Applying Q-Learning to Classic Games: Prisoner's Dilemma, Hawk-Dove, and More

This study investigates the application of Q-learning, a reinforcement learning algorithm, to various game scenarios including the Prisoner's Dilemma, Hawk-Dove, and other cooperation games. The Python code, built with libraries for numerical computations, implements the core Q-learning logic. The experiment explores how players learn optimal strategies under different reward structures.

These structures are defined by two key parameters: G, representing the temptation to defect, and L, the penalty for mutual defection. Players make choices using an epsilon-greedy approach, balancing exploration of new strategies with exploitation of currently successful ones. The code simulates the environment's response based on their choices and updates Q-values, allowing players to learn through trial and error.

The main loop iterates through numerous episodes. With each episode, players adapt their strategies based on the rewards received in previous interactions. Finally, the optimal policy for each player in each scenario is extracted, revealing the most likely action (cooperate or defect) for different game states.

The experiment is further extended by analyzing the impact of the number of cooperation actions available. By varying the range used to represent the probability of choosing cooperation (between 0 and 1), the researchers observe how the optimal strategy changes. This analysis sheds light on how cooperation emerges or breaks down under different conditions within these classic game frameworks.

By analyzing the learned strategies across various reward structures and cooperation levels, this study offers valuable insights into the dynamics of cooperation and defection in these classic game scenarios. It demonstrates the effectiveness of Q-learning in modeling player behavior and predicting optimal strategies in these complex environments.