Comp341 Assignment 5
Burak Yıldırım 72849

Reference code: https://github.com/WilliamLambertCN/CS188-Homework/tree/master/PJ3_reinforcement

Q1)
Markov Decision Process makes our agents deterministic. They are called reflex agents because process agent isn't stochastic, and it gives exact decisions. The procedure of value iteration is called as offline planning because it doesn't learn by trying in real time instead it assigns values for each location and start taking actions according to these values.

Q2)
The noise parameter is decreased from default value of 0.2 to 0.01 because it refers to the frequency of checking the whole map. The value is lowered because checking the whole map is costly and it shouldn't be checked if it isn't really necessary.

Q3)
3a:
-Discount: 0.1
-Noise: 0
-Living Reward: -1
In this part the discount factor is selected low because we need to get to the shortest exit without considering the longer paths that includes exploration steps.

3b:
-Discount: 0.1
-Noise: 0.1
-Living Reward: 0.5
In this part additionally a noise of 0.1 and a living reward of 0.5 is given for the agent to not to consider cliffs and instead prefer the top paths.

3c:
-Discount: 0.9
-Noise: 0
-Living Reward: -1
In this part the discount factor is increased to 0.9 to encourage the agent take the last exit. The noise is selected as 0 and living reward is -1 for agent to pursue a path near cliff.

3d:
-Discount: 0.9
-Noise: 0.2
-Living Reward: 0
In this part the discount factor is increased to 0.9 to encourage the agent take the last exit. The noise is selected as 0.2 and living reward is 0 for agent to pursue a long path.

3e:
-Discount: 0
-Noise: 0.1
-Living Reward: 10
In this part the noise is set to 0.1 and living reward is 10 to make the agent live as much as it can and not taking an exit.

Q4)
In the first command, it executes a value iteration so that it calculates the path beforehand as being offline learning. In the second command, it executes a q-value learning so that the agent can learn by trying actions in real time. Once the agent takes an action, the values change dynamically, and it tries to find the optimal action from the current state. On the other hand, value iteration has constant values for each state, and it doesn't consider the optimal action from a current state.

Q5)
There isn't any learning rate and epsilon values to find the optimal policy will be learned after 50 iterations because the agent isn't taking the optimal actions even though I have tried many possible values for learning rate and epsilon. In most scenarios, the agent keeps switching between two states and when it switched to a different state in the end it can't find an optimal policy.

Q6)
In order to find an optimal policy, q-learning doesn't work well on a large grid because a high number of training episodes are needed to achieve an optimal policy. The exploration process takes a lot of time which can be around infinity for very large grids. As a result, the tabularized q-learning doesn't work for large grids.

Q7)
The behavior of pacman depends on several features. Firstly, it checks for the closest food and tries to reach a food by taking minimum number of steps. The priority of the pacman shifts when there is a ghost nearby. In such case, avoiding the ghost became the pacman's top priority while also having food as a second priority. For instance, if the distance between pacman and food equals to the distance between the pacman and ghost. The pacman will avoid the ghost rather than taking the action of having the food. The behavior of pacman can be observed exactly in this attitude.