Action-value functions

Discounted Return

Definition: Discounted return (aka cumulative discounted future reward).

- $U_t = R_t + \gamma\, R_{t+1} + \gamma^2\, R_{t+2} + \gamma^3\, R_{t+3} + \cdots$

- The return depends on actions $A_t, A_{t+1}, A_{t+2}, \cdots$ and states $S_t, S_{t+1}, S_{t+2}, \cdots$
- Actions are random: $\mathbb{P}[A = a \mid S = s] = \pi(a|s)$. (Policy function.)
- States are random: $\mathbb{P}[S' = s'|S = s, A = a] = p(s'|s, a)$. (State transition.)

Action-Value Functions $Q(s, a)$

Definition: Discounted return (aka cumulative discounted future reward).

- $U_t = R_t + \gamma\, R_{t+1} + \gamma^2\, R_{t+2} + \gamma^3\, R_{t+3} + \cdots$

Definition: Action-value function for policy π.

- $Q_\pi(s_t, a_t) = \mathbb{E}\,[U_t|S_t = s_t, A_t = a_t]$.

  - Taken w.r.t. actions $A_{t+1}, A_{t+2}, A_{t+3}, \cdots$ and states $S_{t+1}, S_{t+2}, S_{t+3}, \cdots$
  - Integrate out everything except for the observations: $A_t = a_t$ and $S_t = s_t$.

Action-Value Functions $Q(s, a)$

Definition: Discounted return (aka cumulative discounted future reward).

- $U_t = R_t + \gamma\, R_{t+1} + \gamma^2\, R_{t+2} + \gamma^3\, R_{t+3} + \cdots$

Definition: Action-value function for policy π.

- $Q_\pi(s_t, a_t) = \mathbb{E}\,[U_t | S_t = s_t, A_t = a_t]$.

Definition: Optimal action-value function.

- $Q^\star(s_t, a_t) = \max_\pi Q_\pi(s_t, a_t)$.
- Whatever policy function π is used, the result of taking a_t at state s_t cannot be better than $Q^\star(s_t, a_t)$.

Approximate the Q Function

Goal: Win the game ($\approx$ maximize the total reward.)

Question: If we know $Q^\star(s, a)$, what is the best action?

- Obviously, the best action is $a^\star = \operatorname{argmax}_a Q^\star(s, a)$.

$Q^\star$ is an indication for how good it is for an agent to pick action a while being in state s.

Approximate the Q Function

Goal: Win the game (≈ maximize the total reward.)

Question: If we know $Q^\star(s, a)$, what is the best action?

- Obviously, the best action is $a^\star = \underset{a}{\operatorname{argmax}}\, Q^\star(s, a)$.

Challenge: We do not know $Q^\star(s, a)$.

Deep Q Network

Approximate the Q Function

Goal: Win the game (≈ maximize the total reward.)

Question: If we know $Q^\star(s, a)$, what is the best action?

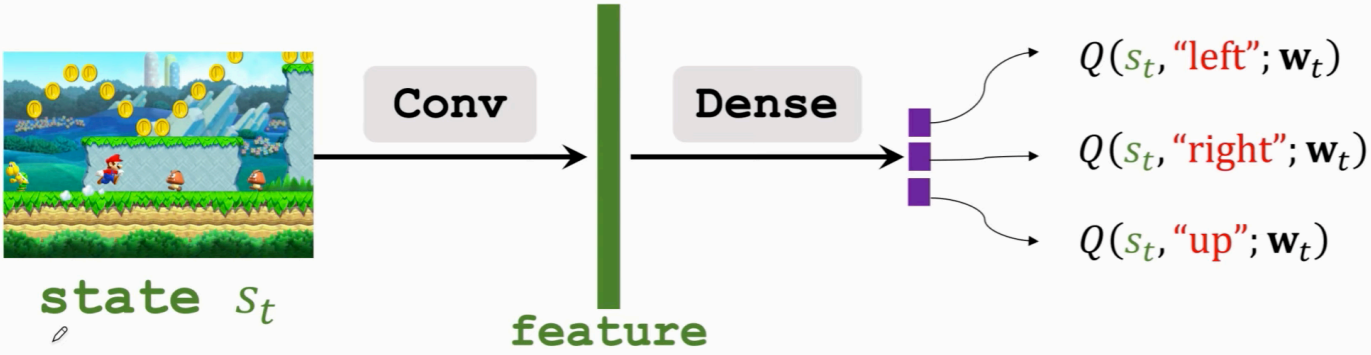
- Obviously, the best action is $a^\star = \underset{a}{\operatorname{argmax}}\, Q^\star(s, a)$.

Challenge: We do not know $Q^\star(s, a)$.

- Solution: Deep Q Network (DQN)
- Use neural network $Q(s, a; \mathbf{w})$ to approximate $Q^\star(s, a)$.

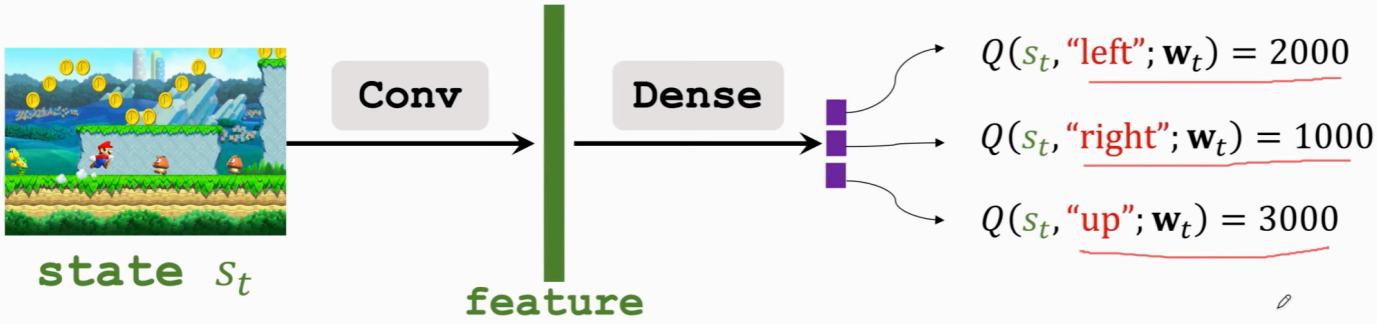
Deep Q Network (DQN)

- Input shape: size of the screenshot.
- Output shape: dimension of action space.



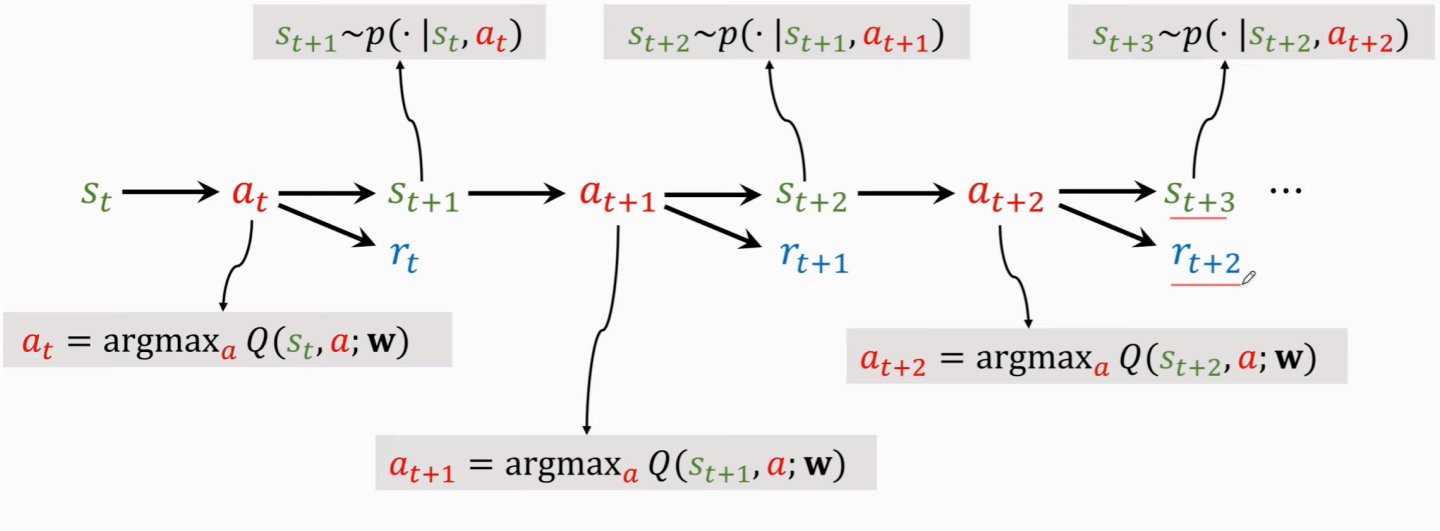
Deep Q Network (DQN)

- Input shape: size of the screenshot.
- Output shape: dimension of action space.



Question: Based on the predictions, what should be the action?

Apply DQN to Play Game



Temporal Difference (TD) Learning