Homework 6

Collaborators:
Name: Zhang Youchao
Student ID: 3170100125

Problem 6-1. A Walk Through Reinforcement Learning

In this problem, you will implement some reinforcement learning algorithms, including Q-learning with table and Q-learning with approximators. You will also get touch with some popular RL techniques and tricks.

Here we use the gym benchmark to do experiments. It provides some handful environments to play with the agents/algorithms you design.

Skeleton code *run.ipynb* are provided for your convenience. Please see the documentation and comments in the notebook for more details. Also, please report critical results in this report. **It should be made that we can judge your assignment without referring to your code (though we may check your code).**

(a) Please implement the Q-Learning algorithm with a look-up table.

Answer:
(I)
Before use a exponential decay strategy to decay the exploration probability.

```
Average reward is 0.28, average step is 7.49
[[0.         0.         0.77378094 0.        ]
 [0.         0.         0.81450625 0.        ]
 [0.         0.857375   0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.9025     0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.46208    0.        ]
 [0.         0.95       0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         1.         0.        ]
 [0.         0.         0.         0.        ]]
```

Figure 1: Without exponential decay

After use a exponential decay strategy to decay the exploration probability.
$epsilon = 1.0$

$min\_epsilon = 0.01$
$epsilon\_decay = 0.9$

```
Average reward is 0.945, average step is 6.073
[[0.73485666 0.77378094 0.         0.73485666]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.77254289 0.81450625 0.         0.72921116]
 [0.         0.         0.         0.        ]
 [0.         0.722      0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.81449582 0.         0.857375   0.7428297 ]
 [0.81449582 0.81448415 0.9025     0.        ]
 [0.82308    0.95       0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.        ]
 [0.         0.         0.         0.85737491]
 [0.77567059 0.949696   1.         0.90244224]
 [0.         0.         0.         0.        ]]
```

Figure 2: Use exponential decay

After training, we can now use the qtable for our MDP.
Average reward is 1.0, average step is 6.0
The slipper problem:
$epsilon = 1.0$
$min\_epsilon = 0.01$
$epsilon\_decay = 0.9$

```
Average reward is 0.6728, average step is 44.1037
[[1.41676015e-01 4.30007586e-03 3.72908044e-02 4.27381936e-03]
 [1.85350824e-03 8.37007459e-04 2.87462394e-04 1.14059023e-01]
 [6.73353612e-04 8.85947733e-04 1.09155585e-03 6.56842886e-02]
 [7.35648208e-04 5.85592198e-04 1.63048148e-03 4.80749941e-02]
 [2.97188248e-01 3.32912805e-03 1.52690005e-03 6.61770090e-04]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [2.70044484e-03 1.57309724e-06 0.00000000e+00 1.91379655e-06]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [1.72723451e-03 9.01201691e-03 1.51495384e-03 5.46524614e-01]
 [8.71972610e-04 5.02576123e-01 6.24059205e-04 2.19131608e-03]
 [8.50283479e-01 1.56176668e-04 1.01230134e-04 1.32161821e-04]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]
 [2.31220417e-02 3.19216591e-02 8.71016052e-01 1.70408113e-03]
 [7.60416523e-02 9.96834470e-01 5.83686294e-02 7.42232259e-02]
 [0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00]]
```

Figure 3: slipper condition

Average reward is 0.744, average step is 44.61

(II)
The phenomenon is that:Keep forward is always greater than backward.

Foward:Average reward is 354.8748748748749
Backward:Average reward is 160.17017017017017
Explain:
First,I think the backward's high probability event is going back and every return reward is 2, so if we forget the forward situation the average reward can be somehow calculated as $100 * 2 * (0.8) = 160$.
Second,I think the Foward's high probability event is going forward. I think once get end,the high probability is still moving on,so the reward 10 can be executed with high probability.Although there is a probability of 0.2 to return to the initial point, because the probability of going forward is the greatest, it takes a few steps to reach the end point (these steps can be considered as a delayed reward). Once the end point is reached, it is the same situation as the previous step.
So on the whole, if keep going forward, the probability of getting 10 reward is higher,if keep going backward, the probability of getting 2 reward is higher. So it finally shows forward>backward.

(b) (Optional, Homework Bonus) Please implement the Q-Learning algorithm with an approximator with the *CartPole* environmant.

Answer: