

DEEP REINFORCEMENT LEARNING

Use deep neural network

deep → many layers

Attempt

↓
make mistake

↓
learn

↓
Master

(dzreni świstaka) → Kod miesniestelmoj
↓
Na skvoju jutna
(cety czas pomezywa ten sam dzreni)

Attempt → make mistake → learn with neural network → Master

I REINFORCEMENT LEARNING

Flappy Birds and MARIO

- State - stan w którym znajduje się aktualnie agent np miejsce w danym momencie
- Action - zbiór wszystkich możliwych ruchów które agent może wykonać.

Flappy Bird - TAP or NOT to tap

Mario - Jump, go Right, go Down

- Reward - punkty które dostaje agent za dobre wykonaną akcje lub traci za błądną akcje

~~Agent~~ Mario + dodatkowe punkty
Flappy - utrata życia

- Agent - obiekt ~~z~~ który wykonuje akcje

w Flappy - ptaszek

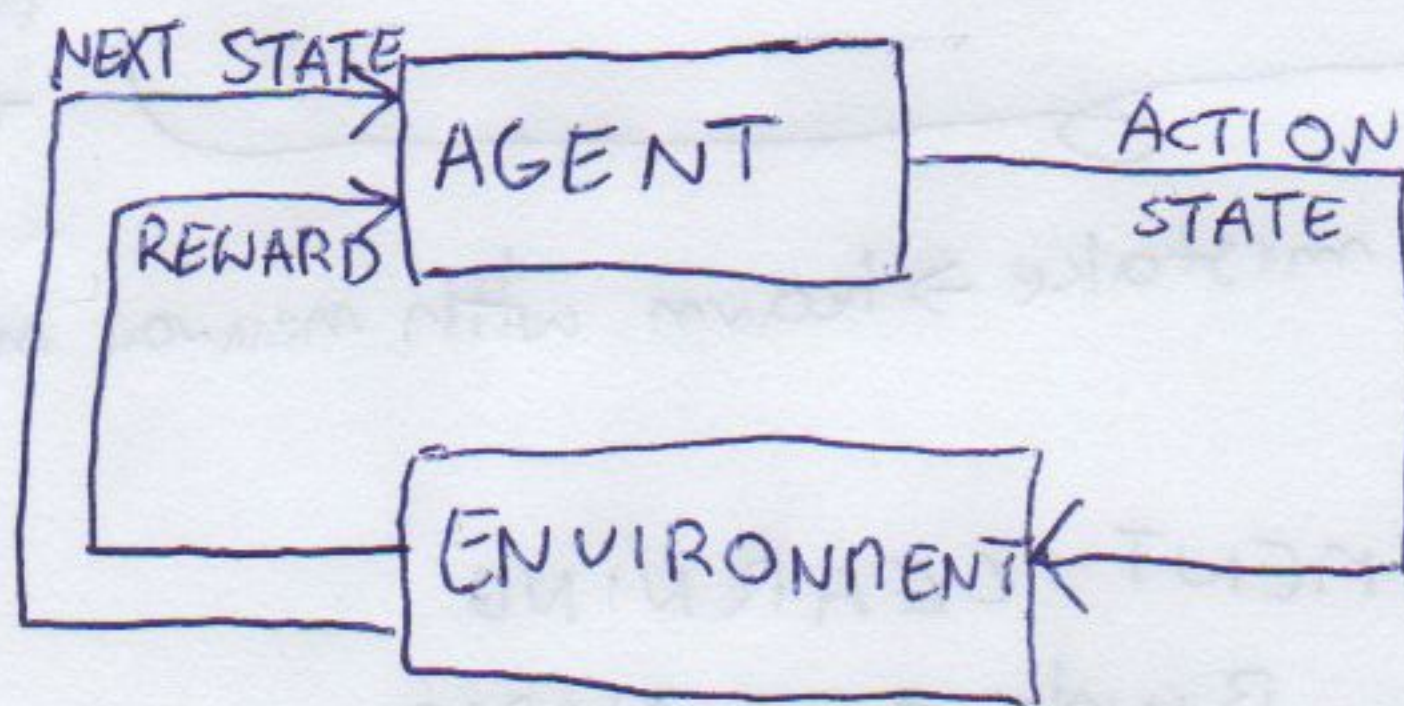
Mario - Mario

Discount factor - przyszłe nagrody mają mniej wagi

$$\text{actual reward} = \gamma^m \cdot \text{reward}$$

$$\gamma \in (0, 1)$$

Environment - stan agenta i akcja są wejściem do środowiska a dla człowieka np. prawo jazdy
Zwraca nagrody i kolejny stan



Value - oczekiwana długo czasowa zwrot ze zmiłko

Policy - Strategia jaką podejmuje agent w celu wykonania kolejnej akcji bazując na aktualnym stanie

First important problem

Domain selection

We can imagine an autonomous reinforcement learning Agent as Blind person with cane and ears

Wygodę środowisko jest określone i znane więc problem jest nieistotny ale np. autonomus cans kamery
Wybiera co jest potrzebne... np. GPS, sensory

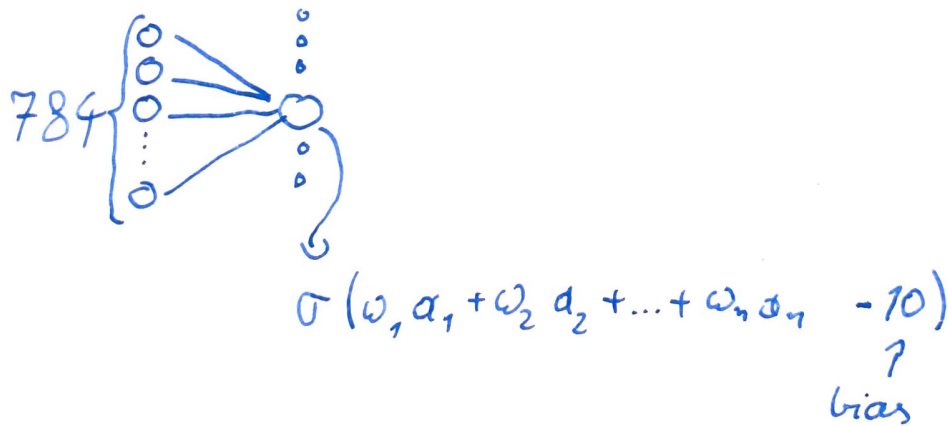
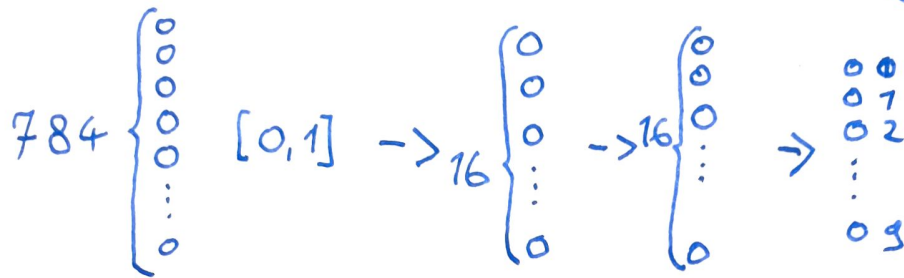
Reward distribution

Nagroda zależy od stanu, np. postąpienie 25 letniej osoby to co innego mając 18 a 90 lat.

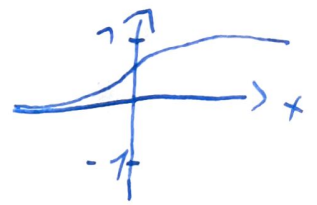
Akcja „Fine” inaczej w teatrze, a inaczej w drodze zotmiemy
Potrzebny kontekst

Deep neural networks

$$3 = 3 = 3 \neq 6 \neq 0$$



$-x_1 ; +x_2$
 sigmoid $\sigma(x) = \frac{1}{1+e^{-x}}$



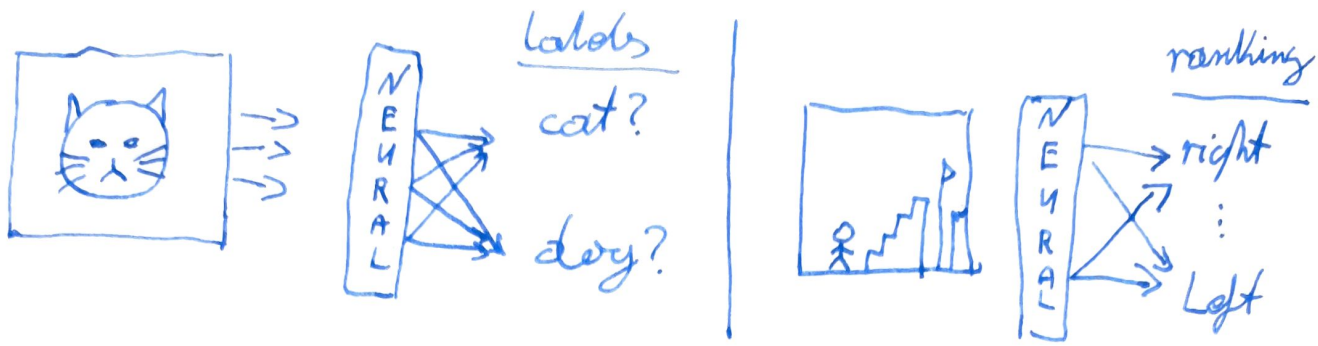
$$\left. \begin{array}{l} 784 \cdot 16 + 16 \cdot 16 + 10 \cdot 16 \\ \text{weights} \\ 16 + 16 + 10 \\ \text{biases} \end{array} \right\} 13,002$$

możliwych do ustawienia parametrów

Problems I $\left\{ \begin{array}{l} \text{---} \\ \text{---} \\ \text{---} \\ \vdots \\ \text{---} \\ \text{---} \\ \text{---} \end{array} \right. \Rightarrow \text{Sure!}$
 I#j #5!

II draw 3 $\Rightarrow ?$

Neural network in reinforcement learning



can be used to recognize current state
and to predict reward for actions (possible).

Visualization

Mario \leq 1000 copies run parallelly
digging through possibilities

Best examples propagate their characteristic
to future generations