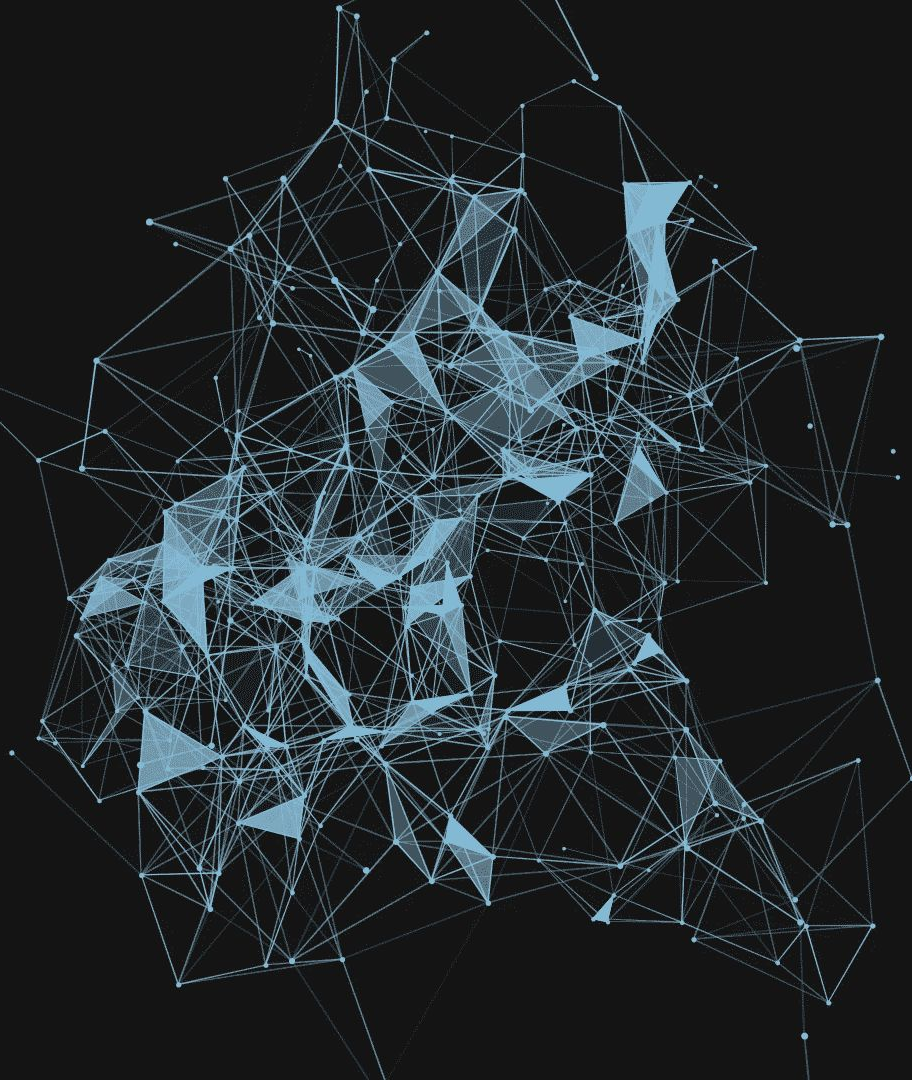


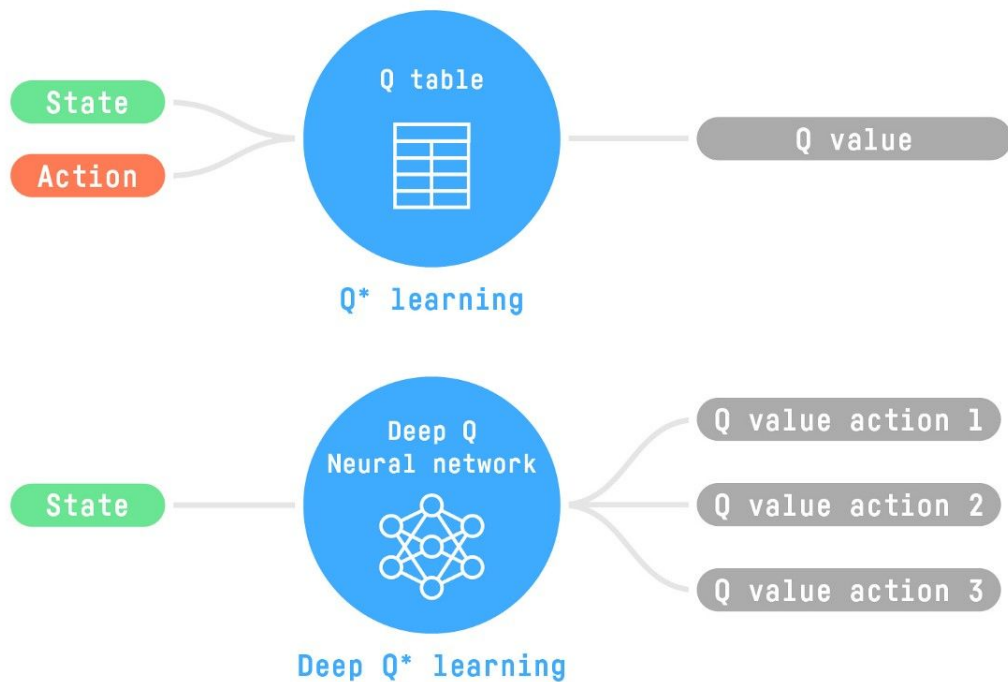
# Reinforcement

# Learning

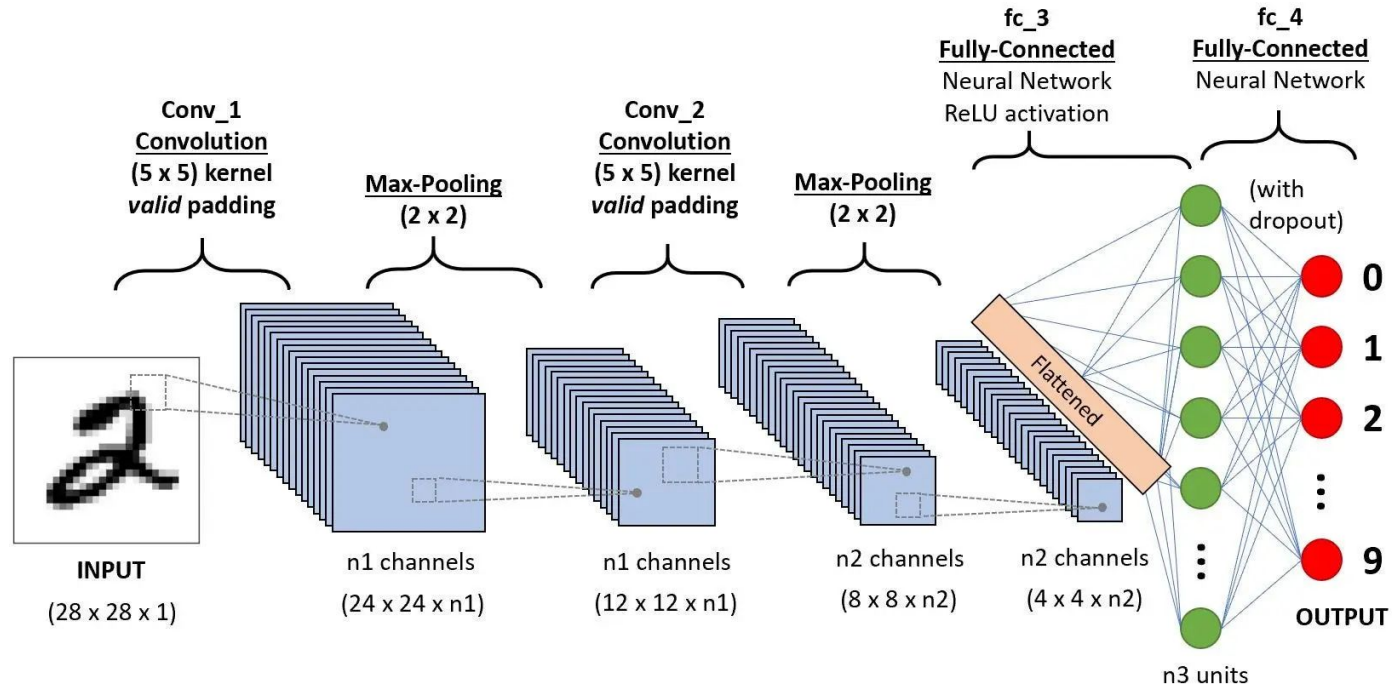
# Lesson - 5



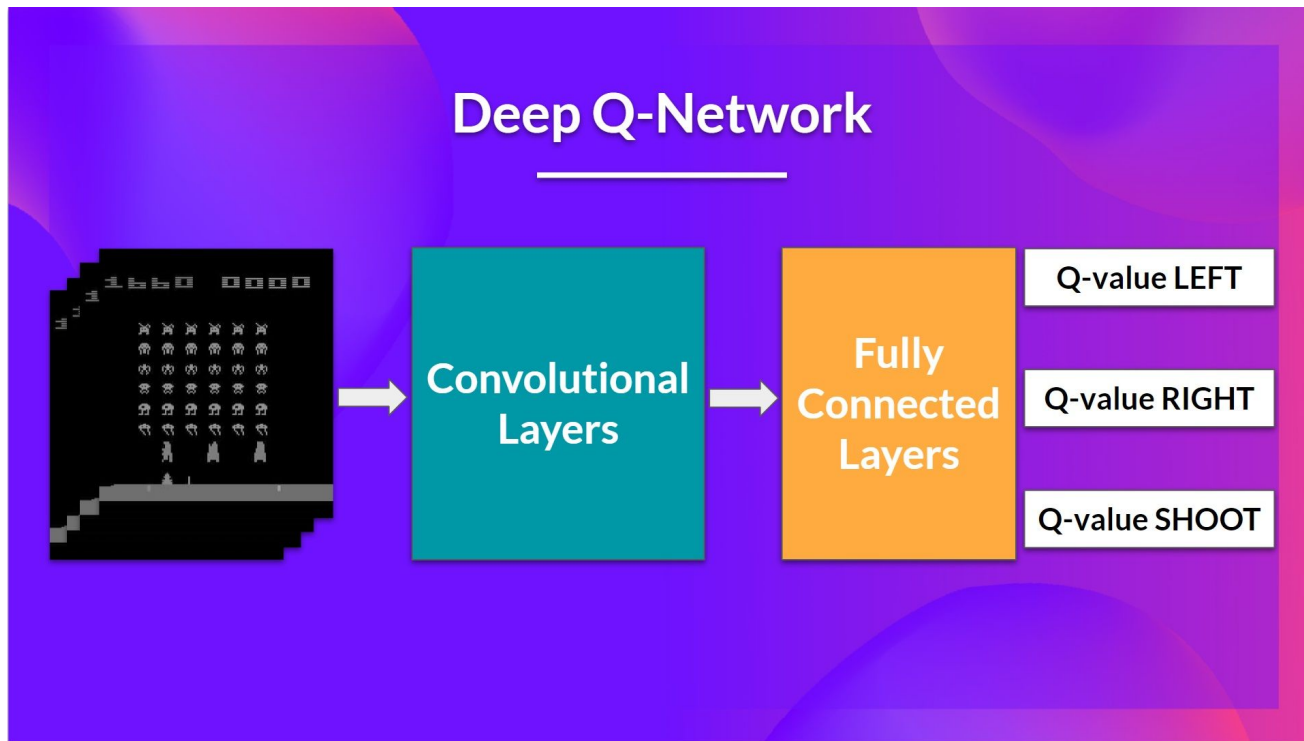
# From Q-Learning to Deep Q-Learning



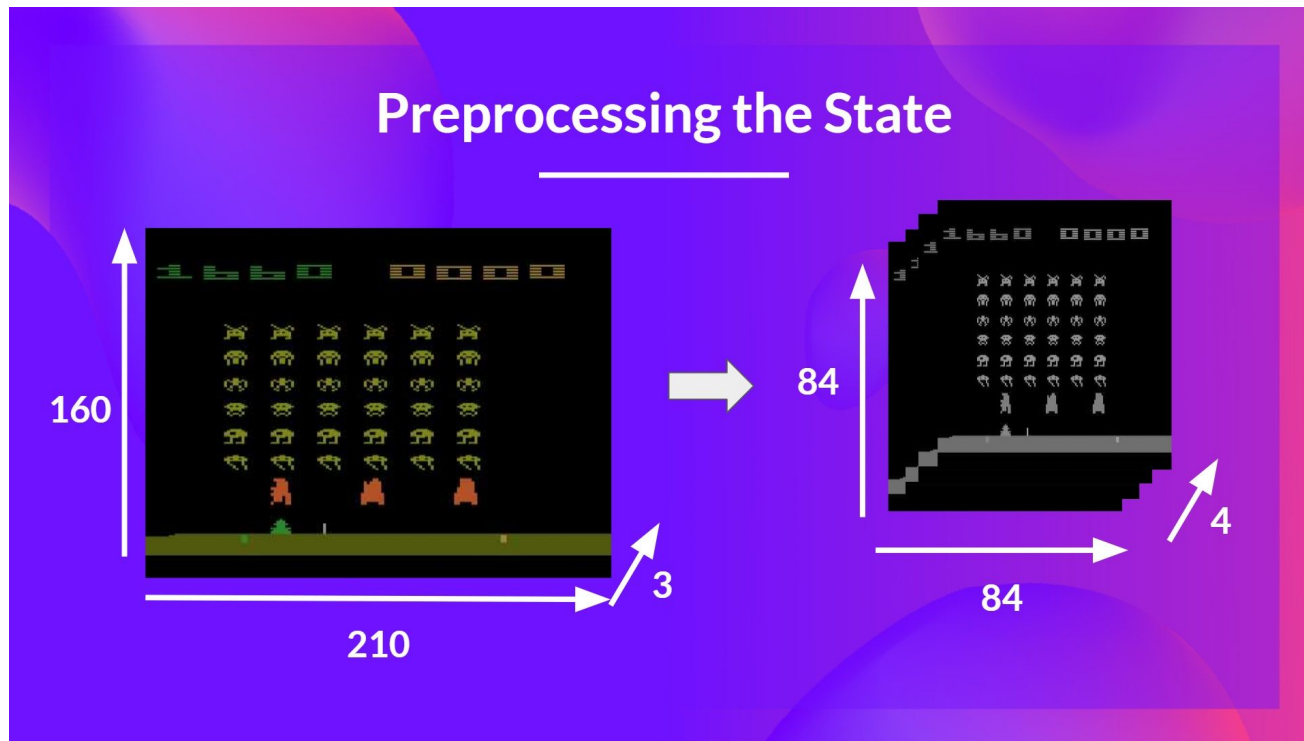
# From Q-Learning to Deep Q-Learning



# From Q-Learning to Deep Q-Learning



# From Q-Learning to Deep Q-Learning



$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha [R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)]$$

New  
Q-value  
estimation

Former  
Q-value  
estimation

Learning  
Rate

Immediate  
Reward

Discounted Estimate  
optimal Q-value  
of next state

Former  
Q-value  
estimation

TD Target

TD Error

## Q-Target

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

$$R_{t+1} + \gamma \max_a Q(S_{t+1}, a)$$

Immediate  
Reward

Discounted Estimate  
optimal Q-value  
of next state

TD Target

## Q-Loss

$$y_j - Q(\phi_j, a_j; \theta)$$

$$[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)]$$

Immediate  
Reward

Discounted Estimate  
optimal Q-value  
of next state

Former  
Q-value  
estimation

TD Target

TD Error

# From Q-Learning to Deep Q-Learning

---

**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**

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

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

        With probability  $\epsilon$  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**

---

Sample

Train