COMP532 Assignment 1 –
Reinforcement Learning

You need to solve each of the following problems. Problem 1 concerns an example/exercise from the book of Sutton and Barto (available in Vital). You must also include a brief report describing and discussing your solutions to the problems. Students can do the assignment in pairs.

- This assignment is worth 15% of the total mark for COMP532
- 80% of the assignment marks will be awarded for correctness of results.
- 20% of the assignment marks will be awarded for the quality of the accompanying report
- Students will do the assignment in groups
- We expect 3 students in one group (it would be fine to have groups of 1, 2, 4 and 5 as well, but it is suggested to have groups of 3), please find your team members on your own, we have the MS Teams "**COMP532-2021**" group to facilitate the group formation
- Only one **single** submission is needed for each group
- The **same marks** will be granted to all the members in the same group
- Please list all your group members (**names, emails, student ids**) in your submitted report

Submission Instructions

- Send all solutions as a single PDF document containing your answers, results, and discussion of the results. Attach the source code for the programming problems as separate files.
- Submit your solution via Canvas.
- **The deadline for this assignment 12/03/2021, 13:00**
- Penalties for late submission apply in accordance with departmental policy as set out in the student handbook, which can be found at
  http://intranet.csc.liv.ac.uk/student/msc-handbook.pdf
  and the University Code of Practice on Assessment, found at
  https://www.liverpool.ac.uk/media/livacuk/tqsd/code-of-practice-on-assessment/code_of_practice_on_assessment.pdf

Problem 1 (24 marks)

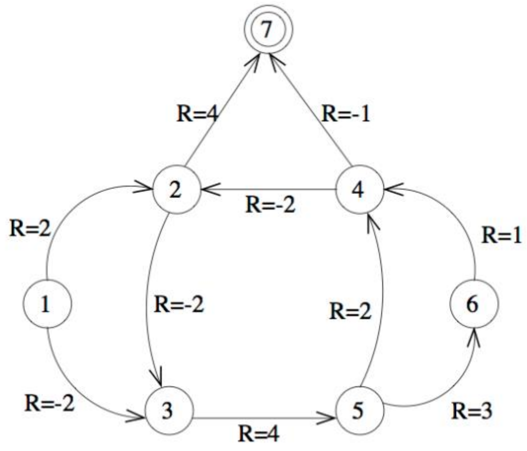
Re-implement in Python the results presented in Figure 2.2 of the Sutton & Barto book comparing a greedy method with two ε-greedy methods ($\varepsilon = 0.01$ and $\varepsilon = 0.1$), on the 10-armed testbed, and present your code and results. Include a discussion of the exploration - exploitation dilemma in relation to your findings.

Problem 2 (16 marks)

Consider a Markov Decision Process (MDP) with states $S = \{4,3,2,1,0\}$, where 4 is the starting state. In states $k \geq 1$ you can walk (W) and $T(k, W, k-1) = 1$. In states $k \geq 2$ you can also jump (J) and $T(k, J, k-2) = 3/4$ and $T(k, J, k) = 1/4$. State 0 is a terminal state. The reward $R(s, a, s') = (s - s')^2$ for all (s, a, s'). Use a discount of $\gamma = 1/2$. Compute both $V^*(2)$ and $Q^*(3, J)$. Clearly show how you computed these values.

Problem 3 (20 marks)

Consider the following Reinforcement Learning problem (the rewards R are tagged to the transitions, the transition probabilities are unknown) with states 1...7, of which state 7 is a terminal state. Let the initial values of all states be 0. Initialize the discount factor γ = 1. What are the values of all states (after each epoch) when Temporal Difference learning is used after the following episodes? The learning parameter α = 0.5 is fixed.
Episode 1: {1, 3, 5, 4, 2, 7}
Episode 2: {2, 3, 5, 6, 4, 7}
Episode 3: {5, 4, 2, 7}



Problem 4 (10 marks)

a) What does the Q-learning update rule look like in the case of a stateless or 1-state problem? Clarify your answer. (2 marks)
b) Discuss the main challenges that arise when moving from single- to multi-agent learning, in terms of the learning target and convergence. (3 marks)

Problem 5 (30 marks)

Re-implement in Python the results presented in Figure 6.4 of the Sutton & Barto book comparing SARSA and Q-learning in the cliff-walking task. Investigate the effect of choosing different values for the exploration parameter ε for both methods. Present your code and results. In your discussion clearly describe the main difference between SARSA and Q-learning in relation to your findings.

Note: For this problem, use $\alpha = 0.1$ and $\gamma = 1$ for both algorithms. The "smoothing" that is mentioned in the caption of Figure 6.4 is a result of 1) averaging over 10 runs, and 2) plotting a moving average over the last 10 episodes.