Programming Assignment-3

Topics in ML (CSE975-Monsoon 2017)
Submission Deadline : 14/11/2017 11.55 PM

November 6, 2017

General Instructions

1. Assignment can be implemented in Matlab/Octave, Python, C/C++, R.
2. Ensure that submitted assignment is your original work. Please do not copy any part from any source including your friends, seniors and/or the internet. If any such attempt is caught then serious actions including an F grade in the course is possible.
3. A single zip file needs to be uploaded to the moodle course portal. The file should contain (1)pdf report file containing the experiment findings (2)the code you have written. Include the assignment number, your name and roll number at the top of the first page of the pdf report.
4. Your grade will depend on the correctness of answers and output. In addition, due consideration will be given to the clarity and details of your answers and the legibility and structure of your code.
5. Assignments submitted through moodle.iiit.ac.in only will be considered for evaluation. No assignment submitted through the mail will be considered.
6. There will not be any deadline extension given.
7. Please make sure that you submit the assignment well ahead of the time to avoid last minute power/internet issues.

Problem 1. [Marks: 5] Consider the Gridworld example as shown in Figure 1. An agent starting in the start state S must reach the goal state G. At each time step, the agent can *go up, down, left or right*. However, the agents movements are a bit noisy since it goes in the intended direction with a high probability a and in one of the two lateral directions with a low probability b. For instance, when executing the action up, the agent will indeed go up by one square with probability a, but may go left with probability b and right with probability b (here $a+b+b=1$). Similarly, when executing the action left, the agent will indeed go left with probability a, but may go up with probability b and down with probability b, When an action takes the agent out of the grid world, the agent simply bounces off the wall and

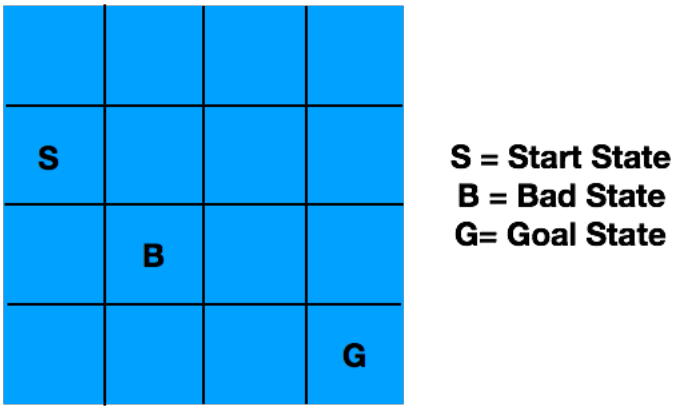

Figure 1: Problem1

stays in its current location. For example, when the agent executes left in the start state it stays in the start state with probability a, it goes up with probability b and down with probability b. Similarly, when the agent executes up from the start state, it goes up with probability a, right with probability b and stays in the start state with probability b. Finally, when the agent is in the goal state, the task is over and the agent transitions to a special end state with probability 1 (for any action). This end state is absorbing, meaning that the agent cannot get out of the end state (i.e., it stays in the end state with probability 1 for every action). The agent receives a reward of 100 when it reaches the goal state (G), -70 for the bad state (B) and -1 for every other state, except the Start (S) state, which has a 0 reward. The agents task is to find a policy to reach the goal state as quickly as possible, while avoiding the bad state.

Simulate SARSA, Q-Learning, Expected SARSA for this problem. Use a discount factor (γ) of 0.99, a transition model with $a = 0.9$ and $b = 0.05$ and a learning rate of $\alpha = 1/N(s, a)$ where $N(s, a)$ is the number of times that action a was executed in state s.

Always starting from the *start* state, SARSA, Q-learning and Expected SARSA for 10,000 episodes, where an episode consists of a sequence of moves from the start state until the end state is reached. Try two different ϵ-greedy exploration functions by setting ϵ to 0.05 and then to 0.2. Report the following.

1. The optimal policies and optimal value functions found for $\epsilon = 0.05$ and $\epsilon = 0.2$ using SARSA. Discuss the impact of ϵ on the convergence of SARSA. More specifically, discuss the impact on the rate of the convergence and the policy that it will eventually converge to. Plot the reward per episode every time an episode ends.

2. The optimal policies and optimal value functions found for $\epsilon = 0.05$ and $\epsilon = 0.2$ using Expected SARSA. Discuss the impact of ϵ on the convergence of Expected SARSA. More specifically, discuss the impact on the rate of the convergence and the policy that it will eventually converge to. Plot the reward per episode every time an episode ends.

3. The optimal policies and optimal value functions found for $\epsilon = 0.05$ and $\epsilon = 0.2$ using Q-learning. Discuss the impact of ϵ on the convergence of Q-learning. More specifically,

discuss the impact on the rate of the convergence and the policy that it will eventually converge to. Plot the reward per episode every time an episode ends.