Abstract

Q-learning and DQN methods are one of the most widely used methods of training, thanks to their efficiency and simplicity. Training an agent using such critic is very straightforward and involves minimal computation complexity. A simple gradient step on a proper loss function will yield the required policy gradient for the actor. These methods rely on accurately simulating and extracting the Q-values for each state-action pair and provide the actor with the required information. As a result, they are very sensitive to randomness in the environment, and noise in the observation space, as such disturbances will complicate the training of the critic and later, the actor itself. Our work is investigating a new technique first proposed by Bertsekas et al in 1997, which relies on a differential training rather than a classic one to retain and provide Q-value information about state-action pairs.

Unlike classic DQN where the critic implements the general Q(s,a) (s:state, a:action) function and each iteration involves the classic 1-step Bellman error update, in the differential version, the inputs to the critic are two state-action pairs, and the critic implements the difference between the Q-value of those two pairs, namely $Q(s_1,a_1) – Q(s_2,a_2)$. This critic could in fact be interpreted as the conventional critic for an environment whose state space consist of (s_1,s_2) and whose action space is (a_1,a_2), where s_i's and a_i's are states and actions from the original environment, with rewards for each step equal to $r(s_1,a_1) – r(s_2,a_2)$. Unlike other alternative methods such as advantage training, the differential training can be trained using effective training methods such as TD(λ), which adds great flexibility in applying this method to various tasks.

Here we have implemented the differential DQN and compared it to a baseline classic DQN in various stochastic environments and different reward distributions. To demonstrate the robustness of this method to observation noise, we have also compared them under different levels of injected noise and observed how their performance varies as the noise level increases.

We simulated both DQN methods in both discrete and continuous LunarLander environment, and in both of those the differential critic was not only able to achieve a higher reward, but the agent actually learned the act of landing the spaceship on the platform, a task the classic DQN was not able to do. We also simulated both methods in the HalfCheetah environment, but unlike in the LunarLander case, the classic DQN had a slightly higher return performance than its same-sized differential counterpart. Our investigation and simulations concluded that this is due to the fact that a same-sized differential critic has a much broader input-space (=twice the number of inputs as a classic DQN), and therefore with a given size of critic network, the classic DQN is much more able and expressive since it has a higher parameter count per input item, and the HalfCheetah environment being very rich observation and action, unlike LunarLander, heavily restricts the expressiveness of the differential critic network since the input space to learn from is so high dimensional, and as a result its performance is limited compared to a classic DQN method.

We also simulated both methods under different observation noise levels in the Pendulum environment and demonstrated how the differential DQN method is able to tolerate more noisy observations and still achieve an acceptable performance, unlike the classic DQN method. Our final experiment was aimed at demonstrating the limitations of differential DQN, and how a constant-reward system, such as CartPole, is a special case where differential DQN will not be able to be trained on, due to its differential nature.

Our work here aimed at implementing and evaluating the differential DQN method described by Bertsekas and show how it compares to the baseline. Our simulation results and investigative evaluation revealed that the differential method, if properly trained, is preferrable to a classic DQN, and is generally more stable and robust.