1 一些和强化学习相关的证明(一)

1.1 价值函数定义

我们定义agent与环境交互的模式如下：

$$S_t \xrightarrow{A_t} R_{t+1}, S_{t+1} \xrightarrow{A_{t+1}} R_{t+2}, S_{t+2} \tag{1}$$

我们假设agent与环境交互时的动作选择过程遵循Markov Decision Process. 即当前状态和奖励只受上一个状态和其动作选择的影响:

$$p(s', r|s, a) = p(S_{t+1} = s, R_{t+1} = r|S_t = s, A_t = a) \tag{2}$$

π代表agent在与环境交互时遵循的策略, 它是一个动作选择的概率分布. 对于$s \in S$:

$$P_\pi \sim P(A_t|s) \tag{3}$$

我们定义状态$s \in S$的价值为:

$$v_\pi(s) \doteq E_\pi[G_t|S_t = s] \tag{4}$$

其中G_t代表从状态s出发的后续所有奖励，它是后续每一步奖励随机变量$\{R_{t+i}\}_{i=1,..}$的总和.

$$G_t = \sum_{i=1}^{\infty} R_{t+i}\gamma^{i-1} = R_{t+1} + \gamma G_{t+1} \tag{5}$$

1.2 价值函数展开

根据(5)，我们有：

$$v_\pi(s) = E_\pi[R_{t+1} + \gamma G_{t+1}|S_t = s] = E_\pi[R_{t+1}|S_t = s] + \gamma E_\pi[G_{t+1}|S_t = s] \tag{6}$$

$E_\pi[R_{t+1}|S_t = s]$代表状态s在遵循策略π时的即时期望奖励:

$$E_\pi[R_{t+1}|S_t = s] = \sum_a \pi(a|s) \sum_r p(r|s, a)r \tag{7}$$

$E_\pi[G_{t+1}|S_t = s]$代表代表状态s在遵循策略π时的未来期望奖励:

$$E_\pi[G_{t+1}|S_t = s] = \sum_a \pi(a|s) \sum_{s'} p(r|s, a)E_\pi[G_{t+1}|S_{t+1} = s'] = \sum_a \pi(a|s) \sum_{s'} p(r|s, a)v_\pi(s') \tag{8}$$

把上述的展开合并到一起，就可以得出价值函数最常用的展开形式：

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)[r + \gamma v_\pi(s')] \tag{9}$$

上述的展开形式还有可以进一步挖掘的地方,注意到:

$$E_\pi[G_{t+1}|S_t = s] = \sum_a \pi(a|s) \sum_{s'} p(s'|s, a)v_\pi(s') = E_\pi[v_\pi(S_{t+1})|S_t = s] \tag{10}$$

我们可以推导出另一个形式:

$$v_\pi(s) = E_\pi[R_{t+1} + \gamma G_{t+1}|S_t = s] = E_\pi[R_{t+1} + \gamma v_\pi(S_{t+1})|S_t = s] \tag{11}$$

我们可以递推扩展:

$$E_\pi[G_{t+1}|S_t=s]=E_\pi[R_{t+2}+\gamma G_{t+2}|S_t=s] \tag{12}$$

$$E_\pi[G_{t+2}|S_t=s]=\sum_a \pi(a|s)\sum_{s'}p(r|s,a)E_\pi[G_{t+2}|S_{t+1}=s'] \tag{13}$$

根据(10)可得:

$$E_\pi[G_{t+2}|S_{t+1}=s']=\sum_a \pi(a|s)\sum_{s'}p(r|s,a)E_\pi[v_\pi(S_{t+2})|S_{t+1}=s']=E_\pi[v_\pi(S_{t+2})|S_t=s] \tag{14}$$

我们最后可以总结出以下递推展开规律:

$$v_\pi(s)=E_\pi[R_{t+1}+\gamma v_\pi(S_{t+1})|S_t=s]=E_\pi[R_{t+1}+\gamma R_{t+2}+\gamma^2 v_\pi(S_{t+2})|S_t=s] \tag{15}$$

1.3 策略改进

假设我们现在遵循确定性策略π, 我们如何改进策略π? 对于这个问题，首先我们需要明确如何比较两个策略的优劣。

对于当前价值函数集合$v_{\pi'}(S)$。如果我们更换为确定性策略π', 对于遵循策略π'的价值函数集合$v_{\pi'}(S)$,如果对于任意$s\in S$有:

$$v_{\pi'}(s)\geq v_\pi(s) \tag{16}$$

我们就可以说策略π'优于π. 但是我们如何才能找到这个策略π'呢? 下面我先对改进方法进行直观的阐述。

假设我们现在有一个策略π':

$$\pi'(s)=\begin{cases}\pi(s) & \text{if } s\neq s_k,\\ a\neq\pi(s) & \text{if } s=s_k.\end{cases} \tag{17}$$

设$q(\cdot,\cdot)$代表固定状态和动作后的价值函数, 我们可以自然的推导出:

$$q(s,\pi'(s))=\begin{cases}v_\pi(s) & \text{if } s\neq s_k,\\ q(s_k,a) & \text{if } s=s_k.\end{cases} \tag{18}$$

如果$q(s_k,a)>v_\pi(s)$, 实际上我们就可以说策略π'优于π. 而且π'在s_k上的局部优势在求解新的Bellmna方程后会扩散到每一个与s_k有直接或间接联系的s上。将状态转移图分解成多个独立子图，如果状态s和s_k在同一个子图上，那么价值函数$v(s)$会得到提升:

$$v_{\pi'}(s)>v_\pi(s) \tag{19}$$

如果状态s和s_k不在同一个子图上，那么价值函数$v(s)$得到保持:

$$v_{\pi'}(s)=v_\pi(s) \tag{20}$$

对于价值函数集合v_π，如果策略π'在一系列状态$\{s_k\}$上都有$q_\pi(s_k,\pi'(s))>v_\pi(s)$. 那么与$s_k$同在一个子图上的其他状态的价值函数也都会得到提升。

现在让我们整理一下这个尚未证明的直观结论。对于任意$s\in S$,如果有:

$$q_\pi(s,\pi'(s))\geq v_\pi(s) \tag{21}$$

那么我们可以得到一个更优的价值函数集合, 对于任意$s\in S$，我们有:

$$v_{\pi'}(s)\geq v_\pi(s) \tag{22}$$

Proof 1

我们先引入一个可以迭代求解价值函数的方法: Iterative Policy Evaluation.

Iterative Policy Evaluation, for estimating $V \approx v_\pi$

Input π, the policy to be evaluated
Algorithm parameter: a small threshold $\theta > 0$ determining accuracy of estimation
Initialize $V(s)$, for all $s \in \mathcal{S}^+$, arbitrarily except that $V(terminal) = 0$

Loop:
  $\Delta \leftarrow 0$
  Loop for each $s \in \mathcal{S}$:
    $v \leftarrow V(s)$
    $V(s) \leftarrow \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)\big[r + \gamma V(s')\big]$
    $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
until $\Delta < \theta$

Figure 1: policy evaluation

如图所示, Iterative Policy Evaluation可以从任意价值函数集合出发, 用指定策略不断修正价值函数集合，最终得到遵循指定策略的价值函数集合。

我们现在从$v_\pi(S)$出发,用策略π'进行Iterative Policy Evaluation得到$v_{\pi'}(S)$. 如果迭代的每一步我们的价值函数集合都能得到提升, 那么我们就完成了证明。 设$v_\pi(S) = v(S)_0$, 那么对于任意$s \in S$, 在第一个循环:

$$\sum_{r,s'} p(r, s'|s, \pi'(s))[r + \gamma v(s')_0] = q(s, \pi'(s)) \tag{23}$$

$$v(s)_1 = q(s, \pi'(s)) \geq v(s)_0 = v_\pi(s) \tag{24}$$

对于任意$s \in S$, 在第二个循环中我们有:

$$\begin{aligned} v(s)_2 &= \sum_{r,s'} p(r, s'|s, \pi'(s))[r + \gamma v(s')_1] \\ &= \sum_{r,s'} p(r, s'|s, \pi'(s))[r + \gamma q(s, \pi'(s))] \\ &\geq \sum_{r,s'} p(r, s'|s, \pi'(s))[r + \gamma v(s)_0] \\ &= v(s)_1 \end{aligned} \tag{25}$$

对于第三个循环:

$$\begin{aligned} v(s)_3 &= \sum_{r,s'} p(r, s'|s, \pi'(s))[r + \gamma v(s')_2] \\ &\geq \sum_{r,s'} p(r, s'|s, \pi'(s))[r + \gamma v(s')_1] \\ &= v(s)_2 \end{aligned} \tag{26}$$

依次类推, 在迭代的过程中我们的$v(s)_t$保持递增直至收敛到$v_{\pi'}(s)$. QED.

对于策略π和状态s, 所有满足$q_\pi(s, A_t) > v_\pi(s)$的动作修改都可以提升价值函数. 将每一个状态s对应的动作修改串联起来就可以求出π'. 为了榨干提升空间，我们可以直接采用贪婪策略求出π':

$$a = argmax_{a'} q_\pi(s, a') \tag{27}$$

最后我们得到了一个Pipeline:

$$v_{\pi_0}(s) \xrightarrow{\pi_1^*} v_{\pi_1}(s) \xrightarrow{\pi_2^*} \xrightarrow{\pi_T^*} v_{\pi_T}(s) \approx v_*(s) \tag{28}$$

对每一步的稳定价值函数集合$v_{\pi_t}(S)$, 我们都用贪婪策略求出新的策略π_{t+1}^*，随后进行Iterative Policy Evaluation得到$v_{\pi_{t+1}}(S)$, 周而复始。 这实际上形成了如下的评估-改进框架:

Policy Iteration (using iterative policy evaluation) for estimating $\pi \approx \pi_*$

1. Initialization
   $V(s) \in \mathbb{R}$ and $\pi(s) \in \mathcal{A}(s)$ arbitrarily for all $s \in \mathcal{S}$

2. Policy Evaluation
   Loop:
      $\Delta \leftarrow 0$
      Loop for each $s \in \mathcal{S}$:
         $v \leftarrow V(s)$
         $V(s) \leftarrow \sum_{s',r} p(s', r|s, \pi(s))[r + \gamma V(s')]$
         $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
   until $\Delta < \theta$ (a small positive number determining the accuracy of estimation)

3. Policy Improvement
   policy-stable $\leftarrow$ *true*
   For each $s \in \mathcal{S}$:
      old-action $\leftarrow \pi(s)$
      $\pi(s) \leftarrow \text{argmax}_a \sum_{s',r} p(s', r|s, a)[r + \gamma V(s')]$
      If *old-action* $\neq \pi(s)$, then *policy-stable* $\leftarrow$ *false*
   If *policy-stable*, then stop and return $V \approx v_*$ and $\pi \approx \pi_*$; else go to 2

Figure 2: Iterative Policy Evaluation

评估-改进框架的是否可以收敛并收敛到到最优呢? 我们先给出最优价值函数$v^*(s)$和最优价值函数$q^*(s, a)$的定义:

$$v^*(s) \doteq max_\pi v_\pi(x) \tag{29}$$

$$q^*(s, a) \doteq max_\pi q_\pi(s, a) \tag{30}$$

Proof 2

如果评估-改进框架不收敛，那么显然框架每次迭代都会产生一个与当前策略不同的新策略，即$\exists s \in S$:

$$\pi_{t+1}(s) \neq \pi_t(s) \tag{31}$$

新策略π_{t+1}也一定在$t = 1..t-1$中没有出现过. 假设$\exists k \leq t-1, \forall s \in S$:

$$\pi_k(s) = \pi_t(s) \tag{32}$$

而$k < t' < t, \forall s \in S$满足:

$$\pi_{t'}(s) \neq \pi_t(s) \tag{33}$$

根据Proof 1, 当条件严格成立时结论也严格成立. 那么我们有如下不等式关系:

$$v_{\pi_k}(s) < v_{\pi_{k+1}}(s) < \cdots < v_{\pi_{t-1}}(s) < v_{\pi_t}(s) = v_{\pi_k}(s) \tag{34}$$

这显然是矛盾的，所以我们每次迭代都会产生一个与之前过往策略都不同的新策略。这隐含着策略空间是无限的。但我们考虑的是离散状态和动作空间，所以策略数量必然是有限的。故评估-改进框架一定会收敛。

当框架收敛时，此时$\forall s \in S$均有:

$$v_{\pi_{T+1}}(s) = max_a q_{\pi_T}(s,a) = v_{\pi_T}(s) \tag{35}$$

之后就是递归重复:

$$v_{\pi_T}(s) = v_{\pi_{T+1}}(s) = v_{\pi_\infty}(s) \tag{36}$$

那么评估-改进框架能否收敛到最优价值函数集合呢？我们继续在1.2节的证明上进行讨论. 我们已知框架在π_T处收敛. 那么$\forall \pi' \in \{\pi\}$, $\forall s \in S$, 策略π_T满足:

$$q_{\pi_T}(s, \pi'(s)) \leq v_{\pi_T}(s) = max_a q_{\pi_T}(s,a) \tag{37}$$

我们将1.2节的证明中所有不等式符号都反过来就可以得到$\forall s \in S$:

$$v_{\pi'}(s) \leq v_{\pi_T}(s) \Leftrightarrow v_{\pi_T}(s) = max_\pi v_\pi(x) \tag{38}$$

根据定义(29)我们可知π_T是最优策略.这说明我们的评估-改进框架不仅收敛, 还可以收敛到最优价值函数集合。

Proof 2的证明过程还可以引出许多有趣的结论:

$$q_{\pi_T}(s,a) = \sum_{s',r} p(s',r|s,a)[r + v_{\pi_T}(s')] = max_\pi \sum_{s',r} p(s',r|s,a)[r + v_\pi(s')] = max_\pi q_\pi(s,a) \tag{39}$$

$$q_{\pi_T}(s,a) = q^*(s,a) \tag{40}$$

$$v^*(s) = max_a q_{\pi_T}(s,a) = max_a q^*(s,a) \tag{41}$$

1.4 价值迭代

我们在1.3节的最后得到了如下结论:

$$v^*(s) = max_a q^*(s,a) = max_a \sum_{s',r} p(s',r|s,a)[r + v^*(s')] \tag{42}$$

(42)类似于Ballman方程, 虽然无法直接求解。但我们可以继续使用Iterative Policy Evaluation进行求解。

Value Iteration, for estimating $\pi \approx \pi_*$

Algorithm parameter: a small threshold $\theta > 0$ determining accuracy of estimation
Initialize $V(s)$, for all $s \in \mathcal{S}^+$, arbitrarily except that $V(terminal) = 0$

Loop:
| $\Delta \leftarrow 0$
| Loop for each $s \in \mathcal{S}$:
| $v \leftarrow V(s)$
| $V(s) \leftarrow \max_a \sum_{s',r} p(s',r|s,a)[r + \gamma V(s')]$
| $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
until $\Delta < \theta$

Output a deterministic policy, $\pi \approx \pi_*$, such that
$\pi(s) = \arg\max_a \sum_{s',r} p(s',r|s,a)[r + \gamma V(s')]$

Figure 3: Value Evaluation

上图过程被称为价值迭代。与1.3节的评估-改进框架相比，价值迭代框架对每种策略只做一次近似评估从而大大简化了过程。