1. (written) For the Q learner we must represent the game as a set of states, actions, and rewards. OpenAI offers two versions of game environments: one which offers the state as the game display (image) and one that offers the state as the hardware ram (array). Which do you think would be easier for the agent to learn from and why?

The display image would be easier for the agent to learn from because every state of the game can be represented by a single frame of the display, the actions can be determined based off of the position of the ball within the frame, and the rewards can be determined by the results of the game. The hardware ram would be more difficult to represent the game as a set of states, actions and rewards because it would not be as intuitive to develop these criteria given an array that contains all the games information.

2. (written) Use the starter code to answer this question. Describe the purpose of the neural network in Q-Learning. Neural networks learn a complicated function mapping their inputs to their outputs. What will the inputs and outputs for this neural network be? What do these variables represent? The neural network in the starter code is class QLearner(nn.Module).

The purpose of the neural network in Q-Learning is to be able to calculate Q values for a given state-action combination without having to maintain a data structure to hold all of the potential values of the q table (could be extremely large). The input into the neural network is the state-action combination and the output are the Q values. Thus, the neural network maps state-action combinations to Q values. The variables represent the neural network itself. The forward function returns the possible actions given a state.

3. What is the purpose of lines 48 and 57 of dqn.py (listed below)? Doesn't the q learner tell us what action to perform?

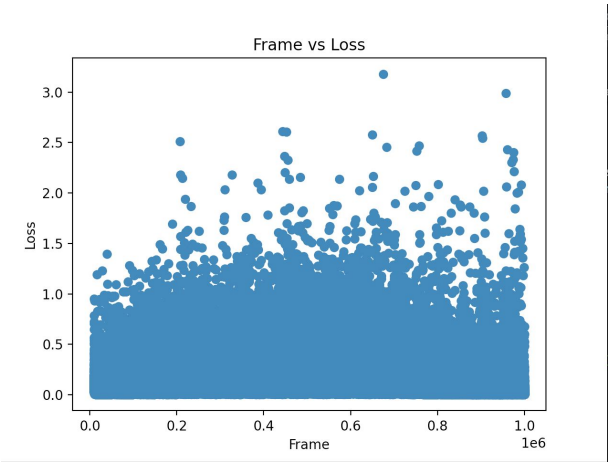
The random function helps with the concept of exploration vs exploitation. This ensures that the q learner is able to explore all potential actions to find the best policy without performing potentially naive actions that it has "learned" to be best.

4. Explain the objective function of Deep Q Learning Network for one-state lookahead below; what does each variable mean? Why does this loss function help our model learn? This is described in more detail in the Mitchell reinforcement learning text starting at page 383.

The theta represents all of the model parameters (state, action). The yi is the next state's target model's expected Q values and the Q(s,a) is the Q values from the current state, action combinations. The loss function calculates the mean squared error of the difference between the next state's target model Q values and the Q values obtained from the current model's Q values given the current state and action to perform. This loss function helps our model learn by the

idea of temporal difference learning. It learns by decreasing the difference between the estimated values of the Q_hat values of the state and its immediate successor.

5. Plot how the loss and reward change during the training process. Include these figures in your report.



As the training progressed, the rewards increased from -20 to 20.
The loss from each frame generally went down over the course of the 1 million frames, but slightly increased (not sure why).