Deep Learning

PALASKOS ACHILLEAS (113)

3. Deep Reinforcement Learning

Atari (PongNoFrameskip-v4)

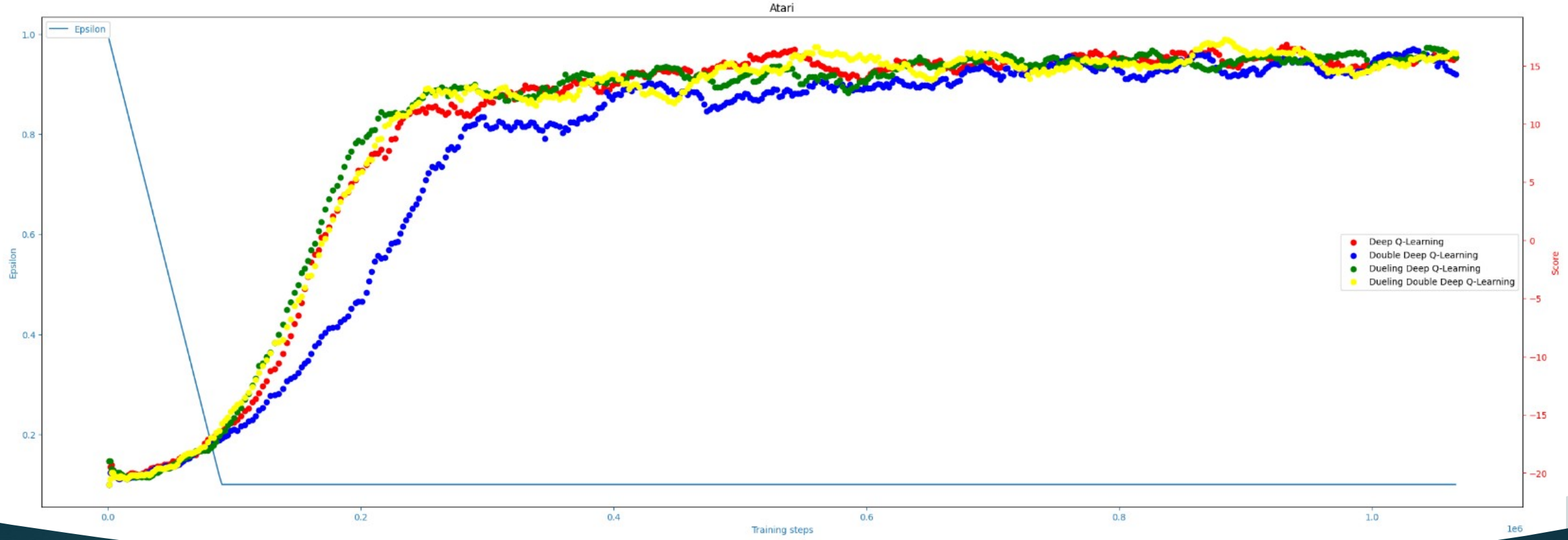
- **Environment**
  - Input to the NN 2 consecutive frames of the game so as to capture the movement of the ball
  - Images resized to lower dimensions scaled from 0 to 1.
- **Replay Buffer**
  - used to store the current states, actions, rewards, next states and whether or not a terminal state (done) was reached.
  - Total memory = 20,000 (when set to 50,000 I ran out of memory).
  - Sampled with a batch size of 32.
- **Model Architecture**
  - 3 convolutional layers
    - 32, 64 and 64 filters respectively
    - ReLU activations
    - Kernel sizes: 8, 4 and 4 respectively
    - Strides: 2, 3 and 1 respectively
  - 2 fully-connected layers
    - ReLU and Linear activations respectively

→

- **2 networks with the previous architecture were used**
  - One that was trained
  - The other was updated every 1000 iterations by using the current weights of the first.

Atari (PongNoFrameskip-v4)

- **Methods 4 methods were implemented *from scratch***
  - Deep Q-Learning
  - Double Deep Q-Learning
  - Dueling Deep Q-Learning
  - Dueling Double Q-Learning
- **Hyperparameters**
  - Number of games = 500
  - Learning rate =
  - (epsilon)
    - was initialized to 1
    - decreased by in each iteration until .
    - After that, it was kept constant



Atari (PongNoFrameskip-v4)

Thank you!