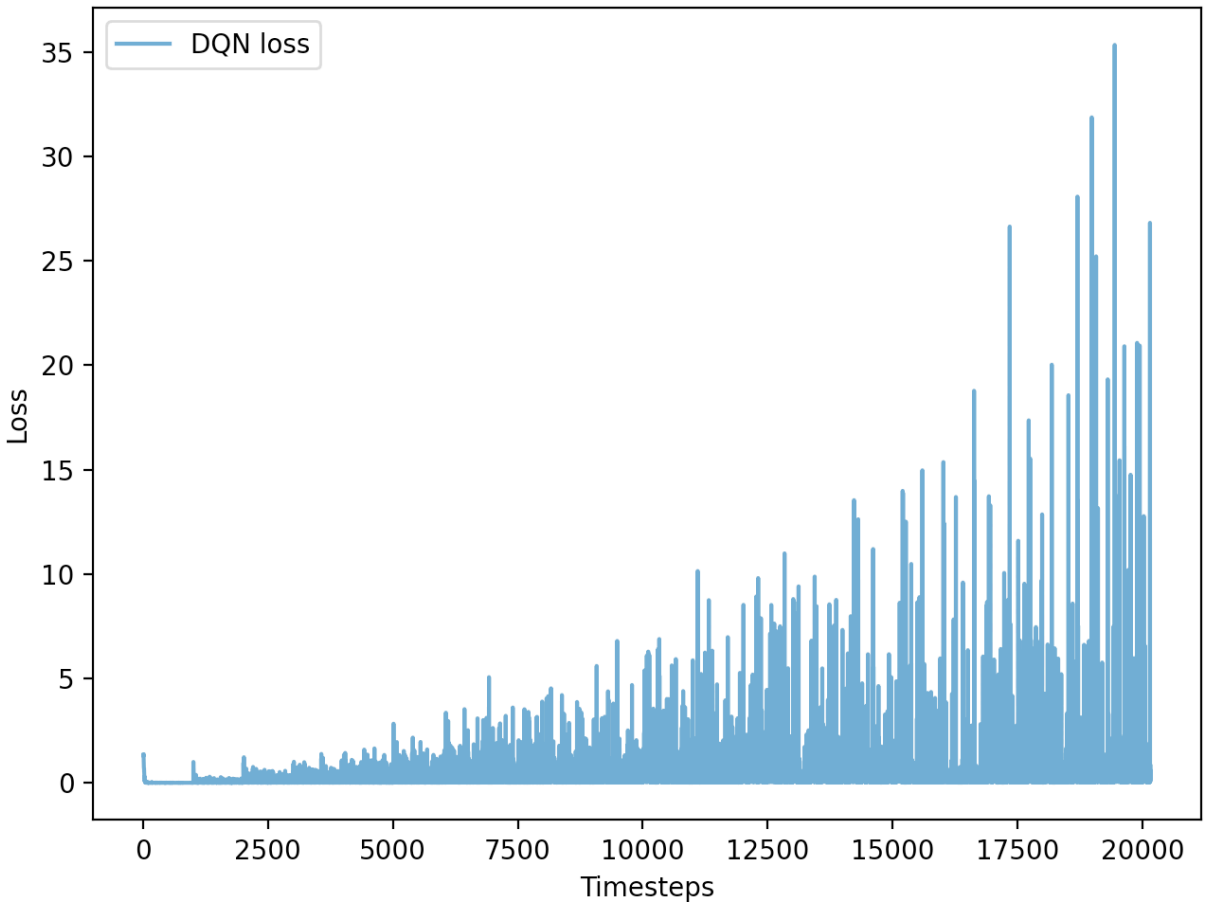

In the loss plot shown we see a linearly increasing loss as time progresses. Rewards increase with time as the agent is learning and performance improves, so the absolute loss also gets larger. It is further a squared loss. Also, as we use mini batch gradient descent, we effectively use a different training set with each iteration and we expect loss to be noisy.

The spikes could be caused by a minibatch that contains some outliers. Also, as we are looking at the square errors here, these will make a more significant contribution in the loss and appear as increasingly large spikes. If we also have our 'done==1' when the game is over, and the agent isn't aware of this then it predicts a high future reward. We are also updating the target network with a frequency of 500, and with each update we see an increase in loss.