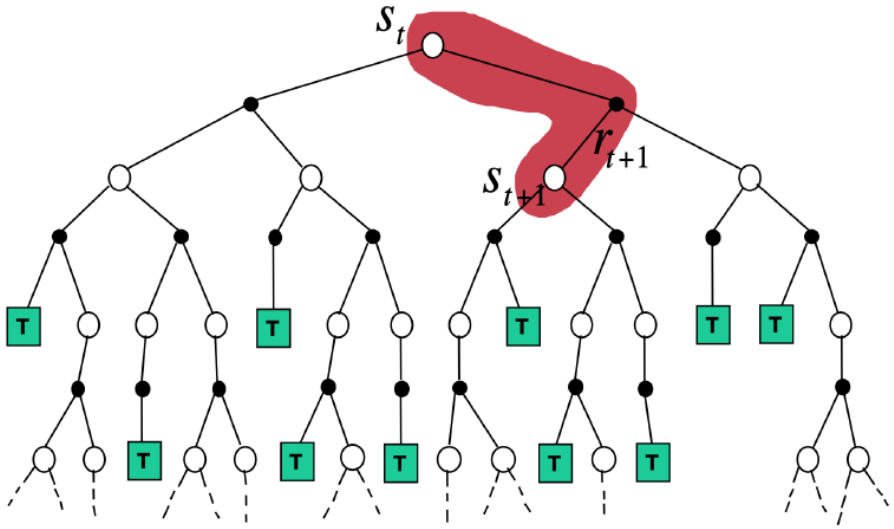
Unified View: Monte-Carlo Backup

$$v(S_t) \leftarrow v(S_t) + \alpha(G_t - v(S_t))$$



Unified View: Temporal-Difference Backup

$$TD(0): v(S_t) \leftarrow v(S_t) + \alpha(R_{t+1} + \gamma v(s_{t+1}) - v(S_t))$$



综合上述三种学习方法的特点，可以小结如下：

当使用单个采样，同时不经历完整的状态序列更新价值的算法是TD学习； 当使用单个采样，但依赖完整状态序列的算法是MC学习； 当考虑全宽度采样，但对每一个采样经历只考虑后续一个状态时的算法是DP学习； 如果既考虑所有状态转移的可能性，同时又依赖完整状态序列的，那么这种算法是穷举（exhausive search）法。

需要说明的是：DP利用的是整个MDP问题的模型，也就是状态转移概率，虽然它并不实际利用采样经历，但它利用了整个模型的规律，因此也被认为是全宽度（full width） 采样的。

Unified View of Reinforcement Learning