Student ID – 201674477 **Name – Naman Ahuja**

COMP532-202223 Assignment 2

Reinforcement Learning for Blackjack: A Deep Q-Learning Approach

Answer 1 -

Introduction-

The objective of this project is to train an agent using deep reinforcement learning to play the game of Blackjack. The agent learns to make optimal decisions by estimating the action-value function through interactions with the environment. The DQN model is employed, which utilizes a neural network to approximate the action-value function. The agent's performance is evaluated based on its ability to maximize rewards and minimize training error.

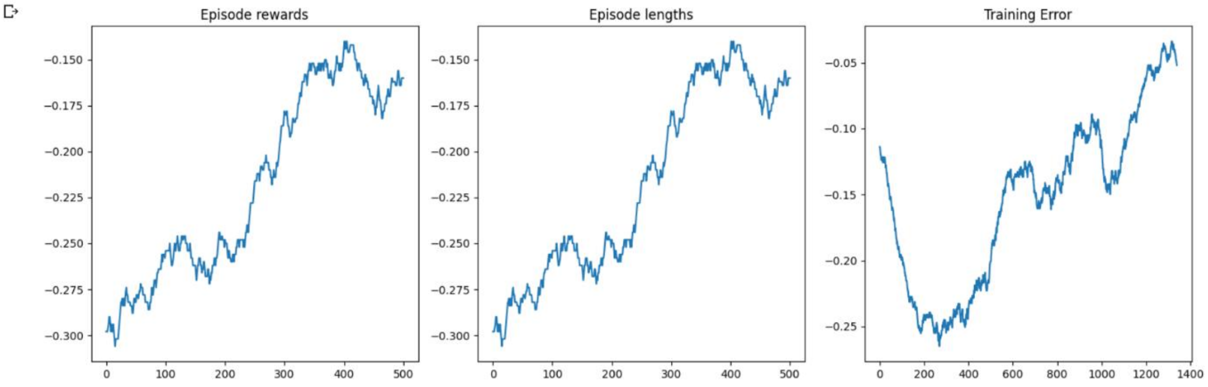
Methodology-

The environment used for training is the Blackjack-v1 environment from the Gymnasium library. The Q-table is initialized with zeros, and the agent follows an epsilon-greedy policy for exploration and exploitation. The Q-values are updated using the Q-learning algorithm, which incorporates the temporal difference error. The agent's epsilon value is decayed over time to gradually reduce exploration.

Results-

The agent was trained for 1000 episodes, and its performance was measured in terms of total reward and training error. The total reward represents the cumulative reward obtained by the agent in each episode, indicating its success in playing the game. The training error measures the temporal difference between predicted and target Q-values, reflecting the learning progress of the agent. The results show that the agent's total reward increases gradually over the training episodes, indicating its learning capability. Initially, the agent explores the environment and receives random rewards. As the training progresses, the agent learns to make better decisions and increases its total reward. However, it is important to note that there may be variations in the reward curve due to the stochastic nature of the game.

The training error plot demonstrates the convergence of the agent's Q-values. As the agent learns, the training error decreases, indicating that the Q-values are getting closer to the optimal values. The reduction in training error signifies the improvement in the agent's ability to estimate the function accurately.



Discussion-

The results of the experiment indicate the effectiveness of the deep reinforcement learning approach for training an agent to play Blackjack. The agent successfully learns to make optimal decisions by maximizing rewards and minimizing training error. The gradual increase in total reward demonstrates the agent's ability to learn a successful strategy for playing the game. The training error curve provides insights into the learning progress of the agent. As the agent interacts with the environment and receives feedback, it adjusts its Q-values to approximate the optimal action-value function. The decreasing training error signifies the convergence of the agent's Q-values towards the optimal values.

The success of the agent can be attributed to the DQN model's ability to approximate the action-value function efficiently. By employing a neural network, the model can capture complex patterns and generalize its knowledge to make informed decisions in different situations. The exploration-exploitation trade-off facilitated by the epsilon-greedy policy ensures that the agent explores the environment initially and gradually exploits its learned knowledge.

It is important to note that the performance of the agent may vary depending on the **hyper-parameters** and the specific blackjack environment. Adjusting the learning rate, epsilon values, and discount factor may yield different results. Furthermore, the stochastic nature of the game introduces randomness in rewards, which can affect the learning process. Possible improvements to the agent's performance include fine-tuning the hyper-parameters, implementing different exploration strategies, or utilizing advanced deep reinforcement learning techniques such as duelling DQN or prioritized experience replay.

Answer 2:

Exploration and exploitation are essential concepts in deep reinforcement learning that play a crucial role in the training process of the agent. In the above code, the agent balances exploration and exploitation through an epsilon-greedy policy.

Exploration:

Exploration refers to the process of the agent actively exploring the environment to gather information about different states and actions. It involves taking actions randomly or selecting sub-optimal actions to discover potentially better strategies. Exploration is crucial in the early stages of training when the agent has limited knowledge about the environment. In the provided code, exploration is implemented using an epsilon-greedy policy. The agent chooses a random action with a probability of epsilon. This randomness allows the agent to explore different states and actions, enabling it to learn about the environment's dynamics and discover potentially optimal strategies. As training progresses, the epsilon value is gradually decayed to reduce exploration and shift towards exploitation.

Exploitation:

Exploitation refers to the process of utilizing the agent's learned knowledge to select actions that are expected to maximize the cumulative reward. It involves selecting actions based on the agent's current estimates of the action-value function, aiming to exploit the known optimal strategies.

By combining exploration and exploitation, the agent can effectively learn and improve its performance over time. Initially, exploration helps the agent gather information about the environment and avoid getting stuck in suboptimal strategies. As training progresses and the agent's knowledge increases, exploitation allows the agent to make more informed decisions based on its learned Q-values, maximizing the cumulative reward. The balance between exploration and exploitation is crucial for successful training. Too much exploration may lead to slow convergence and delayed exploitation of optimal strategies, while too much exploitation may cause the agent to get stuck in local optima and miss out on potentially better solutions. The epsilon-greedy policy used in the code provides a controlled and gradual transition from exploration to exploitation as the agent's knowledge improves.

Conclusion-

In conclusion, exploration and exploitation are fundamental components of deep reinforcement learning. The provided code implements these concepts through an epsilon-greedy policy, allowing the agent to explore the environment and gradually shift towards exploiting its learned knowledge to maximize rewards. This balanced approach enables the agent to learn an effective strategy for playing Blackjack.