

I Reinforcement Learning

II Foundation

* Monte Carlo

Monte Carlo的状态值函数更新公式如下:

$$V(s_t) \leftarrow V(s_t) + \alpha[R_t - V(s_t)] \quad (1)$$

其中 R_t 是每个episode结束后获得的实际累积回报, α 是学习率,这个式子的直观的理解就是用实际累积回报 R_t 作为状态值函数 $V(s_t)$ 的估计值。具体做法是对每个episode,考察实验中 s_t 的实际累积回报 R_t 和当前估计 $V(s_t)$ 的偏差值,并用该偏差值乘以学习率来更新得到 $V(s_t)$ 的新估值。

* TD(0)

把等式1中 R_t 换成 $r_{t+1} + \gamma V(s_{t+1})$,就得到了TD(0)的状态值函数更新公式:

$$V(s_t) \leftarrow V(s_t) + \alpha[r_{t+1} + \gamma V(s_{t+1}) - V(s_t)] \quad (2)$$

等式2

section name goes here

* term definition

DEF: term - and it's definition

* an example

EX: example heading

* a system of equations

$$\begin{cases} 2x + 4y = 2 \\ 2x + 6y = 3 \end{cases}$$

* working a multistep problem

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \xrightarrow{R_1 + R_2} \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \Rightarrow \begin{cases} x = 1 \\ y = 2 \\ z = 3 \end{cases}$$

* a vector in \mathbb{R}^3

$$v = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

* a multi-step process

A $\xrightarrow{\text{do stuff}}$ B $\xrightarrow{\text{more stuff}}$ C

* an enumerated list

1. this is the first item in an enumerated list

2. this is the second item in an enumerated list

* manually Broken lines

the first line

the second line

the third line

* some math

$$\int_a^b f(x) dx \int f(x) dx \frac{\pi}{2} \sqrt{\theta} n = 1, 2, 3 \dots 4$$