RL Assignment1

Requirements

Use python3

Implement iterative policy evaluation methods and policy iteration methods. Then run the two methods to evaluate and improve an uniform random policy $\pi(n|\cdot) = \pi(e|\cdot) = \pi(s|\cdot) = \pi(w|\cdot) = 0.25$

My Implementation

Policy Iteration

Modify the policy according to the estimation of status value.

First make a policy evaluation by setting theta as 0.01.

For each policy we can go to east, west, south, or north. Calculate the value of each and then add them up

Then do the policy improvement by finding the policy that brings the maximum reward.

If the policy for all grids is the same as what is done before, assert that it reaches a policy-stable state.

Value Iteration

Make choice according to status value directly.

For each move, make the best choice which has the largest V.

Do this until the value is convergence.

Result

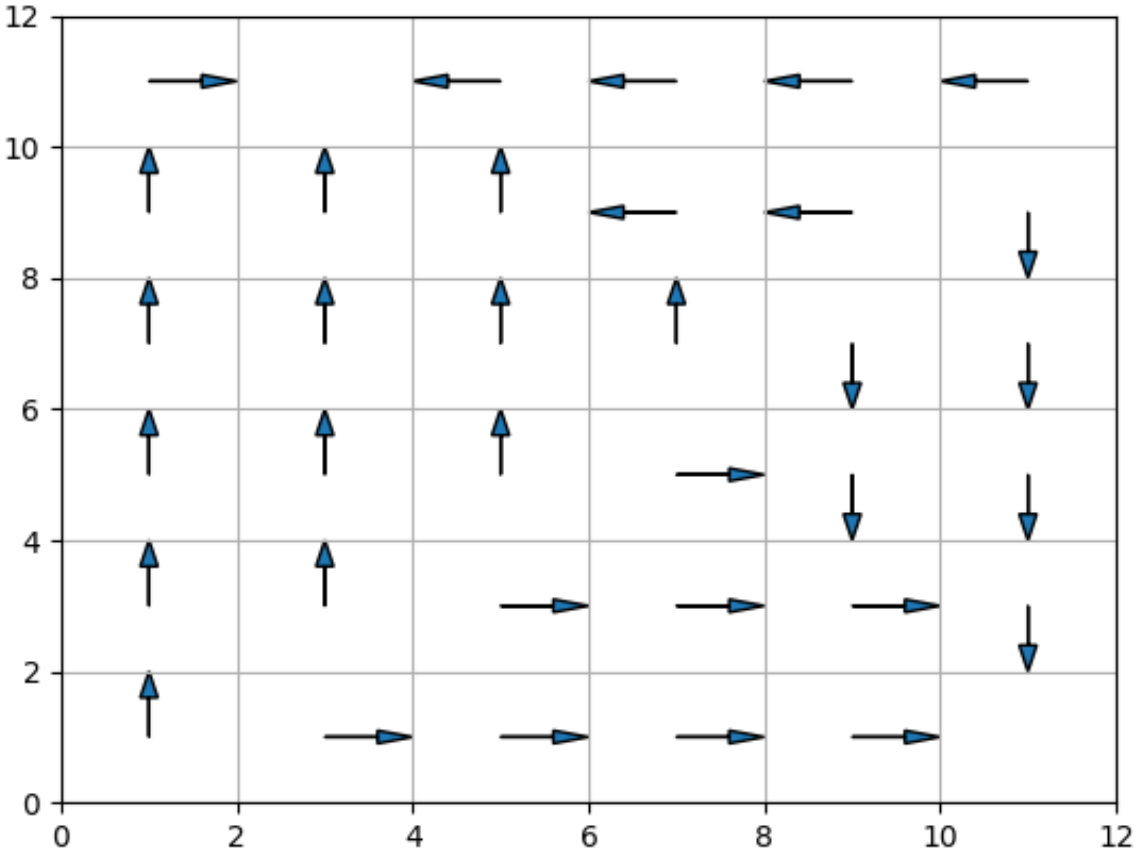
Policy Iteration Result

-18.10	**0**	**-29.10**	**-43.88**	**-51.34**	**-54.45**
-32.21	-30.04	-39.44	-47.22	-51.72	-53.58
-44.49	-44.55	-47.39	-49.86	-50.76	-50.59
-52.74	-52.29	-51.74	-50.07	-46.87	-43.45
-57.46	-56.14	-53.23	-47.83	-39.23	-28.89
-59.53	-57.62	-53.21	-44.79	-29.34	0

`print in python`

```
[[-18.09781267   0.         -29.09982712 -43.8806671  -51.34356824
  -54.4513566 ]
 [-32.20578337 -30.04490178 -39.4357903  -47.22023165 -51.72265995
  -53.58389497]
 [-44.49097563 -44.55067258 -47.39227447 -49.85680577 -50.75826788
  -50.5929254 ]
 [-52.73856165 -52.28960749 -51.74081355 -50.07102149 -46.87456219
  -43.44992603]
 [-57.46071582 -56.1447437  -53.22548399 -47.82533049 -39.23044629
  -28.89217365]
 [-59.52631134 -57.62000088 -53.20577739 -44.78610735 -29.33753095
    0.        ]]
```

Final state



the blank areas are the terminal states.

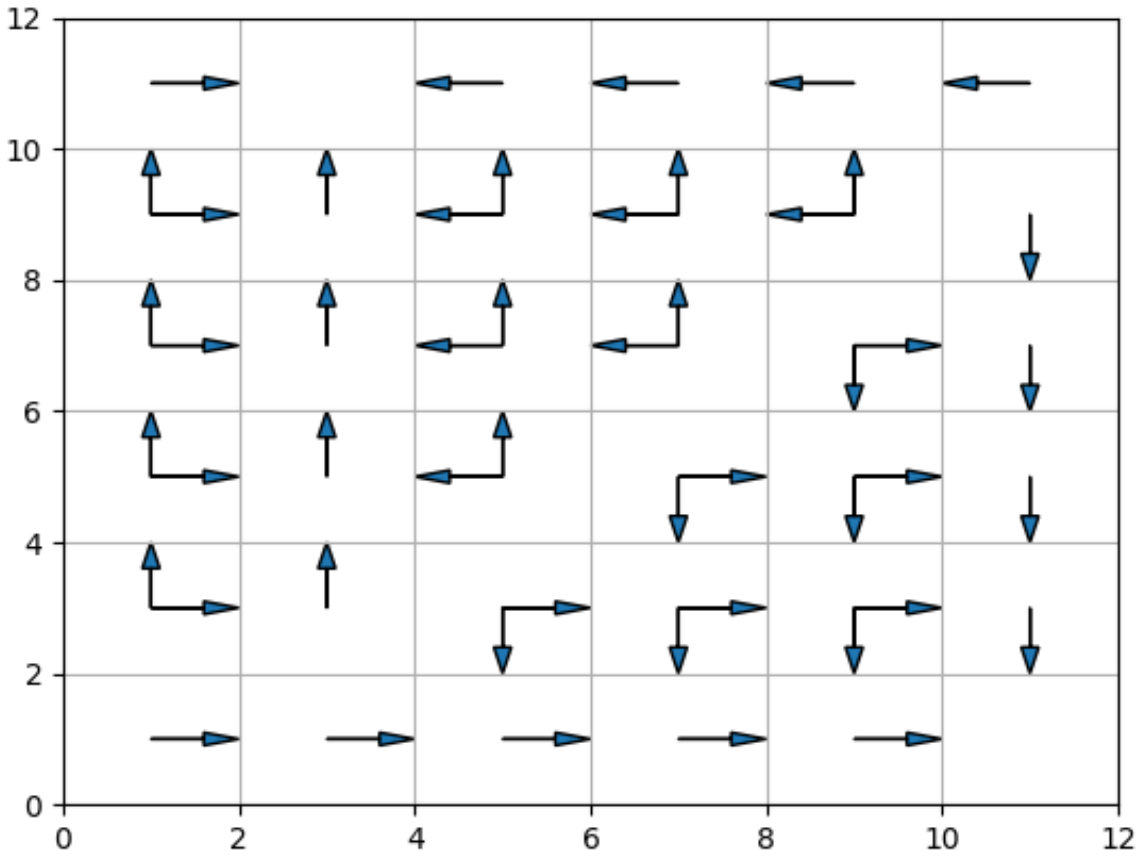
Value Iteration Result

-0.25	**0**	**-0.25**	**-0.31**	**-0.33**	**-0.33**
-0.31	-0.25	-0.31	-0.33	-0.33	-0.33
-0.33	-0.31	-0.33	-0.33	-0.33	-0.33
-0.33	-0.33	-0.33	-0.33	-0.33	-0.31
-0.33	-0.33	-0.33	-0.33	-0.31	-0.25
-0.33	-0.33	-0.33	-0.31	-0.25	0

`print in python`

```
[[-0.25       0.        -0.25      -0.3125    -0.328125  -0.33203125]
 [-0.3125    -0.25      -0.3125    -0.328125  -0.33203125 -0.33203125]
 [-0.328125  -0.3125    -0.328125  -0.33203125 -0.33203125 -0.328125  ]
 [-0.33203125 -0.328125  -0.33203125 -0.33203125 -0.328125  -0.3125    ]
 [-0.33300781 -0.33203125 -0.33203125 -0.328125  -0.3125    -0.25      ]
 [-0.33300781 -0.33203125 -0.328125  -0.3125    -0.25       0.       ]]
```

Final State



Summary and thinking

Policy Iteration is simple. While doing Value Iteration, I met some problems. Setting different theta may turn to different results. Because I do policy iteration first, setting a small number makes it do much steps to reach convergence. So I set it to 0.01 at first. However, in value iteration it just need several steps to reach this state. Therefore, it's not convergence and can't be seen as a final state. After that, I make the parameter smaller to 0.00001 to get the result.