**Project Two – Design Defense**

**CS-370**

**Professor Travis Smith**

**Southern New Hampshire University**

**Robert Marlatt**

**December 10, 2023**

This project was to design a way to solve a pirate maze game using reinforcement learning. The player (pirate) can go up, down, left or right to navigate the maze. The pirate will encounter blocks to their progress along the way. The goal is to get to the reward (treasure) at the end of the maze. This type of game can be solved easily by a human if they are looking at a map of the maze. If they are not, they would direct the pirate to move around the maze, avoiding blocks, until they reach the treasure. The human would also be able to use prior experience playing similar games, and their intuition to figure out the best possible path to the treasure.

The reinforcement learning algorithm in this notebook is playing the game as well, but the way that they play is based on random guesses as to the best path forward. The game loop makes a random choice of a direction to go. If it made the correct choice, it would move forward and make a new choice. If it made an incorrect choice, it would remember that and make a new choice to try again. This system continues until the treasure is found, or there are no more valid moves. Next time it plays the game, it will use the choices it made from prior games to inform its new choices, providing a weight to the correct choice. Eventually, over many iterations of the game, the reinforcement learning algorithm with learn the correct path through the maze and be able to accurately finish the game.

The biggest difference between the human and the algorithm approach is that the human is using their past experience to inform their choice of direction, whereas the algorithm approach is using knowledge gained by playing the game over and over, randomly picking a route until the correct one is reached. The biggest similarity is the fact that they are both randomly picking directions to traverse the playing field, not knowing what is surrounding them until they encounter it.

In reinforcement learning, exploration is a tactic used to select actions that the agent has not yet attempted or that it thinks will yield new knowledge or information that will enhance its long-term performance. The goal of exploration is to obtain new knowledge and information that will enable the agent to make better decisions and perform better in the future. Exploitation is the process of making decisions based on the agent's present knowledge in order to maximize its short-term performance, and it is defined as the strategy of selecting behaviors that the agent believes would yield the highest immediate reward given the knowledge it has learnt thus far (Yang, 2022).

With this Pirate Maze program, we used a mixture of exploration and exploitation in our modeling. We want the agent to randomly explore a lot in the early epochs but use the knowledge that it has gained over epochs more and more until the optimal result is achieved. We do this by using a learning decay rate. A simple alpha value is calculated each run by using the reciprocal of the epoch number multiplied by the decay rate:



Once the alpha falls below the epsilon threshold (ε), the agent stopped using random moves, and started using the knowledge it gained during prior runs (Haswani, 2020).

Using this method, a mix of exploration and exploitation based on the actual outcome of the runs, is the ideal method for pathfinding training for this game. This allows the training to adapt to the actual runs, instead of using an arbitrary number throughout the training. It allows the agent to randomly build up a knowledge base of the maze prior to using its knowledge base to correctly run the maze. Adding in positive rewards for correct moves and negative rewards for incorrect moves helps refine the reinforcement training, giving greater weight to correct responses for future runs.

The GameExperience and TreasureMaze classes are driven by a Deep Q Network (DQN) that was built in Python. The pirate agent was rewarded for reaching the treasure and penalized for being blocked, visiting the same square twice, or attempting an invalid move. The pirate was trained using reinforcement learning on the best course of action for each state, with two learning strategies used: exploration and exploitation. The action-state pairs were mapped using DQN, and a reward or punishment was assigned to each action. The rewards help the agent determine the optimal course of action when it moves from exploration to exploitation. This decision was made using a decaying learning function, whereby exploration is employed above a threshold (ε) and exploitation is used below.

**References**

Haswani, V. (2020, September 3). Learning rate decay and methods in deep learning. Medium. <https://medium.com/analytics-vidhya/learning-rate-decay-and-methods-in-deep-learning-2cee564f910b>

Yang, A. (2022, July 24). What is exploration vs. exploitation in reinforcement learning? Medium. <https://angelina-yang.medium.com/what-is-exploration-vs-exploitation-in-reinforcement-learning-a3b96dcc9503>