强化学习基础

目录

1.Q-learning

Q-learning 算法是一种 value-based 的强化学习算法，Q即为Q(s,a)，就是在某一时刻的state状态下，采取动作action能够获得收益的期望，环境根据agent的动作反馈相应的reward奖励，所以算法的主要思想就是将state和action构建一张Q_table表存储Q值，然后根据Q值选取能够获得最大收益的动作。

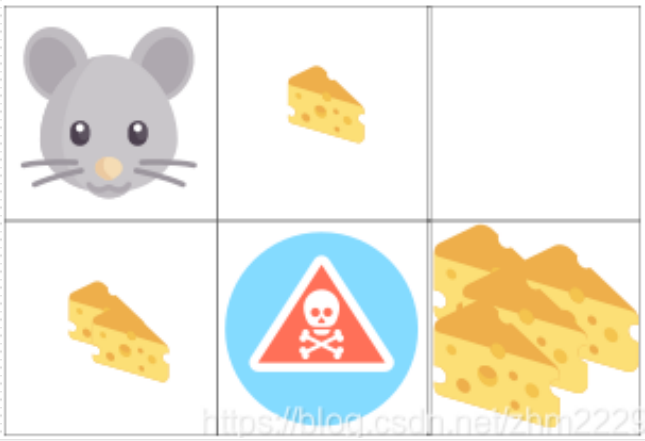
Q-learning是基于off-policy时序差分法，而且使用贝尔曼方程可以对马尔科夫过程求解最优策略。

1. Initialize Q-values ($Q(s,a)$) arbitrarily for all state-action pairs.
2. For life or until learning is stopped...
3. Choose an action (a) in the current world state (s) based on current Q-value estimates ($Q(s,\cdot)$).
4. Take the action (a) and observe the the outcome state (s') and reward (r).
5. Update $Q(s,a) := Q(s,a) + \alpha\left[r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right]$

1.Q-learning

实例：

如图是一个迷宫游戏，agent小老鼠最开始在(0,0)位置，分别在(0,1),(0,2),(1,0),(1,1),(1,2)处可获得+1,0,+2,-10,+10的奖励值。当agent位于(1,1),(1,2)时，游戏结束。Agent的动作有四个，分别时上下左右。



	←	→	↑	↓
Start	0	0	0	0
Small cheese	0	0	0	0
Nothing	0	0	0	0
2 small cheese	0	0	0	0
Death	0	0	0	0
Big cheese	0	0	0	0

1.Q-learning

实例：

Q-learning根据$\epsilon - greedy$选择动作：以ϵ的概率随机选择动作，以$1 - \epsilon$的概率贪心的（greedy）选择动作

	←	→	↑	↓
Start	0	0	0	0
Small cheese	0	0	0	0
Nothing	0	0	0	0
2 small cheese	0	0	0	0
Death	0	0	0	0
Big cheese	0	0	0	0

We move at random (for instance, right)

1.Q-learning

实例：

Q-learning使用贝尔曼方程更新：

$$NewQ(s,a) = Q(s,a) + \alpha[R(s,a) + \gamma max Q'(s',a') - Q(s,a)]$$

New Q value for that state and that action

Current Q value

Learning Rate

Reward for taking that action at that state

Discount rate

Maximum expected future reward **given the new s' and all possible actions at that new state**

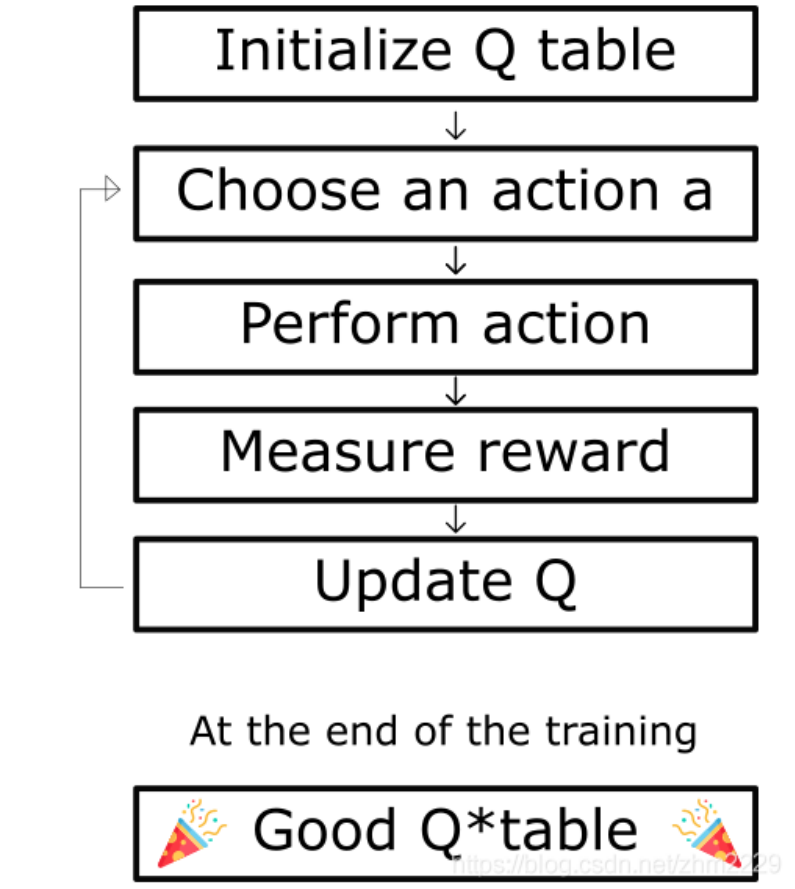
$\gamma = 0.9, \alpha = 0.1$

	←	→	↑	↓
Start	0	0.1	0	0
Small cheese	0	0	0	0
Nothing	0	0	0	0
2 small cheese	0	0	0	0
Death	0	0	0	0
Big cheese	0	0	0	0

1.Q-learning

实例：

Agent在每个step的时候都会用上面的方法迭代更新一次Q-table，直到Q-table不再更新，或者到达游戏设置的结束局数。



2.Sarsa

和Q-learning类似，两者的区别在于：更新Q表的时候，选择的策略不同。Sarsa更新Q表的策略与选择动作策略一致，均采用$\epsilon - greedy$。而Q-learning更新Q表采用$greedy$ 策略，选择动作采用$\epsilon - greedy$ 。

Initialize $Q(s, a)$ arbitrarily
Repeat (for each episode):
  Initialize S
  Choose A from S using policy derived from Q (e.g., ε-greedy)
  Repeat (for each step of episode):
    Take action A, observe R, S'
    Choose A' from S' using policy derived from Q (e.g., ε-greedy)
    $Q(S, A) \leftarrow Q(S, A) + \alpha[R + \gamma Q(S', A') - Q(S, A)]$
    $S \leftarrow S'$; $A \leftarrow A'$;
  until S is terminal

Initialize $Q(s, a)$ arbitrarily
Repeat (for each episode):
  Initialize S
  Repeat (for each step of episode):
    Choose A from S using policy derived from Q (e.g., ε-greedy)
    Take action A, observe R, S'
    $Q(S, A) \leftarrow Q(S, A) + \alpha[R + \gamma \max_a Q(S', a) - Q(S, A)]$
    $S \leftarrow S'$;
  until S is terminal

3 实验任务及报告提交要求

实验任务

- 在给定迷宫环境中实现Q-learning和Sarsa算法。

报告提交要求

- 提交一个压缩包。压缩包命名为：“学号_姓名_作业编号”，例如：20220525_张三_实验10。
- 压缩包包含三部分：code文件夹和实验报告pdf文件
  - Code文件夹：存放实验代码
  - Pdf文件格式参考发的模板
- 如果需要提交新版本，则在压缩包后面加_v1等。如“学号_姓名_作业编号_v1.zip”，以此类推。
- 截止日期：当天完成，即6月9日24点
- 提交邮箱：zhangyc8@mail2.sysu.edu.cn