

Assignment 3: Monte Carlo

一、 作业内容：利用强化学习蒙特卡罗算法找到最短路径（完成 mc.py 文件补全）

Part 1: 函数 `Reward_state_action(s, a)`

```
141     # 状态转移函数，需要返回本次动作是否到达终点，本次动作的reward，本次动作后的下一个state
142     def Reward_state_action(s, a):
143         # 向上移动
144         if a == 0:
145             # Fill and submit this code
146
147         # 向下移动
148         elif a == 1:
149             # Fill and submit this code
150
151         # 向左移动
152         elif a == 2:
153             # Fill and submit this code
154
155         # 向右移动
156         else:
157             # Fill and submit this code
158
```

Part 2: 函数 `epsilon_greedy(qtem, s, epsilon)`:

```
171     def epsilon_greedy(qtem, s, epsilon):
172         # -----
173         # epsilon_greedy, 用于采样实现的动作选择
174         # -----
175         return action
176
```

Part 3: 函数 Monte_Carlo(num, epsilon, gamma):

```
177 # on policy 基于epsilon-greedy策略进行num次实验，每次实验包含一个完整的episode，再根据episode进行策略改进
178 def Monte_Carlo(num, epsilon, gamma):
179     # -----
180     # 定义状态-动作 函数qfunc (Q[s,a])并初始化
181     # 定义Nqfunc统计某次episode中(s,a)出现的次数
182     # 定义一个实验状态-动作函数qtem用于在采样实验中尽行动作的选择
183     # 进行num次循环
184         # 采用epsilon-greedy策略进行第K次episode采样实验
185         # 定义三个数组用于存储第K次episode采样实验的states,actions,reward序列
186         # 随机选择初始位置
187         # 标记是否到达终点
188         # 基于epsilon-greedy策略选择动作并执行动作
189         # 行动完成后返回:
190             # 这次行动是否到达终点
191             # 这次行动后到达了哪一个状态
192             # 这次行动获得的奖励
193         # 更新实验的轨迹直到得到完整过程(起始位置->终止为止)的序列
194
195         # 针对刚才生成的一幕完整的实验episode，进行策略改进
196         # 定义并初始化回报值 g
197         # 计算采样序列的起始状态回报值 g
198         # 反向遍历采样到的序列，进行计算
199         # 正向遍历采样到的序列当中的每一个状态-动作对，并更新qfunc
200         # 把新的s-a的回报g和旧的qfunc[s,a]的回报，一起重新计算，求得更新后的 qfunc[s,a]
201         # 需要注意每次计算时都需要更新g，以对应下一个状态
202     # num次循环后得到最终的qfunc
203     # -----
```

Part 4: 函数 get_shortest_path(qfunc):

```
196 #根据最终的qfunc得到最优路径
197 def get_shortest_path(qfunc):
198     # -----
199     # 根据最终得到的qfunc输出最优的路径
200     # -----
201     return s_path
202
203
```

二、作业提交要求：打包提交（实验报告+完整代码（mc.py））

1、实验报告:包括对蒙特卡罗算法的理解、实现的整体思路、实验

运行截图、结果分析与遇到的问题

2、完整代码文件（mc.py）

3、请与 10 月 8 日前将压缩包上传到[北航网盘 class3_MC文件夹下](#)。命名

要求：学号_姓名_蒙特卡罗作业