DQN 算法

初始化策略网络参数 θ

复制参数到目标网络 $\hat{Q} \leftarrow Q$

初始化经验回放 D

for 回合数 $= 1, M$ **do**

重置环境，获得初始状态 s_t

for 时步 $= 1, t$ **do**

根据 $\varepsilon - greedy$ 策略采样动作 a_t

环境根据 a_t 反馈奖励 s_t 和下一个状态 s_{t+1}

存储 transition 即 (s_t, a_t, r_t, s_{t+1}) 到经验回放 D 中

更新环境状态 $s_{t+1} \leftarrow s_t$

更新策略：

从 D 中采样一个 batch 的 transition

计算实际的 Q 值，即 $y_j = \begin{cases} r_j & \text{对于终止状态} s_{j+1} \\ r_j + \gamma \max_{a'} Q\left(s_{j+1}, a'; \theta\right) & \text{对于非终止状态} s_{j+1} \end{cases}$

对损失 $\left(y_j - Q\left(s_j, a_j; \theta\right)\right)^2$ 关于参数 θ 做随机梯度下降

每 C 步复制参数 $\hat{Q} \leftarrow Q$

end for

end for

SoftQ 算法

初始化参数 θ 和 ϕ

复制参数 $\bar{\theta} \leftarrow \theta, \bar{\phi} \leftarrow \phi$

初始化经验回放 D

for 回合数 $= 1, M$ **do**

  for 时步 $= 1, t$ **do**

    根据 $a_t \leftarrow f^{\phi}(\xi; \mathbf{s}_t)$ 采样动作，其中 $\xi \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$

    环境根据 a_t 反馈奖励 s_t 和下一个状态 s_{t+1}

    存储 transition 即 (s_t, a_t, r_t, s_{t+1}) 到经验回放 D 中

    更新环境状态 $s_{t+1} \leftarrow s_t$

    待完善

  end for

end for