Multi-Agent Systems

RESIT Final Homework Assignment

MSc AI, VU

E.J. Pauwels

Version: February 9, 2023— **Deadline: Friday, 3 March 2023 (23h59)**

IMPORTANT

- This project is an **individual assignment.** So everyone should hand in their own copy.
- Both of the questions below require programming. In addition to the report (addressing the questions and discussing the results of your experiments), also provide the code on which your results are based (e.g. as Python notebooks). However, make sure that the report is self-contained, i.e. that it can be read and understood without the need to refer to the code. **Only the report will be marked, so it's NOT sufficient to submit only a notebook**.
- Store the pdf-report and the code in a zipped folder that you upload to canvas. **Use your name and VU ID as name for the zipped folder**, e.g.

  `Jane_Doe_1234567.zip`

- This assignment will be **graded.** The max score is 4 and will count towards your final grade.
- Your **final grade (on 10)** will be computed as follows:

  Assignments 1 thru 5 (max 1) + Individual assignment (max 4) + Final exam (max 5)

- Good luck!

1 Monte Carlo Simulation (30%)

The COVID scare was a wake-up call, highlighting the vulnerability of our hyper-connected modern world to the threat of pandemics. Now that the COVID urgency has passed, governments across the globe are rushing to update their emergency action plans and disaster scripts.

As part of this effort, you've been hired by a medical team that has been tasked with developing fast procedures to detect a blood-borne virus. Since these tests need to be administered to large groups in the population, and testing resources are limited, the medics have come up with the following procedure. They started from the assumption that they need to test N blood samples (of as many different individuals) and that N is large (e.g. $N = 10^6$). Furthermore, the probability that an individual is infected is p, where p is relatively small, e.g. $p < 0.1$.

Based on these assumptions they propose the following procedure to **minimise the number of tests** they have to run: Rather than testing each sample individually, take a batch of k samples and mix them. Then this mixed sample is tested for the presence of the viral antigen:

- If the mixed sample tests **negative** (i.e. no viral antigen is detected), then all the individual samples were clear, and you therefore have the result for all k individual samples that went into the batch.

- If the mixed sample tests **positive** (i.e. the viral antigen is present indicating infection), one needs to retest all individual samples that went into the batch, in order to find out which individual(s) are actually infected.

Questions

1. Use Monte Carlo simulation to estimate the optimal batch size k (i.e. the one that minimises the expected number of tests) for a given value of p where p can take values between 10^{-1} and 10^{-4}.

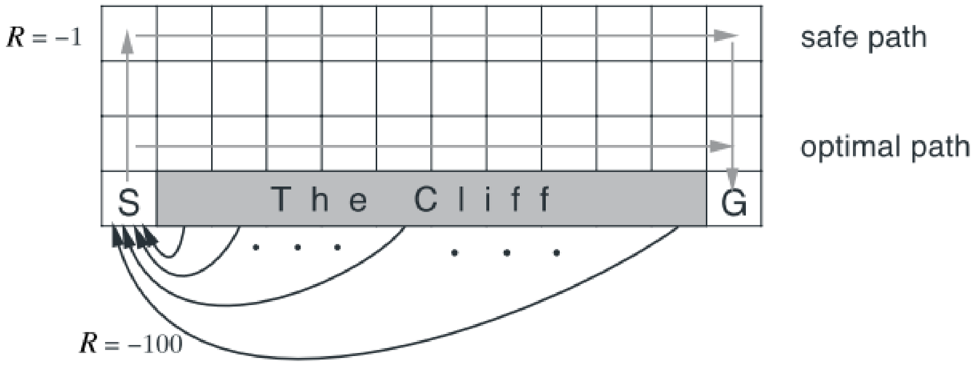
2. Quantify the expected reduction in workload (compared to testing all samples individually).

2 Reinforcement Learning: Cliff Walking (70%)

Consider the cliff-walking example (Sutton & Barto, ex. 6.6. p.108). Assume that the grid has 21 columns and 3 rows (above or in addition to the cliff). This is a standard undiscounted, episodic task, with start (S) and goal (G) states, and the usual actions causing movement up, down, right, and left. Reward is -1 on all transitions except:

- the transition to the terminal goal state (G) which has an associated reward of $+20$;

- transitions into the region marked *The Cliff*. Stepping into this region incurs a "reward" of -100 and also terminates the episode.



Questions

1. Use both SARSA and Q-Learning to construct an appropriate policy. Do you observe the difference between the SARSA and Q-learning policies mentioned in the text (safe versus optimal path)? For Q-learning, experiment with a replay-buffer. Discuss.

2. Try different values for ϵ (parameter for ϵ-greedy policy). How does the value of ϵ influence the result? Discuss.

3. Now assume that there is a snake pit in the cell on the top row and 11th column, i.e. right in the middle of the "safe path". Stumbling in this snake pit also carries a penalty (negative reward) of -100. Does this change the results obtained by SARSA or Q-learning? Discuss.