

Maximize cumulative future reward

$$Q^*(s, a) = \max_{\pi} \mathbb{E}[R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \dots |$$

$$s_t = s, a_t = a, \pi]$$

CNN Model

Problem:

① correlations present in the sequence of observations

"small update" to  $Q$   $\longrightarrow$  "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

$$\mathcal{R}_t = (S_t, a_t, r_t, S_{t+1})$$

$$\mathcal{D}_t = \{ \mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_t \}$$

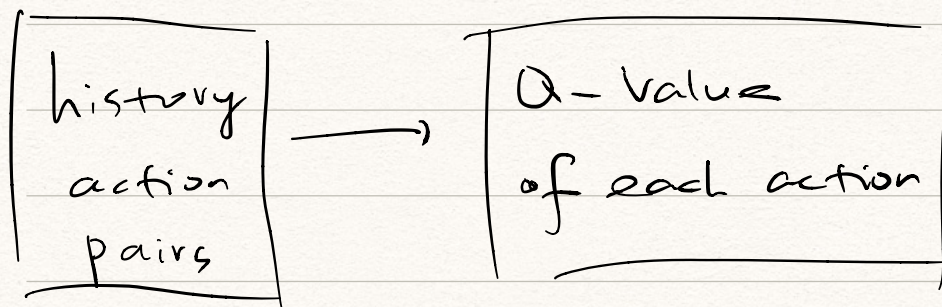
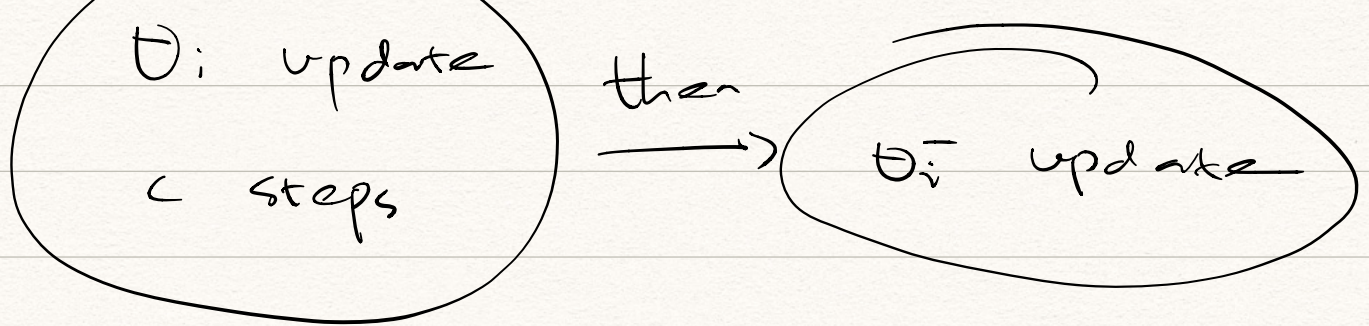
$$(s, a, r, s') \sim U(\mathcal{D})$$

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

network parameters  
used to compute the  
target at iteration  $i$

parameters of the Q-network  
at iteration  $i$







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

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

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

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

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

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

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

$$\mathbb{E}_{s, a, r} [\gamma \mathbb{E}_{s'} [y]]$$



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

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

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

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



### Algorithm 1: deep Q-learning with experience replay.

Initialize replay memory  $D$  to capacity  $N$

Initialize action-value function  $Q$  with random weights  $\theta$

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  $\varepsilon$  select a random action  $a_t$

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

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  $\theta$

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

**End For**


**End For**

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

Target Network  $\theta^-$       Q-network  $\theta$

$$L_i(\theta_i) = \mathbb{E}_{(s, a, r, s') \sim U(D)}$$

$$\left[ \left( r + \gamma \max_{a'} Q(s', a'; \theta_i^-) \right) - \left( Q(s, a; \theta_i) \right) \right]^2$$


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