Plan

Recap

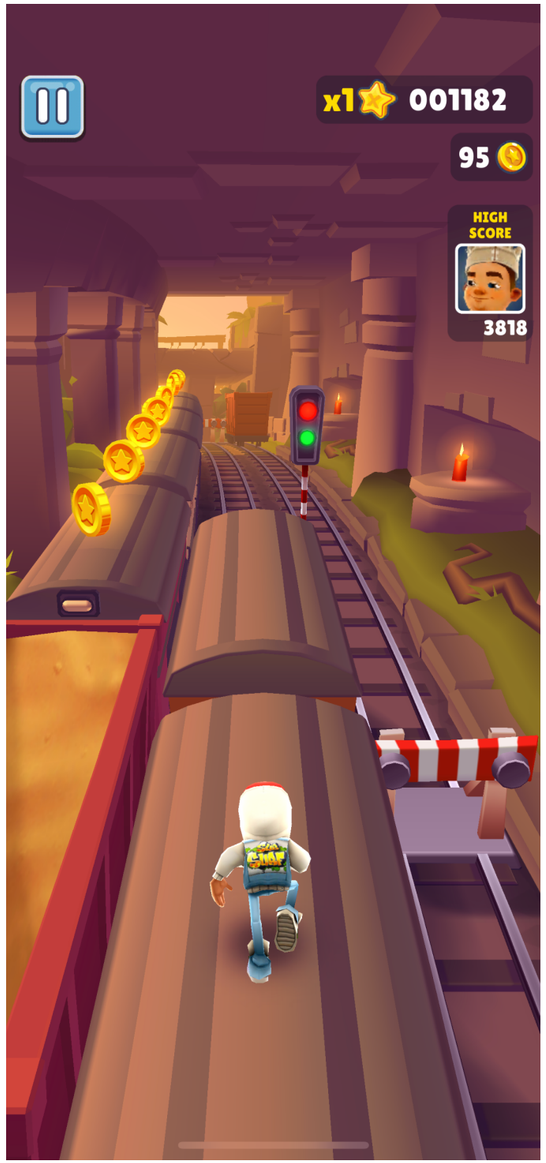
Logistics

Q Learning

Deep Q Learning

Comparison to Policy Gradients

Recap



State

Action

Reward

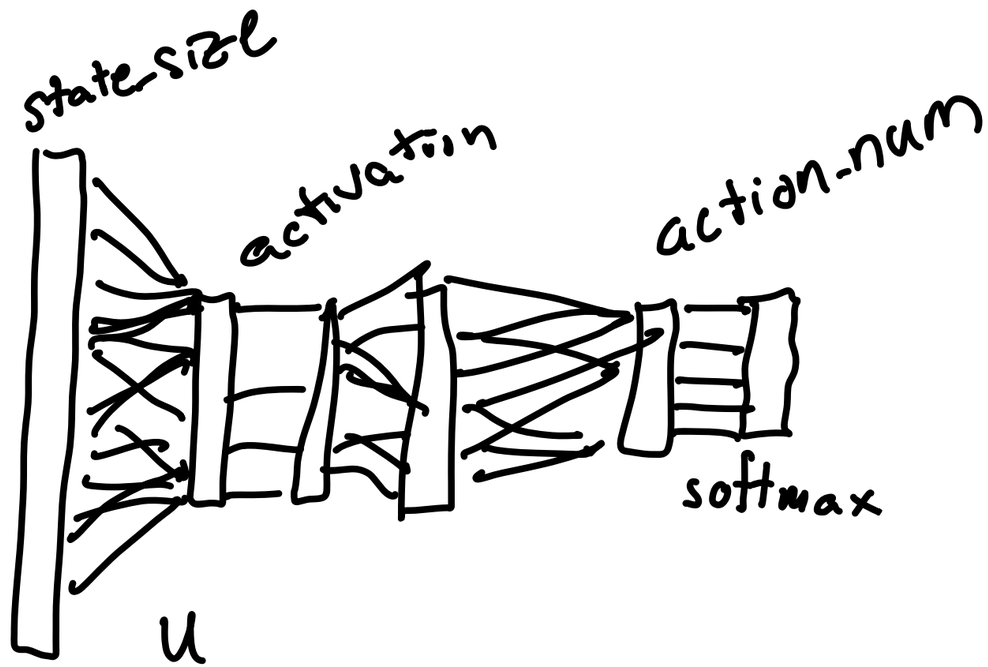
$$\min_{\theta} - \mathbb{E}_{\tau \sim \pi_\theta}[R(\tau)]$$

$$\pi: \mathbb{R}^{\text{state\_size}} \rightarrow \mathbb{R}^{\text{action\_num}}$$

$$\theta \leftarrow \theta + \eta \underbrace{\frac{\partial\, \mathbb{E}[R(\tau)]}{\partial \theta}}_{\mathbb{E}\left[R(\tau) \cdot \frac{\partial \log \pi(\tau)}{\partial \theta}\right]}$$

REINFORCE

1. Sample trajectory τ according π
2. Calculate $R(\tau)$
3. $\theta \leftarrow \theta + \eta R(\tau) \cdot \frac{\partial \log \pi(\tau)}{\partial \theta}$



Logistics

Grades

↳ By point

↳ Participate and turn everything → B or higher

↳ Improvement ☺

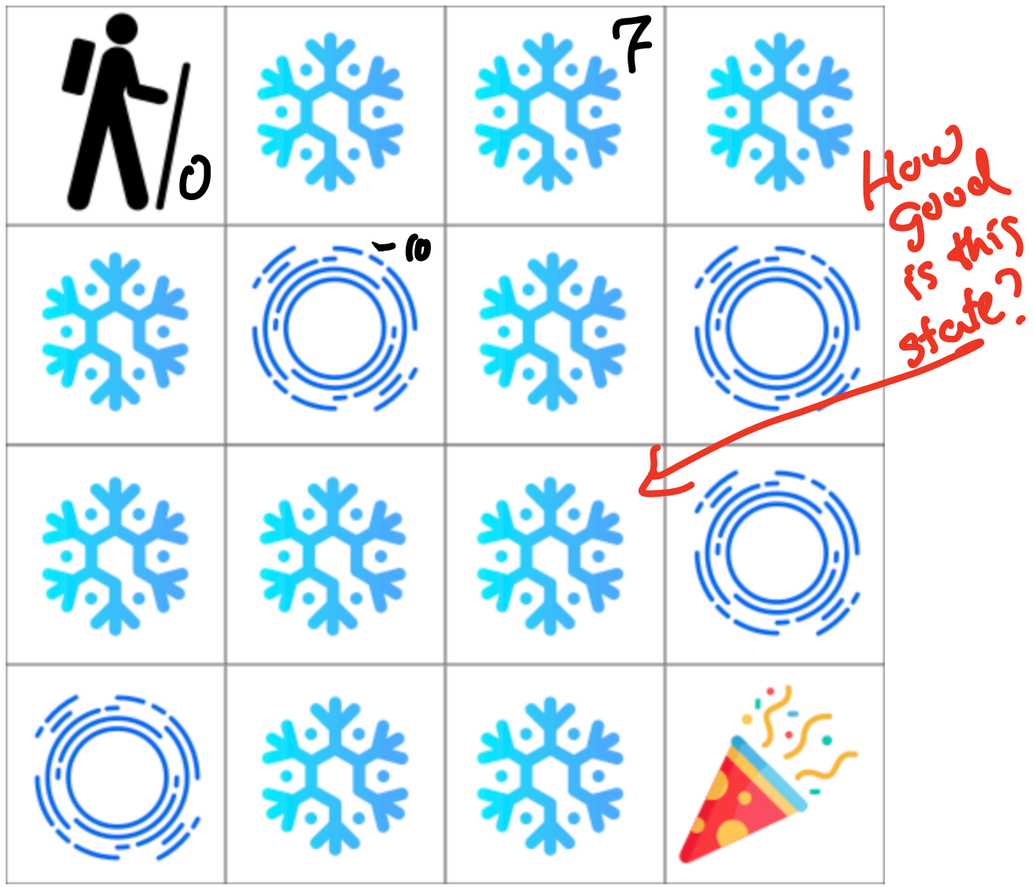
Form 23/25

Proposal

Project

Q Learning

Goal: Go to good states



What makes a state good?

$$\tau = (s_0, a_0, s_1, a_1, s_2, a_2, \dots)$$

"gain" $r_t = r(s_t, a_t)$

$$G_t = r_t + r_{t+1}\gamma + r_{t+2}\gamma^2 + r_{t+3}\gamma^3 + \dots$$

$(s_t, a_t, s_{t+1}, a_{t+1}, \dots)$

$\alpha \quad \gamma < 1 \quad \gamma \approx .99$

Value of state s is

$$\mathbb{E}_{\tau \sim \pi}[G_t \text{ when } s_t = s]$$

↑ expected discounted reward when we start in s and follow policy π

We need to know how to get to good states.

Need action-value function

$Q^{\pi}(s,a)$ = the expected long term reward of taking action a in state s then following policy π

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}\left[G_t \mid s_t = s, a_t = a\right]$$

$$\text{best action} = \underset{a}{\operatorname{argmax}}\ Q^{\pi}(s,a)$$

Bellman Equation

$$Q^{\pi}(s,a) = \mathbb{E}\left[G_t \mid s_t = s, a_t = a\right]$$

$$= \mathbb{E}\left[\sum_{i=0}^{\infty} \gamma^i r_{t+i} \mid s_t = s, a_t = a\right]$$

$$= \mathbb{E}\left[r_t + \sum_{i=1}^{\infty} \gamma^i r_{t+i} \mid s_t = s, a_t = a\right]$$

$$= r_t + \gamma \mathbb{E}\left[\sum_{i=1}^{\infty} r_{t+i} \gamma^{i-1} \mid s_{t+1} = s', a_{t+1} \text{ is best action in state } s'\right]$$

$$\sum_{i=0}^{\infty} r_{t+1+i} \gamma^i \leftarrow r_{t+1} \cdot 1 + r_{t+2} \cdot \gamma + r_{t+3} \gamma^2 + \dots$$

$$= r_t + \gamma \mathbb{E}\left[G_{t+1} \mid s_{t+1} = s', a_{t+1} \text{ is best}\right]$$

$$= r_t + \gamma \max_{a'} \mathbb{E}\left[G_{t+1} \mid s_{t+1} = s', a_{t+1} = a'\right] = r_t + \gamma \max_{a'} Q(s', a')$$

$\tilde{\pi}$

deterministic

$r_t = r(s_t, a_t)$

s' is state if we take a in state s

Bellman Equation

(s, a, r, s')

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

want

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

fixed Q

$$\mathcal{L} = \frac{1}{2}\left(Q(s,a) - \underbrace{\left(r + \gamma \max_{a'} Q(s', a')\right)}_{\text{target}} \right)^2$$

$$\frac{\partial \mathcal{L}}{\partial Q(s,a)} = \left(Q(s,a) - \text{target} \right) \frac{\partial Q(s,a)}{\partial Q(s,a)}$$

Exploration vs Exploitation

Initialize s $\quad (s, a, s', r)$

Repeat until done:

1. Choose $\underset{a}{\text{argmax}}\ Q(s,a)$ with prob. $1-\epsilon$, random else
2. Take a and observe s', r
3. $Q(s,a) \mathrel{-}= \eta\,(Q(s,a) - \text{target})$

$$\theta \mathrel{-}= \eta \frac{\partial \mathcal{L}}{\partial \theta} = \eta\,(Q(s,a) - \text{target}) \frac{\partial Q(s,a)}{\partial \theta}$$

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

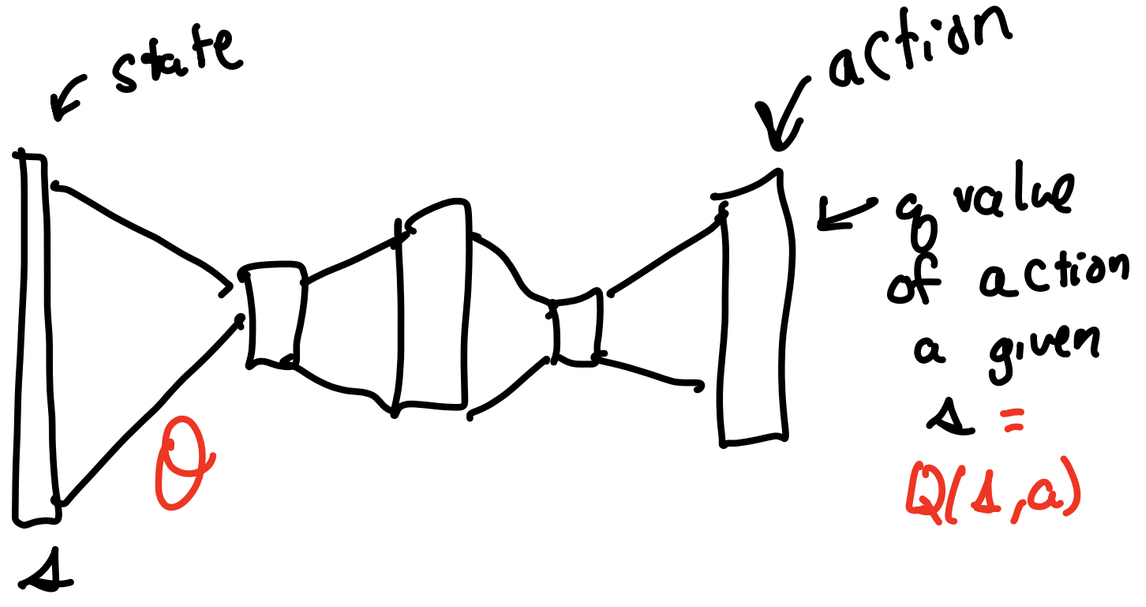
$Q =$

states: 3^{361} (rows)

actions: 361 (columns)

$$\begin{bmatrix} & \\ & \\ & \end{bmatrix}$$

Q = neural net ☺



Policy

- no intermediate value
- Rock paper scissors need distribution

Deep Q Learning

- Estimate value (helpful in general)
- More sample efficient