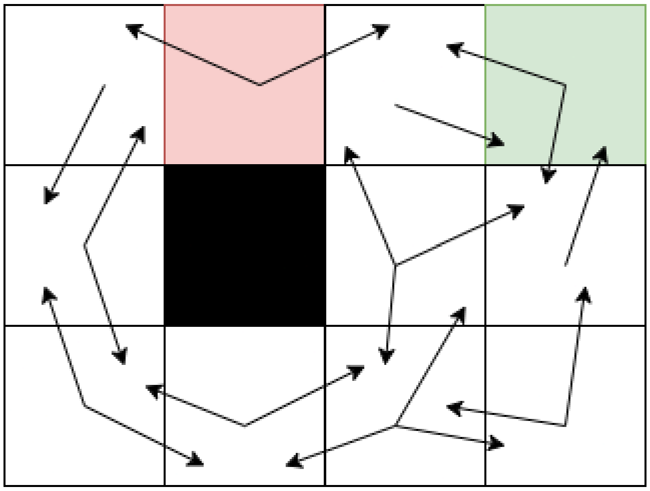
Github Link: https://github.com/juliusatgit/AdvancedDeepLearning

Task 1.1 - Optimal Policy π*

We have four directions: North, East, South and West



Task 1.2 – Value Function V

γ = 0.9, H = 100, P = 1

I am assuming that the reward is 0 on the white cells, as this is not clearly defined in the assignment sheet.

V*(4,3) = 1 (Terminal state)
V*(3,3) = max(0 + γ * V(4,3), 0 + γ * V(2,3), 0 + γ * V(3,2)) = max(0.9, 0, -0.9) = 0.9
V*(2,3) = -1 (Terminal State)
V*(3,1) = γ^2 * V(3,3) = 0.9^2 * 0.9 = 0.729 (optimal path is through {(3,1), (3,2), (3,3), (4,3)}
V*(1,1) = γ^2 * V(3,1) = 0.59049 (optimal path is {(1,1), (2,1), (3,1), (3,2), (3,3), (4,3)} – Note: {(1,1), (1,2), (1,3), (3,2)} would result in a negative value, so we dropped it

Task 1.3 -
P = 0.8 which means, we have to consider other possibilities as well:
γ = 0.7, H = 100

V*(4,3) = 1 (Terminal State)
V*(3,3) = γ * (0.8 * V*(4,3) + 0.1 * V*(3,2) + 0.1 * V*(2,3)) = 0.7 * (0.8 * 1 + 0.1 * 0 + 0.1 * -1) = 0.49

Task 2 – Questions

Task 2.1: Exploration vs. Exploitation Problem in RL

The exploration vs. exploitation problem refers to the trade-off between exploring new actions to gain more knowledge about the environment (exploration) and choosing known actions that yield high rewards (exploitation). Balancing both is essential for effective learning.

Task 2.2: Credit-Assignment Problem in RL

The credit-assignment problem is about identifying which past actions contributed to a specific outcome or reward. This is especially challenging when rewards are delayed, making it hard to determine the impact of individual decisions.

Task 2.3: What is the Markov Property?

The Markov property states that the future state depends only on the current state and action but not on the sequence of past states. A process with this property is called a Markov process.

Task 2.4: Does Chess Satisfy the Markov Property?

Formally, chess satisfies the Markov property if the state includes all necessary information like piece positions, castling rights, en passant possibilities, and whose turn it is. However, representing the full state accurately can be complex in practice.

Task 2.5: What is the Difference Between Q-Learning and Deep Q-Learning?

Q-learning uses a table to store Q-values for each state-action pair, which becomes impractical in large or continuous spaces. Deep Q-learning replaces the table with a neural network to approximate Q-values, making it suitable for more complex environments.