RL assignment3 report

520030910342 Jiyu Liu

1. Implementation of the DQN and Double DQN algorithms

Note: This section only contains the discussion of coding problems. Plaese refer to the code folder for the programming part.

(a) The vanilla DQN surpasses DQN without target network at almost all stages (please refer to figure [1]). The reason is that without target network, since the Q-values and target values are determined by the same set of parameters θ, the target value would change constantly during training process, which results in the instability of network training. Using target network for target value in TD loss and updating its parameters every C steps can help address the problem above well, thus the vanilla DQN will have a better performance than DQN without target network.

(b) The vanilla DQN surpasses DQN without replay buffer at almost all stages (please refer to figure [1]). The reason is that if we do not use replay buffer, every sample consist of state, action, next state, reward and terminal flag will be used to update the network only once, which is very inefficient and is not independent and identically distributed (since the next sample is related to the current sample). However, with experience reply method, not only can we remove the correlation of samples, but also we can reuse one sample for several times and improve the efficiency of parameter updating.

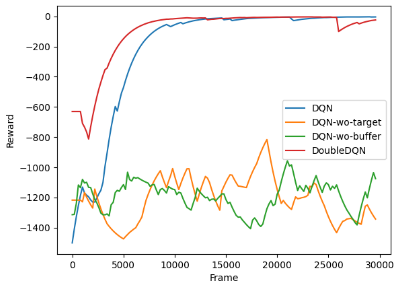

Figure 1: *Reward Comparison*

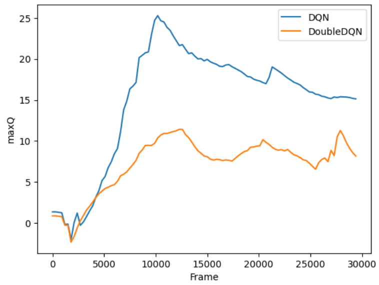

Figure 2: *maxQ comparison*

(c) The maximum Q-value of Double DQN is less than that of vanilla DQN (please refer to figure [2]). That's because every time we calculate the target value of TD loss $r + \gamma \max_{a'} Q_\theta(s', a')$ using deep networks, there always exist some positive or negative prediction errors. If we use one network to compute the action that makes Q-value largest and the corresponding Q-value, the positive error will accumulate and get larger and larger, which will definitely affect the performance. To address this problem, Double DQN method use the evaluation network to predict the action and use the target network to predict the corresponding Q-value. So the target value of TD loss in Double DQN method can be represented as $r + \gamma Q_{\theta'}(s', \operatorname{argmax}_{a'} Q_\theta(s', a'))$. By separating the prediction of action and Q-value into two networks, the accumulation of positive errors will be suppressed, just as the output shows.