Homework Assignment # 3

Due: Thursday, March 16, 2017, 11:59 p.m.

Total marks: 100

Question 1. [5 MARKS]

Exercise 6.2 in Sutton and Barto 2nd Ed., 2016. [Answer each of the three parts to this question]. Think of the transitions and rewards that could produce this update to V(A). The third part is looking for a numerical answer.

Question 2. [5 MARKS]

Exercise 6.3 in Sutton and Barto 2nd Ed., 2016. Start thinking about what might happen with α's larger and smaller than those tested. Also Figure 2.6 might give you some inspiration of how we usually expect α to behave.

Question 3. [5 MARKS]

Exercise 6.5 in Sutton and Barto 2nd Ed., 2016.

Question 4. [55 MARKS]

Programming question

[**Part one: worth 45**] Program a windy Gridworld with King's Moves. Re-solve the windy gridworld task assuming eight possible actions, including the diagonal moves, rather than the usual four. As in Example 6.5 we will use $Q_0(s, a) = 0$ for all s, a, and $\epsilon = 0.1$ and $\alpha = 0.5$. You will implement two agents and compare their performance: (1) Sarsa as described on page 138, and (2) expected Sarsa as described on page 142.

Plot the number of steps to goal over the first 100 episodes (in RL-glue you can use RL_episode() function), averaged over 100 runs of the experiment. This is called a *learning curve*. In RL-glue we can access the number of steps taken in an episode by calling RL_num_steps(). RL_episode can be given an integer **k** as input. If **k**=0, then episodes are run until the environment program signals termination. Otherwise, RL_episode(**k**), terminates episodes after **k** steps. If you use $\mathbf{k} \neq 0$, make sure **k** is very large otherwise it will make your results worse. You should submit one plot with two lines—one for each of the two agents—or two separate plots.

This will require you to implement several things:

1. a simulation of the windy gridwold problem, as described in Example 6.5, except with 8 actions (King's moves) (an Environment program)
2. Sarsa, and Expected Sarsa (Agent programs)
3. code to run the experiment for 100 episodes, averaging over 100 runs (Experiment program)

Please submit your plots and ALL your code (including any scripts and data processing).

[**Part two: worth 10**] Experiment with different α and ϵ values for one of the agents (either Sarsa or Expected Sarsa). Discuss how changing α and ϵ effects the agent's learning performance. Please include graphs to help with your explanation.

Question 5. [5 MARKS]

Exercise 6.9 in Sutton and Barto 2nd Ed., 2016.

Question 6. [5 MARKS]

Exercise 7.1 in Sutton and Barto 2nd Ed., 2016.

Question 7. [20 MARKS]

Programming question:

Resolve the windy gridworld with King's moves from Question #4, this time using n-step Sarsa (page 157). The environment and experiment program will be the same. However, you will implement one new agent: n-step Sarsa. Test n values of 1, 5, 10, and 20. Plot the number of steps to goal over the first 100 episodes, averaged over 100 runs of the experiment. Your plot should have four lines (learning curves), one for each value of n.

You may have to experiment with different initializations of the value function Q_0, different exploration rates ϵ, and different learning rates α.

Please submit your plots and ALL your code (including any scripts and data processing).

Homework policies:

Your assignment will be submitted as a single pdf document and a zip file with code, on canvas. The questions must be typed; for example, in Latex, Microsoft Word, Lyx, etc. or must be written legibly and scanned. Images may be scanned and inserted into the document if it is too complicated to draw them properly. All code (if applicable) should be turned in when you submit your assignment. Use the RL-glue framework available on the course webpage (your code will be in c/c++), and any language of choice for plotting the results (learning curves).

Policy for late submission assignments: Unless there are legitimate circumstances, late assignments will be accepted up to 5 days after the due date and graded using the following rule:

on time: your score 1

1 day late: your score 0.9

2 days late: your score 0.7

3 days late: your score 0.5

4 days late: your score 0.3

5 days late: your score 0.1

For example, this means that if you submit 3 days late and get 80 points for your answers, your total number of points will be $80 \times 0.5 = 40$ points.

All assignments are individual work, no exceptions. All the sources used for problem solution must be acknowledged, e.g. web sites, books, research papers, personal communication with people, etc. You are expected to solve problems from scratch. That means, if you are asked to code something, do not find code online and modify it. Write it from scratch yourself. Academic honesty is taken seriously; for detailed information see Indiana University Code of Student Rights, Responsibilities, and Conduct.

Good luck!