蒙特卡洛方法

刘婷婷

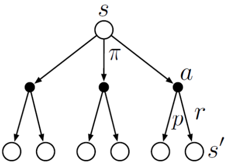
2020 年 5 月 22 日

目录

动态规划的局限

- 状态价值更新：

$$v^{(k+1)}(s) = \sum_{a \in A} \pi(a|s)(R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a v^{(k)}(s'))$$

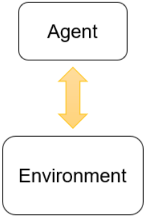


Backup diagram for v_π

- 两个局限：
  - 每次更新一个状态的价值，需要遍历计算后续所有状态的价值。
  - 很多时候，状态转移概率 $P_{ss'}^a$ 未知，无法使用动态规划求解。

蒙特卡洛方法

- 并非一个特定的算法，而是一类随机算法的统称
- 基本思想：用事件发生的“频率”来决定事件的“概率”
- MC 方法特点：
  - 可以通过随机采样得到近似结果
  - 采样越多，越近似真实值
- Model-free：蒙特卡洛方法直接从采样的轨迹序列中学习



轨迹序列1：loc1, east, 2, loc3, west, 4, loc10, north, 10, des

轨迹序列2：loc2, north, 4, loc1, west, -1, loc6, south, 7, des

轨迹序列3：loc8, east, 0, loc6, west, 4, loc7, east, 8, des

⋮

蒙特卡洛预测 (策略评估)

- 目标：根据策略 π 采样的轨迹序列学习 $v_\pi(s)$

$$S_1, A_1, R_2, \cdots, S_k \sim \pi$$

- 价值函数 $v_\pi(s)$ 的定义：

$$v_\pi(s) = \mathbb{E}_\pi(G_t | S_t = s)$$

- 累积奖励 G_t 的定义：

$$G_t = R_{t+1} + \gamma R_{t+2} + \cdots \gamma^{T-1} R_T$$

- 蒙特卡洛策略评估：使用经验奖励的平均来代替期望奖励

$$v_\pi(s) \approx average(G_t), \quad s.t. \quad S_t = s$$

序列中重复出现的状态

同一个状态在一个完整的轨迹序列中重复出现，该状态的价值该如何计算?

- 首次访问 MC：仅把状态序列中第一次出现状态时的奖励值纳入到奖励平均值的计算
- 每次访问 MC：状态序列中每次出现这个状态，都计算对应的奖励值并纳入到奖励平均值的计算

举例

序列：$loc_1, east, 2, loc_3, west, 4, loc_5, north, -1, loc_3, east, 5, des$
设 $\gamma = 0.9$，计算 loc_3 在这个序列中的收获值。
首次访问 MC:

$$G(loc_3) = 4 + (-1) * 0.9 + 5 * 0.9^2 = 7.15$$

$$N(loc_3) = 1$$

每次访问 MC：

$$G(loc_3) = 4 + (-1) * 0.9 + 5 * 0.9^2 + 5 = 12.15$$

$$N(loc_3) = 2$$

MC 预测问题——增量计算

状态价值计算：平均所有该状态的价值之和，最后取平均
存在问题：浪费存储空间
解决方案：**增量计算**

$$\begin{aligned}\mu_k &= \frac{1}{k}\sum_{j=1}^{k} x_j = \frac{1}{k}(x_k + \sum_{j=1}^{k-1} x_j) \\ &= \frac{1}{k}(x_k + (k-1)\mu_{k-1}) = \mu_{k-1} + \frac{1}{k}(x_k - \mu_{k-1})\end{aligned}$$

状态价值更新公式可以改写为：

$$N(S_t) = N(S_t) + 1$$

$$V(S_t) = V(S_t) + \frac{1}{N(S_t)}(G_t - V(S_t))$$

首次访问 MC 计算价值函数伪代码

First-visit MC prediction, for estimating $V \approx v_\pi$

Input: a policy π to be evaluated

Initialize:

  $V(s) \in \mathbb{R}$, arbitrarily, for all $s \in \mathcal{S}$

  $Returns(s) \leftarrow$ an empty list, for all $s \in \mathcal{S}$

Loop forever (for each episode):

  Generate an episode following π: $S_0, A_0, R_1, S_1, A_1, R_2, \ldots, S_{T-1}, A_{T-1}, R_T$

  $G \leftarrow 0$

  Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:

    $G \leftarrow \gamma G + R_{t+1}$

    Unless S_t appears in $S_0, S_1, \ldots, S_{t-1}$:

      Append G to $Returns(S_t)$

      $V(S_t) \leftarrow \text{average}(Returns(S_t))$

蒙特卡洛控制

控制问题：即找到最优价值函数和最优策略

迭代过程：

$$\pi_0 \xrightarrow{E} q_{\pi_0} \xrightarrow{I} \pi_1 \xrightarrow{E} q_{\pi_1} \xrightarrow{I} \pi_2 \xrightarrow{E} \cdots \xrightarrow{I} \pi_* \xrightarrow{E} q_{\pi_*}$$

动作价值计算公式：

$$N(S_t, A_t) = N(S_t, A_t) + 1$$
$$Q(S_t, A_t) = Q(S_t, A_t) + \frac{1}{N(S_t, A_t)}(G_t - Q(S_t, A_t))$$

MC 方法计算动作价值函数的好处

DP:

$$\pi(s) = \arg\max_a \sum_{s',r} p(s', r|s, a)[r + \gamma V(s')]$$

MC:

$$\pi(s) = \arg\max_a Q(s, a)$$

未访问的状态动作对的价值

问题：当状态空间和动作空间较大，采样序列不充分时，有许多状态-动作对没有被访问到，导致动作价值无法更新。
解决方案：

- 探索开端 (exploring starts)：限制每个状态-动作对都可能作为序列的起始，并且采样次数尽可能多
- 随机策略 (stochastic policy)：选择一个随机策略，该策略保证在每个状态下，每个动作被选择的概率都非 0

探索开端的蒙特卡洛控制

Monte Carlo ES (Exploring Starts), for estimating $\pi \approx \pi_*$

Initialize:

$\pi(s) \in \mathcal{A}(s)$ (arbitrarily), for all $s \in \mathcal{S}$

$Q(s,a) \in \mathbb{R}$ (arbitrarily), for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

$Returns(s,a) \leftarrow$ empty list, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

Loop forever (for each episode):

Choose $S_0 \in \mathcal{S}$, $A_0 \in \mathcal{A}(S_0)$ randomly such that all pairs have probability > 0

Generate an episode from S_0, A_0, following π: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$

$G \leftarrow 0$

Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:

$G \leftarrow \gamma G + R_{t+1}$

Unless the pair S_t, A_t appears in $S_0, A_0, S_1, A_1 \ldots, S_{t-1}, A_{t-1}$:

Append G to $Returns(S_t, A_t)$

$Q(S_t, A_t) \leftarrow \text{average}(Returns(S_t, A_t))$

$\pi(S_t) \leftarrow \arg\max_a Q(S_t, a)$

随机策略方法

选择一个随机策略，保证在每个状态下，所有动作被选择的概率都非 0。
这种方法包含两种策略：

- on-policy：用于采样轨迹序列的策略和要评估和更新的策略是同一个，如：$\epsilon - greedy$
- off-policy：采样轨迹序列的策略和要评估和更新的策略不同，如：重要性采样

On-policy first-visit MC control (for ε-soft policies), estimates $\pi \approx \pi_*$

Algorithm parameter: small $\varepsilon > 0$
Initialize:
  $\pi \leftarrow$ an arbitrary ε-soft policy
  $Q(s,a) \in \mathbb{R}$ (arbitrarily), for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$
  $Returns(s,a) \leftarrow$ empty list, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

Repeat forever (for each episode):
  Generate an episode following π: $S_0, A_0, R_1, \dots, S_{T-1}, A_{T-1}, R_T$
  $G \leftarrow 0$
  Loop for each step of episode, $t = T-1, T-2, \dots, 0$:
    $G \leftarrow \gamma G + R_{t+1}$
    Unless the pair S_t, A_t appears in $S_0, A_0, S_1, A_1 \dots, S_{t-1}, A_{t-1}$:
      Append G to $Returns(S_t, A_t)$
      $Q(S_t, A_t) \leftarrow \text{average}(Returns(S_t, A_t))$
      $A^* \leftarrow \arg\max_a Q(S_t, a)$ (with ties broken arbitrarily)
      For all $a \in \mathcal{A}(S_t)$:

$$\pi(a|S_t) \leftarrow \begin{cases} 1 - \varepsilon + \varepsilon/|\mathcal{A}(S_t)| & \text{if } a = A^* \\ \varepsilon/|\mathcal{A}(S_t)| & \text{if } a \neq A^* \end{cases}$$

off-policy 的学习

off-policy 的方法使用两个策略：

- **行为策略**：用来采样轨迹序列，一般更具探索性
- **目标策略**：待评估和更新的策略，一般使用贪心策略。

重要性采样

用来评估随机变量在一个分布上的期望值，但采用的样本是来自另一个分布。

$$
\begin{aligned}
\mathbb{E}_{X\sim P}[f(X)] &= \sum P(X)f(X) \\
&= \sum Q(X)\frac{P(X)}{Q(X)}f(X) \\
&= \mathbb{E}_{X\sim Q}[\frac{P(X)}{Q(X)}f(X)]
\end{aligned}
$$

重要性采样率

- 根据目标策略和行为策略下采样到某个轨迹的概率比值，得到加权的奖励。该比值称为**重要性采样率**
- 给定初始状态 S_t，接下来的状态动作轨迹为：$A_t, S_{t+1}, A_{t+1}, \cdots, S_T$，其在策略 π 下发生的概率为：

$$
\begin{aligned}
& Pr\{A_t, S_{t+1}, A_{t+1}, \cdots, S_T | S_t\} \\
& = \pi(A_t|S_t)p(S_{t+1}|S_t, A_t)\pi(A_{t+1}|S_{t+1}) \cdots p(S_T|S_{T-1}, A_{T-1}) \\
& = \prod_{k=t}^{T-1} \pi(A_k|S_k)p(S_{k+1}|S_k, A_k)
\end{aligned}
$$

- 重要性采样率：

$$
\rho_{t:T-1} \doteq \frac{\prod_{k=t}^{T-1} \pi(A_k|S_k)p(S_{k+1|S_k,A_k})}{\prod_{k=t}^{T-1} b(A_k|S_k)p(S_{k+1|S_k,A_k})} = \prod_{k=t}^{T-1} \frac{\pi(A_k|S_k)}{b(A_k|S_k)}
$$

价值函数计算

- 原始重要性采样

$$V(s) \doteq \frac{\sum_{t\in\mathcal{T}(s)} \rho_{t:T-1} G_t}{|\mathcal{T}(s)|}$$

- 加权重要性采样

$$V(s) \doteq \frac{\sum_{t\in\mathcal{T}(s)} \rho_{t:T-1} G_t}{\sum_{t\in\mathcal{T}(s)} \rho_{t:T-1}}$$

例子

仅有一条轨迹序列：$s_1, a_1, r_2, s_2, a_2, r_3, des$
加权重要性采样：$V(s_2) = r_3$
原始重要性采样：$V(s_2) = \rho_{t:T-1} * r_3$

两者的区别在于价值估计的偏差和方差。

- 原始重要性采样是对 $v_\pi(s)$ 的无偏估计，但方差较大
- 加权重要性采样是有偏的，但方差更小

总结

- MC 方法基于轨迹序列计算动作价值函数
- 策略评估（预测问题）:

$$Q(S_t, A_t) = Q(S_t, A_t) + \frac{1}{N(S_t, A_t)}(G_t - Q(S_t, A_t))$$

- 策略提升（控制问题）:

$$\pi(s) = \arg\max_a Q(s, a)$$

- 要注意的点:
  - 一个状态在一个完整的轨迹序列中重复出现，其状态价值的计算方法
  - 为提高计算效率，对价值函数进行增量计算
  - 状态-动作对未访问到，价值函数无法更新问题的解决方案
  - 思考动态规划和蒙特卡洛方法的区别