文字作业

1 我们针对 V 函数给出了策略评价、策略迭代和值迭代算法。现在要求:

1 Q 函数的策略评价算法

我们根据贝尔曼期望方程-q函数的定义:

$$Q_{\pi}(s,a) = R(s,a) + \gamma \sum_{s' \in S} P_{ss'}^{a} \sum_{a' \in A} \pi(a'|s') Q_{\pi}(s',a')$$

得到一下迭代式:

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

所以我们可得迭代式评价算法:

```
    for k = 1,2,... do
        for s in S do
            for a in s do
                使用迭代式更新值函数Q_{k+1}(s,a)
            end for
        end for
    end for
```

2 Q 函数的策略迭代算法

由于策略迭代是不断的执行策略评价和策略提升,直到策略不能再被提升为止,所以策略迭代算法为:

```
随机初始化Q(s,a) 和 \pi(s)
repeat
    对于当前策略π,使用迭代式策略评价的算法估算Q_{\pi}(s,a)得到Q(s,a)
    使用贪婪策略提升得到\pi^{\prime}(s)
until 策略保持不变\pi^{\prime}(s) = \pi(s), \forall s
```

3 Q 函数的值迭代算法

我们根据贝尔曼最优方程-q函数的定义:

$q_*(s,a) = R(s,a) + \gamma \sum_{s,\in S} P_{ss,}^{a} \underset{a,}{max} q_*(s,,a,)$

就能直到当前状态的Q函数可以通过后继状态的Q函数来表达,所以我们就可以通过迭代的方式不断评估Q函数,同时在每次迭代时使用贪婪的策略提升,算法为:

```
for k = 1,2,... do
    for s in S do
        通过Q_{k}(s^{\prime}) 更新 Q_{k+1}(s)
    end for
end for
```

2 思考 ε 贪婪策略和贪婪策略有什么不同?各有什么优缺点?

不同在于 ε 贪婪策略还是有一定几率不选择最优动作,带来的影响可能是会增加agent对环境模型的信息掌握程度.
贪婪的策略效率较高,可以很快求出最优解, 但对于环境模型有变化的情况,可能会导致无法求出最优解
ε 贪婪策略效率相对教低, 但可以通过 ε 随机的部分来提升对未知环境的感知,进而求出一些贪婪策略得不到的最优解.

比如不是全观测的迷宫问题,可能需要一部分的随机部分去探索未知的路径,要不然可能一直陷入死路.

编程作业

code1.py