Neural q-learning for 2048

Kristian Wichmann

February 17, 2018

1 The q function

In reinforcement learning, an agent following a policy π can make its decisions based on the q-function, also known as the action-value function:

$$q_\pi(s,a) = \mathbb{E}_\pi[G_t|S_t = s, A_t = a] \tag{1.1}$$

I.e. the expected value of the future reward function G (which we'll get back to in a minute) given that we take action a while being in state s at time t and then following policy π thereafter.

Usually, the function G will be discounted. Which means that while we do want to account for future rewards, rewards that are far in the future should matter less. So in each time step into the future we multiply by a discounting factor $\gamma \in [0,1]$, so that:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \gamma^3 R_{t+4} + \cdots \tag{1.2}$$

Now, assuming we know the system (or game in this case) is in the state s, we should evaluate $q_\pi(s,a)$ for all possible actions and let the agent take the action with the highest expected future reward.