CSC3022H: Machine Learning

Assignment 6

Reinforcement Learning

Department of Computer Science
University of Cape Town

Due: Friday, 29th May, 2020, 5.00 PM

Problem Description

Implement (in C++) the *Value Iteration* algorithm (detailed in chapter 3 [Sutton and Barto, 1998] and chapter 13 [Mitchell, 1997]) in order to find the optimal value (V^*) for each state in a small grid-world (figure 1). Use the following information:

1. The agent has 4 actions { *left, right, up, down* }, and the grid-world 6 states { s_1, s_2, s_3, s_4, s_5, s_6 }. Figure 1 shows the possible transitions between states (actions for given states).
2. The state transition distribution $P^a_{ss'}$ is deterministic, so $P^a_{ss'} = 1.0$ for all states and actions.
3. Rewards for all state transitions are zero ($R^a_{ss'} = 0$), except the following:

   $(1, 1) \rightarrow (2, 1)$; $R^a_{ss'}$=50

   $(2, 0) \rightarrow (2, 1)$; $R^a_{ss'}$=100
4. State s_3 is the terminal state.
5. The discount factor is 0.8, **i.e.** $\gamma = 0.8$.

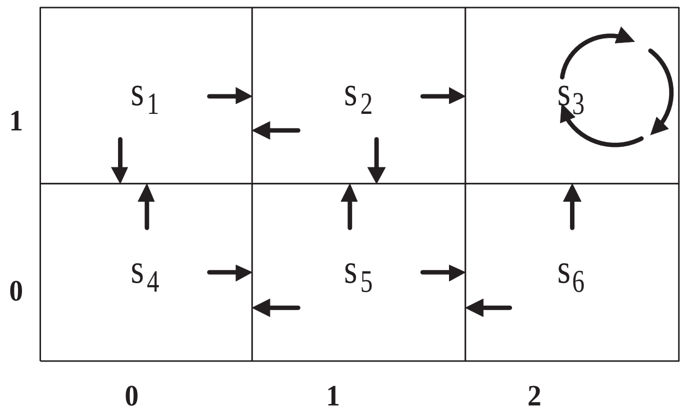

Figure 1: A small grid-world, where arrows show possible transitions between states. Note that state s_3 is a terminal state.

Question 1: How many iterations does it take for the *Value Iteration* algorithm to converge? In an output text file list the optimal values (V^* for each state).

Question 2: Assume we start in state s_1, give the states that form the optimal policy (π^*) to reach the terminal state (s_3).

Question 3: Is it possible to change the reward function function so that V^* changes, but the optimal policy (π^*) remains unchanged?

If yes, describe how the reward function must be changed and the resulting change to V^*. Otherwise, briefly explain why this is impossible.

In a ZIP file, place the source code, makefile, and output text file (answers to questions 1, 2, 3).

Alongside your archive, please submit a file containing the hash, of your archive, produced by a MD5 checksum. See the link below for a tutorial on how to produce the hash:
https://www.tecmint.com/generate-verify-check-files-md5-checksum-linux/

Failure to include a '.md5' file waives your right to appeal your mark in relation to issues that arise from corrupted, missing or incorrect submissions.

References

[Mitchell, 1997] Mitchell, T. (1997). *Machine Learning: Chapter 13: Reinforcement Learning.* McGraw Hill, New York, USA.

[Sutton and Barto, 1998] Sutton, R. and Barto, A. (1998). *An Introduction to Reinforcement Learning (Chapter 3).* John Wiley and Sons, Cambridge, USA.