Exercise 2

Tommaso Brumani - 100481325
ELEC-E8125 - Reinforcement Learning

September 26, 2022

Task 1

The notebook code and '.pkl' file for this task can be found attached to the report submission, named 'value_iteration_T1.ipynb' and 'value_policy_T1.pkl' respectively.
By running the algorithm several times it can be observed that the sailor sometimes fails to reach the goal and instead "crashes" on the rock tiles, although this happens far less frequently than the sailor reaching the harbor.

Question 1.1

In this sailor gridworld, the sailor (and his boat) could be considered the agent, whereas the environment is the sea.

Question 1.2

The state value for the harbor and rocks is always zero because they are terminating states and thus, if the starting position was a rock or the harbor, the agent would gain 0 cumulated discounted reward before stopping.

Question 1.3

The sailor initially chose the "dangerous" path between the rocks, but decided to follow the "safe" path once the penalty for hitting the rocks was raised to -10.
This is likely because a higher penalty makes it more convenient for the agent to be averse to risk.

Task 2

The notebook code for this task can be found attached to the report submission, named ‘value_iteration_T2.ipynb‘.
The value function does not converge in 30 iterations with a threshold $\epsilon = 10^{-4}$.
Because of this it cannot be said for certain whether the policy has converged or not since, although it is possible that the computed policy at the 30th iteration is the same as an optimal policy by chance, it might still change an arbitrary number of times before the value function has converged.
This is because the policy is determined from the value function: it will in general converge when the value function converges, but it is possible for the algorithm to reach an optimal policy before then.

Task 3

The notebook code for this task can be found attached to the report submission, named ‘value_iteration_T3.ipynb‘.
When running the algorithm until convergence with a rock penalty equal to -2, the value iteration converges at the 43rd iteration.

Task 4

The notebook code for this task can be found attached to the report submission, named ‘value_iteration_T4.ipynb‘.
When evaluating the learned policy for 1000 episodes with -2 rock penalty and 100 iterations of the algorithm, the average and standard deviation of the discounted return are found to be respectively:

$$avg = 0.5168139630818775$$

$$std = 1.398906907652462$$

Question 4.1

Sutton and Barto [1] define the value function as ”functions of states [...] that estimate how good it is for the agent to be in a given state”, specifying that ”’how good’ here is defined in terms of future rewards that can be expected, or, to be precise, in terms of expected return”. From this, we can surmise that the value function for a given state corresponds to the expected return for that state.

Question 4.2

The value iteration approach cannot be used when dealing with an unknown environment because it requires knowledge of the transition probabilities and rewards matrices to calculate the optimal value function.
A Reinforcement Learning algorithm such as Monte-Carlo or Temporal Difference would be more suited for the task.

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction.* MIT press, 2018.