CS512– Artificial Intelligence

Instructor : Shashi Shekhar Jha (shashi@iitrpr.ac.in)

Assignment - 4 | Due on 17/05/2020 2400 Hrs (50 Marks)

Submission Instructions:

All submission is through google classroom in one zip file. In case you face any trouble with the submission, please contact the TAs:

- Armaan Garg, 2019CSZ0002@iitrpr.ac.in
- Rahul Kumar Rai, 2018csz0004@iitrpr.ac.in

Your submission must be your original work. Do not indulge in any kind of plagiarism or copying. Abide by the honour and integrity code to do your assignment.

As mentioned in the class, late submissions will attract penalties.

Penalty Policy: There will be a penalty of 20% for every 24 hr delay in the submission.
E.g. For the 1st 24 hr delay the penalty will be 20%, for submission with a delay of >24 hr and < 48 hr, the penalty will be 40% and so on.

You submission must include:

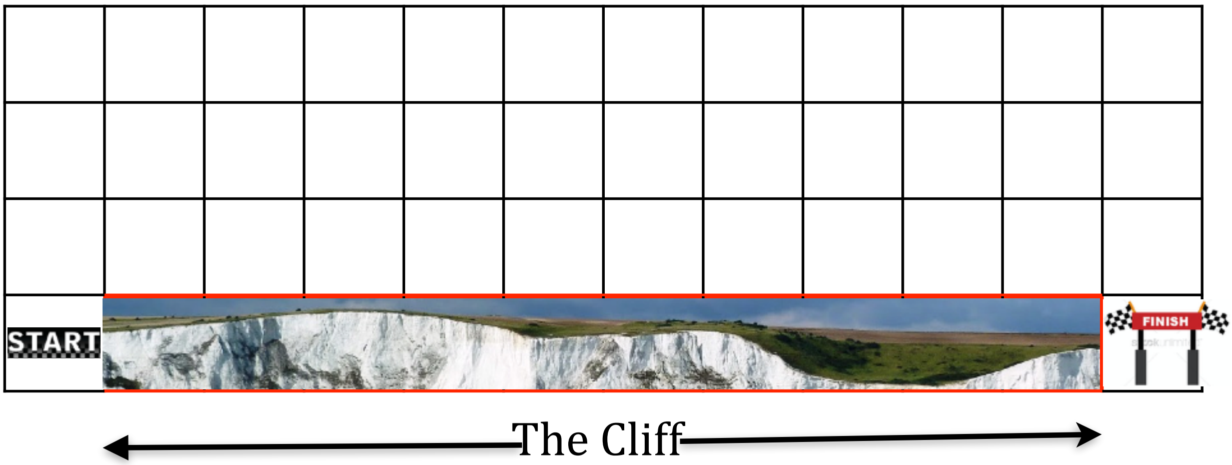
- A legible PDF document with all your answers to the assignment problems, stating the reasoning and output.
- A folder named 'code' containing the scripts for the assignment along with the other necessary files to run yourcode.
- A README file explaining how to execute your code.

Naming Convention:

Name the ZIP file submission as follows:
Name_rollnumber_Assignmentnumber.zip
E.g. if your name is ABC, roll number is 2017csx1234 and submission is for lab1 then you should name the zip file as: ABC_2017csx1234_lab1.zip

Assignment: Cliff Walking

Consider the grid-world shown below:



This grid world has episodic tasks, with start and goal states, and the usual actions causing movement up, down, right, and left.

Reward is -1 on all transitions except those into the region marked “The Cliff”. Stepping into this region incurs a reward of -100 and sends the agent instantly back to the start.

Consider two reinforcement learning algorithms viz. Q-learning and SARSA and the ϵ-greedy policy:

Q-Learning:

$$Q(s,a) = Q(s,a) + \alpha\{R(s,a) + \gamma \max_{a'} Q(s',a') - Q(s,a)\}$$

SARSA (full form - **S**tate **A**ction **R**eward **S**tate **A**ction):

$$Q(s,a) = Q(s,a) + \alpha\{R(s,a) + \gamma Q(s',a') - Q(s,a)\}$$

ϵ-greedy policy:

$$a* = \underset{a \in A}{argmax}\; Q(s,a)$$

$$\pi(a\,|\,s) = \begin{cases} 1 - \epsilon + \dfrac{\epsilon}{|A|} & if\ a = a* \\ \dfrac{\epsilon}{|A|} & if\ a \neq a* \end{cases}$$

The ϵ-greedy policy is a stochastic policy that has a probability distribution over all the actions (A) in a state. The action at the goal state is Exit action which results in a Reward of 0.

Find the optimal policies generated by the two algorithms - Q-Learning and SARSA under the following parameterization:

1. Undiscounted rewards, $\epsilon = 0.1$, $\alpha = 0.5$ [10 points]
2. $\gamma = 0.9$, $\epsilon = 0.1$, $\alpha = 0.5$ [10 points]
3. $\gamma = 0.9$, $\alpha = 0.5$ and ϵ is decreased to 0 from 0.1 with time [15 points]

Q.1. For each case of parameterization, plot the "Sum of rewards in episode” versus “Episode #” for Q-Learning and SARSA algorithms. [3 points]

Q.2. For each case of parameterization, plot the optimal policy obtained using the Q-Learning and SARSA algorithm. [2 points]

Q.3. Based on your experimentation, which one has better online performance - Q-Learning or SARSA and why? [5 points]

Q.4. Suppose action selection is greedy. Is Q-learning then exactly the same algorithm as SARSA? Will they make exactly the same action selections and weight updates? Supplement your answer with results from your implementation. [5 points]