ECS7002P - Artificial Intelligence in Games
Individual Assignment: Breakout

Animesh Devendra Chourey

210765551

Google colab does not allow us to train for 2000000 frames. The maximum number of frames that I could train on were 800000 frames and the reward value I received was 3.04.

Question 1

The selection and execution process for each action is done by the agent according to an e-greedy policy based on Q. Since using histories of arbitrary length as inputs to a neural network can add much more overhead to the algorithm, fixed length of histories are used by Q-function created by the w function described in the code. The standard online Q-learning is modified by the algorithm in two ways to make it suitable for training large neural networks without diverging.

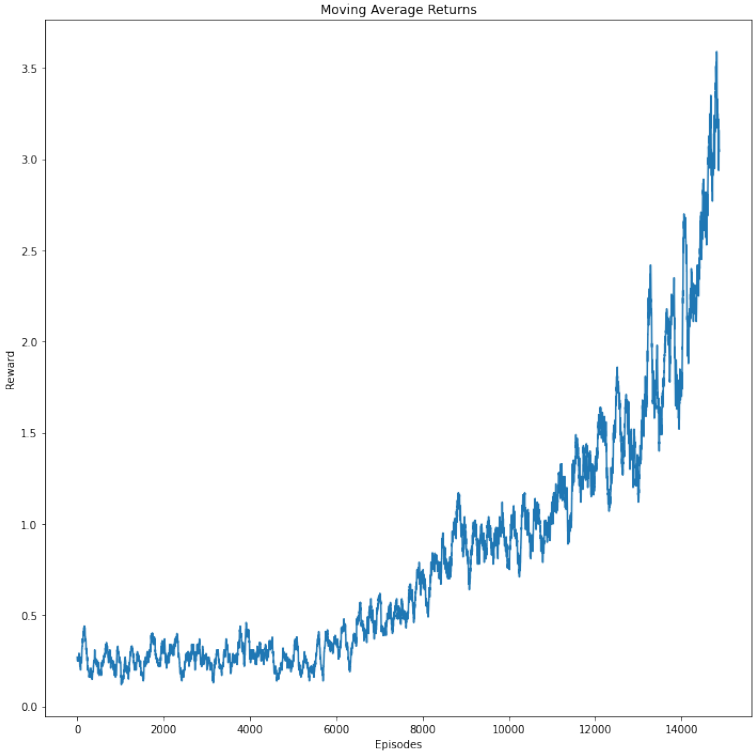
Question 2

The network Q is cloned into a target network Q' to generate targets for every update in Q-learning. If the target Q-network is not added, our second pass then occurs using the same weights in the network as first pass. Given this, the outputted Q-values will update, but our target Q-values will also, as they are calculated using the same weights. Our Q-values will move closer to target Q-values with each iteration, but the target Q-values will also move in the same direction. Adding target Q-network makes this divergence much more unlikely.

Question 3

The variable done_sample shows whether the episode has terminated or not. In the baseline implementation, when done_sample is true the Q values is set to -1, but this is not ideal for playing atari breakout game as -1 say game has ended which is not true. So, we have to change the equation such that even after encountering episode has end it continuously check for other possible action. So the equation retrain the old value when when episode has ended i.e done_sample is 1.

Question 4



Question 5

- **MaxAndSkipEnv :** This wrapper class provides the opearation to skip the frames. The skipped frame is returned to the environment state, same action is performed on the skipped frame, then the rewards are stacked and only the maximum value of the last two frames are taken.
- **EpisodicLifeEnv :** This wrapper class is used to speed up the training of the agent while also to avoid death as much as possible. To do this *done* is set to *True*.
- **WarpFrame :** This wrapper is used to process the picture data of the frame. The RGB image is converted into grayscale image. The image is rreshaped into size of 84x84 pixels.
- **ScaledFloatFrame :** The main aim of this wrapper class is to normalize the image from 0-255 to 0-1.
- **ClipRewardEnv :** The score measurement is different for every different game. Therefore to merge measurements and learning, all the rewards are

defined as 1(if reward>0), 0(if reward=0), -1(reward<0)

- **FrameStack :** This wrapper class combines k-number of frames of grayscale images into one frame so that CNN has some sequence information.