Lab 4

Rasmus Tengstedt

April 2024

1 Introduction

Since the terminal state equals 1, we have $V^*(4,3) = 1$.

To calculate the value of an adjacent tile, we use the following equation:

$$\begin{aligned} V^*(3,3) = {} & \text{Probability of successful transition} \times (\gamma \times \text{Value of next state}) \\ & + \text{Probability of unsuccessful transition} \times (\gamma \times \text{Value of alternative state}) \end{aligned}$$

Given that the Probability of successful transition is 1 and the Probability of unsuccessful transition is 0, we simplify this to:

$$V^*(3,3) = \gamma \times \text{Value of next state}$$

With the Value of the next state $V^*(4,3) = 1$, we have:

$$V^*(3,3) = 0.9 \times 1 = 0.9$$

We can similarly calculate the values for other tiles as follows:

2 Task 1.2

Given $\gamma = 0.9$:

$$V^*(3,3) = \gamma \times V^*(4,3) = 0.9 \quad \text{(the agent moves right from (3,3) to (4,3))}$$

$$V^*(2,3) = -1 \quad \text{(since we are at the end state)}$$

$$V^*(3,1) = \gamma^3 \times V^*(4,3) = 0.729 \quad \text{(the agent moves up from (3,1) to (3,2) and then right to (4,3))}$$

$$V^*(1,1) = \gamma^5 \times V^*(4,3) = 0.59 \quad \text{(the agent follows a longer path to (4,3))}$$

Summarizing the values:

$$\begin{aligned}
V^*(4,3) &= 1 \\
V^*(3,3) &= 0.9 \\
V^*(3,1) &= 0.729 \\
V^*(2,3) &= -1 \\
V^*(1,1) &= 0.59
\end{aligned}$$

3 Task 1.3

In task 1.3, the Probability of successful transition is 0.8 and the Probability of unsuccessful transition is 0.2. Thus, we have:

$$\begin{aligned}
V^*(3,3) = 0.8 \times (\gamma \times 1) \\
+ 0.2 \times (\gamma \times \text{Value of alternative state})
\end{aligned}$$

Without information about the alternative state, we calculate only the first part. Assuming the alternative state is the same as the current state, the value would be 0. Therefore:

The terminal state equals 1, $V^*(4,3) = 1$

$$V^*(3,3) = 0.8 \times \gamma \times V^*(4,3) = 0.8 \times 0.9 \times 1 = 0.72$$

4 Task 2

4.1 Task 2.1

Exploration focuses on finding new solutions that might be optimal, while exploitation aims to find the best possible solution in a given area. This is akin to finding the global min/max of a multi-variable function, avoiding local minima. In reinforcement learning (RL), the agent must find the optimal path using a combination of exploration and exploitation. One method is the ϵ-greedy strategy, where the agent selects the best-known action most of the time, but occasionally chooses an action at random with probability ϵ.

4.2 Task 2.2

The credit-assignment problem in RL involves determining which actions are responsible for a certain reward. Since rewards are often delayed, linked to a sequence of actions, it is challenging to identify the optimal set of actions. Solutions include adding a timer to update weights more frequently, reducing long gaps between updates.

4.3 Task 2.3

The Markov property states that only the current state is needed to determine the future state. The current state results from recursively processing states from the first to the current one, making historical states irrelevant once the current state is known.

4.4 Task 2.4

In a given board state, the history of states (how the pieces got to the state) is irrelevant. The optimal move is determined by the current situation, satisfying the Markov property.

4.5 Task 2.5

Q-learning and Deep Q-learning are RL algorithms for decision-making problems modeled as Markov Decision Processes (MDPs). Deep Q-learning extends Q-learning by using a neural network instead of a tabular representation and incorporating methods for handling replay.