RL Assignment 3

陈麒麟 517030910155

作业要求:

- 利用 Q-Learning/SARSA 分别模拟计算 12 $\times$ 4 Cliff Walking 的最优路径

代码实现

详细信息参见 *MFC.py* :

- 全局参数在代码头给出，可以根据情况自定义，参数表为:
  - alpha\epsilon\gamma\格子世界长度\格子世界宽度\episodes\batch_size
- 代码模块话实现，observe\greedy\$\epsilon - greedy$ 分别实现

测试

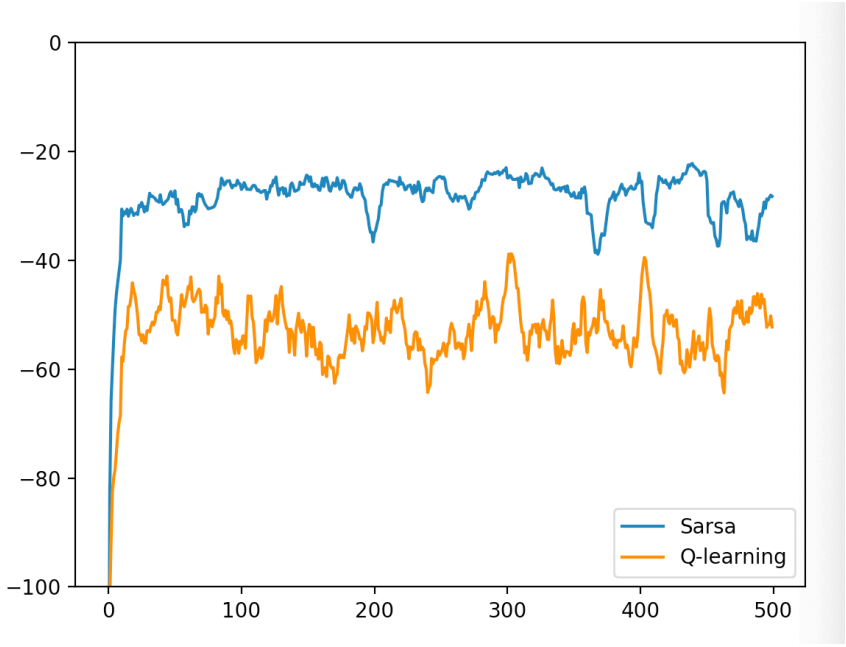
Condition of $\epsilon = 0.1$

- 最优路径对比:

```
optimal travel path by Sara:
→   →   →   →   →   →   →   →   →   →   →   ↓
↑   0   0   0   0   0   0   0   0   0   0   ↓
↑   0   0   0   0   0   0   0   0   0   0   ↓
↑   0   0   0   0   0   0   0   0   0   0   G
optimal travel path by Q-Learning:
0   0   0   0   0   0   0   0   0   0   0   0
0   0   0   0   0   0   0   0   0   0   0   0
→   →   →   →   →   →   →   →   →   →   →   ↓
↑   0   0   0   0   0   0   0   0   0   0   G
```

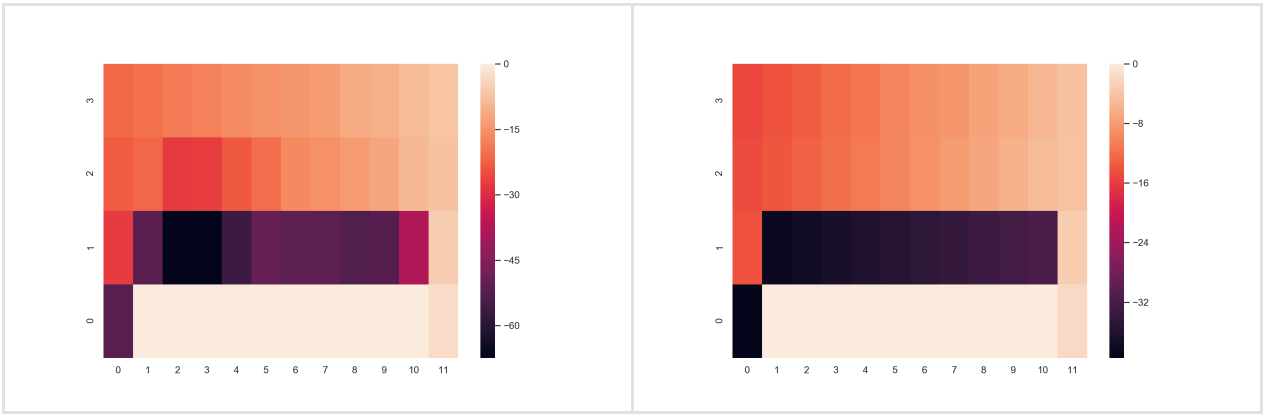
这里我们得到了与Assignment一致的结果，两种算法选择不同路径本质上是因为Q-Learning的Target选择是绝对的greedy策略，保证了Agent在Q值进入收敛后不会记录可能掉入悬崖的状态动作的Q值，而Sarsa的 $\epsilon - greedy$ 的target选择策略在收敛后仍受cliif影响，因此需要远离。

- reward收敛对比：



这里我们看到Sarsa的探索性使得其收敛速度慢于Q-learning，但是Q-Learning也由于选择接近Cliff的路而收敛于较小的reward累计，这是由于算法决定的

- 平均Q值heatmap（left: Sarsa, right: Q-learning）



平均Q值的heatmap并看不出算法的差别，智能判断出隆重算法的Qtable都认定了Cliff周围是危险的

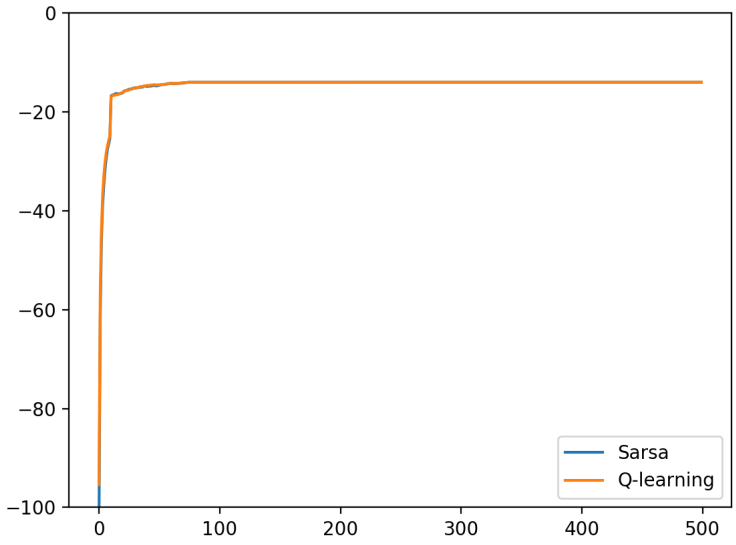
Condition of $\epsilon = 0$

- 最优路径对比：

```
optimal travel path by Sara:
0   0   0   0   0   0   0   0   0   0   0   0
0   0   0   0   0   0   0   0   0   0   0   0
→   →   →   →   →   →   →   →   →   →   →   ↓
↑   0   0   0   0   0   0   0   0   0   0   G
optimal travel path by Q-Learning:
0   0   0   0   0   0   0   0   0   0   0   0
0   0   0   0   0   0   0   0   0   0   0   0
→   →   →   →   →   →   →   →   →   →   →   ↓
↑   0   0   0   0   0   0   0   0   0   0   G
```

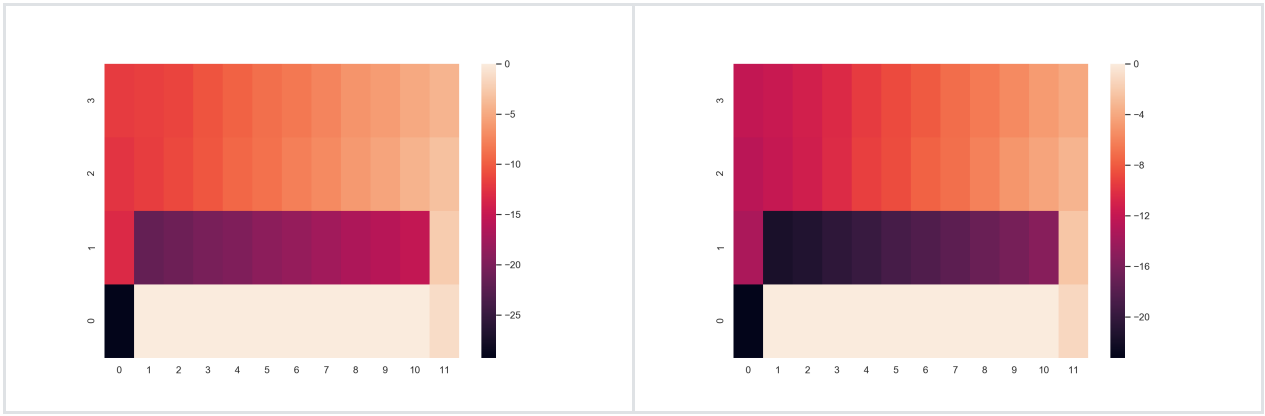
这里我们看到 $\epsilon = 0$ 时两种算法最优路径一致，这时Sarsa收敛时不再受 $\epsilon - greedy$ 影响， 本质上两种算法都退化为TD(0)

- reward收敛对比：



这里我们看到 $\epsilon = 0$ 时两种算法收敛情况一致，这说明两种算法都退化为TD(0)的结论是正确的

- 平均Q值heatmap（left: Sarsa, right: Q-learning）



平均Q值的heatmap并看不出算法的差别，智能判断出隆重算法的Qtable都认定了Cliff周围是危险的