**Assignment 6: Reinforcement Learning**

**Analysis Report:**

DQN Agent for LunarLander-v2

In this assignment, we built a **DQN Agent** to successfully land the Lunar Lander in OpenAI Gym’s LunarLander-v2 environment. The DQN policy network utilized the ReLU (Rectified Linear Unit) activation function in the two hidden layers due to its computational efficiency and ability to mitigate the vanishing gradient problem, making it suitable for deep networks. We employed the MSE (Mean Square Error) loss function during training because it measures the difference between the predicted Q-values and the target Q-values, helping the network learn to approximate the Q-function. Additionally, this loss function penalizes larger errors more heavily, encouraging the network to focus on reducing significant discrepancies and leading to more accurate Q-value approximations.

**After training** the agent through a process of interacting with the environment, storing experiences, and updating the network, it successfully learned to land the Lunar Lander. In episode 970, the agent achieved a total reward of 282.73, demonstrating a consistent performance with an average reward exceeding 200 in the last 100 episodes. This benchmark signifies a successful landing strategy. The environment was considered solved in 877 episodes, as the agent achieved an average score of 200.62 over the last 100 episodes at that point. This indicates that by episode 877, the agent had developed a robust policy for landing the Lunar Lander consistently. The continued training up to episode 970 further solidified this performance, as evidenced by the high rewards achieved in subsequent episodes. According to the following training results, the agent's learning curve exhibits a constant progression:

* **Episode 100**: Average Score: 120.73
* **Episode 200**: Average Score: 127.54
* **Episode 300**: Average Score: 123.22
* **Episode 400**: Average Score: 117.45
* **Episode 500**: Average Score: 125.79
* **Episode 600**: Average Score: 140.39
* **Episode 700**: Average Score: 141.35
* **Episode 800**: Average Score: 156.37
* **Episode 900**: Average Score: 188.03
* Episode 970, Total Reward: 282.73
* Environment solved in 877 episodes! Average Score: 200.62

This progression indicates a constant improvement in the agent's performance. The cumulative rewards per 50 episodes remain relatively stable, indicating a consistent application of the optimal policy. Before the final training run, the agent struggled with landings, resulting in numerous negative rewards and a prolonged learning phase. However, by manually adjusting the hyperparameters, we were able to achieve the desired outcome. The agent's ability to consistently achieve high rewards demonstrates its successful learning and application of an effective landing strategy.

**The challenges** faced in this experiment were mostly related to finding the best hyperparameters to allow the agent to land successfully and some technical issues related to the installation of the ‘gym[box2d]’ dependency. We made several adjustments, and training runs until we achieved the best results. In terms of training duration, it was considered relatively long with a regular GPU in Google Colab.

As a **recommendation**, we could perform hyperparameter tuning to experiment with different learning rates, batch sizes, and epsilon decay rates to optimize performance and allow the agent to learn quickly without requiring many cycles that consume computational resources and do not yield more rewards.

A graph with blue lines and orange lines

Description automatically generated

Figure Performance Metrics