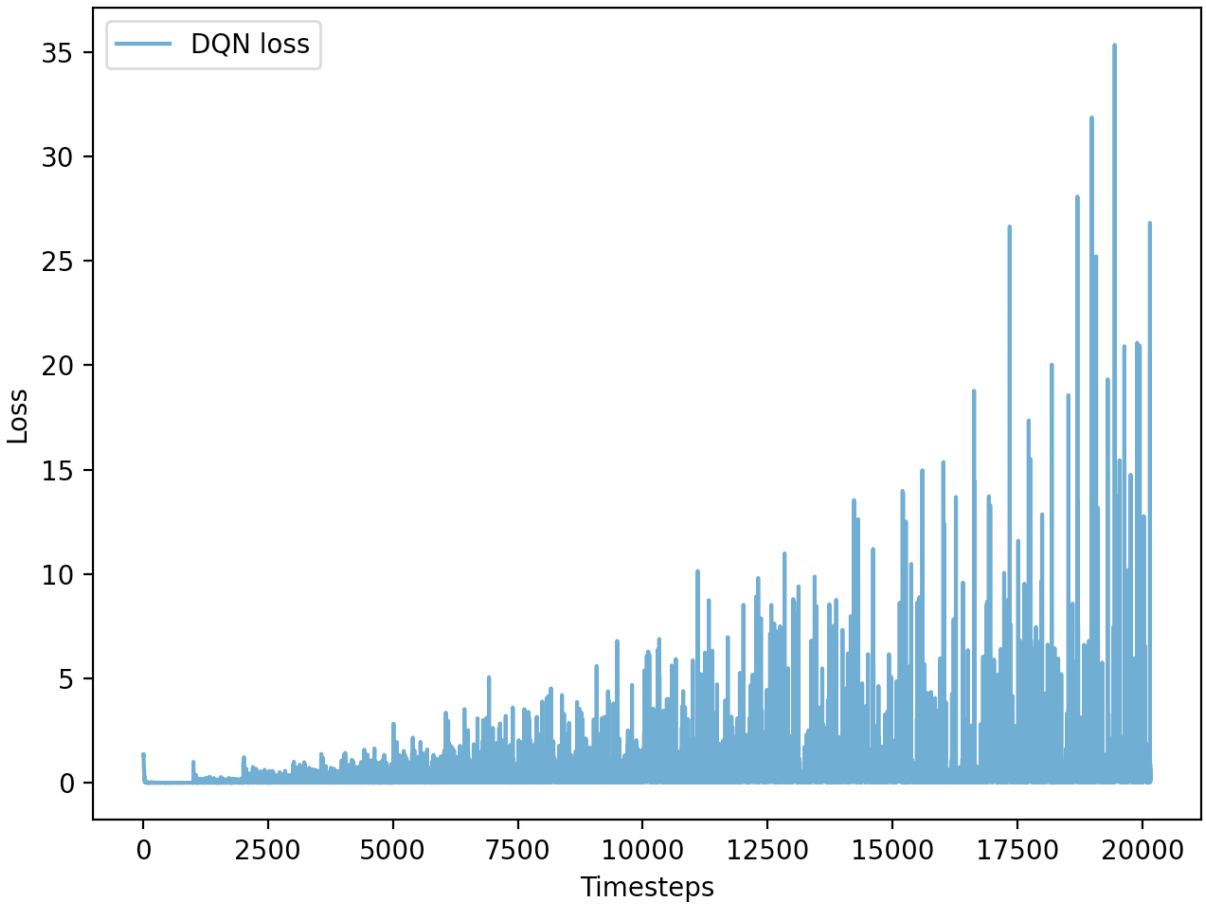

We see a broadly linear increase in loss with time. Rewards increase with time as the agent is learning and performance improves, so absolute loss also gets larger. We compare q values from the batch experience with the reward (given an action), plus optimal q value from the network at the next step, so we are using Adam optimiser on predicted (as opposed to real) values, causing additional loss. Also, in mini batch GD, we essentially use different training data at each iteration that will increase noise. The spikes can be caused by a minibatch containing outliers. As we are looking at the square errors, these will make more significant contributions appearing as increasingly large spikes. As we also update target network with a set frequency, we expect a periodic impact on the loss. If we also indicate the game is over, and an agent isn't aware, artificially high future rewards could be predicted, and hence higher losses.