Maximize cumulative future reward

$$\boxed{Q^*(s,a) = \max_{\pi} \mathbb{E}\left[r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \cdots \middle| s_t = s, a_t = a, \pi \right]}$$

CNN Model

Problem:

① correlations present in the sequence of observations

"small update" to Q ⟶ "significantly change" the policy

↓

"change data distribution"

Experience Replay (randomize over the data)

② Correlation between action-values Q

and target value $r + \gamma \max_{a'} Q(s', a')$

iterative update that adjust the action-values

Q toward target values

$Q(s, a; \theta_i)$ CNN model

$$\underbrace{e_t = (s_t, a_t, r_t, s_{t+1})}$$

$$D_t = \{e_1, e_2, \dots, e_t\}$$

$$(s, a, r, s') \sim U(D)$$

$$L_i(\theta_i) = \mathbb{E}_{(s,a,r,s') \sim U(D)}\left[\left(r + \gamma \max_{a'} Q(s', a'; \theta_i^-) - Q(s, a; \theta_i)\right)^2\right]$$

network parameters used to compute the target at iteration i

parameters of the Q-network at iteration i

θ_i update
C steps $\xrightarrow{\text{then}}$ θ_i^- update

history action pairs $\longrightarrow$ Q-Value of each action

$$R_t = \sum_{t'=t}^{T} \gamma^{t'-t} r_{t'}$$

$$Q^*(s,a) = \mathbb{E}_{s'}\left[r + \gamma \max_{a'} Q^*(s',a') \mid s,a \right]$$

$$\boxed{Q(s,a;\theta) \approx Q^*(s,a)}$$

"use a function approximator to estimate the action-value function"

$$Q(s,a;\theta) \approx Q^*(s,a)$$

$$y = r + \gamma \max_{a'} Q(s',a';\theta_{i}^{-})$$

$$L_i(\theta_i) = \mathbb{E}_{s,a,r}\left[\left(\mathbb{E}_{s'}[y \mid s,a] - Q(s,a;\theta_i) \right)^2 \right]$$

$$= \mathbb{E}_{s,a,r,s'}\left[(y - Q(s,a;\theta_i))^2 \right] +$$

$$\mathbb{E}_{s,a,r}\left[\mathbb{V}_{s'}[y] \right]$$

$$\nabla_{\theta_i} L(\theta_i) = \mathbb{E}_{s,a,r,s'}\left[\left(\boxed{r + \gamma \max_{a'} Q(s', a'; \theta_i^-)} - Q(s, a; \theta_i)\right) \nabla_{\theta_i} Q(s, a; \theta_i)\right]$$

$\theta_i^- = \theta_{i-1}$ from previous iteration

$$Q(s,a) \leftarrow Q(s,a) + \alpha\left[r + \gamma \max_{a'} Q(s', a') - Q(s,a)\right]$$

$$L(\theta) = \mathbb{E}\left[\left(\boxed{r + \gamma \max_{a'} Q(s', a'; \theta)} - Q(s, a; \theta)\right)^2\right]$$

Algorithm 1: deep Q-learning with experience replay.

Initialize replay memory D to capacity N

Initialize action-value function Q with random weights θ

Initialize target action-value function $\hat{Q}$ with weights $\theta^- = \theta$

For episode = 1, M **do**

  Initialize sequence $s_1 = \{x_1\}$ and preprocessed sequence $\phi_1 = \phi(s_1)$

  For $t = 1$,T **do**

    With probability ε select a random action a_t

    otherwise select $a_t = \text{argmax}_a Q(\phi(s_t), a; \theta)$

    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 D

    Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from D

    Set $y_j = \begin{cases} r_j & \text{if episode terminates at step j+1} \\ r_j + \gamma \max_{a'} \hat{Q}(\phi_{j+1}, a'; \theta^-) & \text{otherwise} \end{cases}$

    Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ with respect to the network parameters θ

    Every C steps reset $\hat{Q} = Q$

  End For

End For

$$\boxed{r_j + \gamma \max_{a'} \hat{Q}(\phi_{j+1}, a'; \theta^-)} - \boxed{Q(\phi_j, a_j; \theta)}$$

Target Network θ^- Q-network θ

$$L_i(\theta_i) = \mathbb{E}_{(s,a,r,s') \sim U(D)} \left[\left(\left(r + \gamma \max_{a'} Q(s', a'; \theta_i^-) \right) - \left(Q(s, a; \theta_i) \right) \right)^2 \right]$$

$$r + \gamma \max_{a'} Q^*(s', a')$$