Reward System & Q-learning

Deadline: 15 Aban

Assignment 1

1 Reward Modification

Consider a tabular Markov Decision Process (MDP) with $0 < \gamma < 1$, and no terminal states. The agent will act indefinitely. The original optimal value function for this problem is denoted as V_1, and the optimal policy is π_1.

a Now, suppose someone introduces a slight reward adjustment of c to all transitions in the MDP. Describe the expression for the new optimal value function. Can the optimal policy in this altered scenario change? Explain the reasons behind your answer.

b Instead of introducing a small reward bonus, someone decides to scale all rewards by an arbitrary real constant $c \in \mathbb{R}$. Are there instances where the new optimal policy remains π_1, and the resulting value function can be expressed as a function of c and V_1? If such cases exist, provide the resulting expression and elucidate under what conditions they occur. Conversely, explain when the optimal policy might change and why. Is there a choice of c for which all policies are optimal?

c Suppose the MDP now includes terminal states that can end an episode. Does the presence of terminal states affect your response to part (a)? If yes, offer an example MDP where your responses to part (a) and this part would diverge.

2 Pacman

In this part, you will learn how to solve the Pacman game with Q-learning. In this problem, an environment is given to the program. The environment consists of Agents, Dots, Walls, and Ghosts(Adding ghosts will have bonus points and it is optional). To discretize and digitize the roads, we convert them into small unit squares. The starting point of the agent's movement is also determined. The goal is to collect all the Dots in the environment without touching any Ghosts(Of course you can not pass the walls). In the figure below, the ghosts are marked in red. W stands for wall, A stands for Agent, D stands for Dot, G stands for Ghost, and E for empty cell(A cell becomes empty after the agent collects the dot).

A	D	D	D	D	D	D	D	D
D	W	W	W	D	W	W	W	D
D	W	D	D	D	D	D	W	D
D	D	D	W	E	W	D	D	D
D	W	D	W	G	W	D	W	D
D	W	D	D	W	D	D	W	D
D	D	D	D	D	D	D	D	D

Tabel 1: An example of a Pacman board

1. Explain what the number of states depends on. Can the number of states be reduced?

2. Define Action, State, Rewards, and Goal State for the problem and provide your explanation.

3. First, consider the environment as shown above. Analyze the effect of γ for at least 3 gamma values equal to 0.25, 0.5, and 1. Also, for at least 3 α values, analyze the results and determine the impact of α.

4. Display the environment as a weighted graph. The vertices of different modes and their weights are the amount of Reward after applying the corresponding Action. You can mention the corresponding Action as a Label on each edge.

5. Draw the Q-Table for this problem.

6. Test your code on another arbitrary environment and report the result.

Bonus Parts:

7. Displaying the episodes and rewards resulting from various actions in each state graphically (image or animation) will be a plus point and optional.

8. In the original part of the Pacman assignment, you don't have to add a ghost in the environment, but if you do so, it is an extra point and optional. (You can use random walk for the ghost.)