SYSU & 朝旭 机器学习研修班 讲义 7

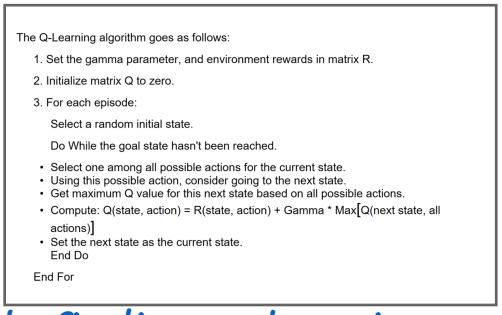
强化学习与深度强化学习

主讲人：付星宇

Step1. 增强学习与 DQN 基础知识

https://www.intelnervana.com/demystifying-deep-reinforcement-learning/

Step2. Q 迭代实战



The Q-Learning algorithm goes as follows:

1. Set the gamma parameter, and environment rewards in matrix R.
2. Initialize matrix Q to zero.
3. For each episode:

   Select a random initial state.

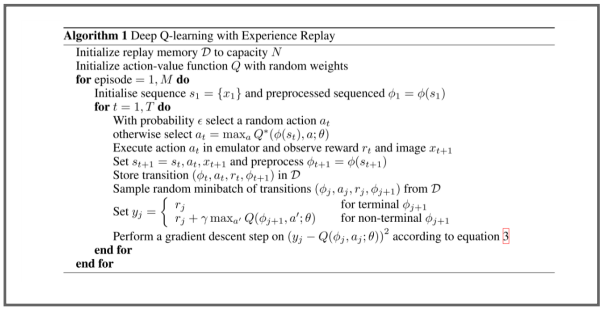
   Do While the goal state hasn't been reached.

- Select one among all possible actions for the current state.
- Using this possible action, consider going to the next state.
- Get maximum Q value for this next state based on all possible actions.
- Compute: Q(state, action) = R(state, action) + Gamma * Max[Q(next state, all actions)]
- Set the next state as the current state.
  End Do

End For

http://mnemstudio.org/path-finding-q-learning-tutorial.htm

Step3. DQN 实战



Algorithm 1 Deep Q-learning with Experience Replay

Initialize replay memory $\mathcal{D}$ to capacity N
Initialize action-value function Q with random weights
for episode $= 1, M$ **do**
  Initialise sequence $s_1 = \{x_1\}$ and preprocessed sequenced $\phi_1 = \phi(s_1)$
  for $t = 1, T$ **do**
    With probability ϵ select a random action a_t
    otherwise select $a_t = \max_a Q^*(\phi(s_t), a; \theta)$
    Execute action a_t in emulator and observe reward r_t and image x_{t+1}
    Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$
    Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in $\mathcal{D}$
    Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from $\mathcal{D}$
    Set $y_j = \begin{cases} r_j & \text{for terminal } \phi_{j+1} \\ r_j + \gamma \max_{a'} Q(\phi_{j+1}, a'; \theta) & \text{for non-terminal } \phi_{j+1} \end{cases}$
    Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ according to equation 3
  end for
end for

https://yanpanlau.github.io/2016/07/10/FlappyBird-Keras.html

非常抱歉，这周时间不是很多，故没有详细写一份讲义，读者按顺序好好读懂这三篇博客，就可以对 Q 学习有基本的认识了。