

06 - Evoluční návrh neuronové sítě v modelové úloze

Petr Buchal

Vysoké učení technické v Brně, Fakulta informačních technologií
Božetěchova 1/2. 612 66 Brno – Královo Pole
xbucha02@stud.fit.vutbr.cz



10. Květen, 2020

Popis varianty:

Pro zvolený typ neuronové sítě a problém rozumné složitosti, který má řešit (můžete si vymyslet vlastní, třeba i jen teoretický), použijte evoluční algoritmus k natrénování neuronové sítě, případně i návrhu/optimalizaci její struktury. Prezentujte statistické výsledky evoluce a vybrané výsledky řešení.

Doporučení: Zvažte popužití SW z úlohy 5. cvičení EVO a použijte jej jako základ pro vaše řešení, přičemž zvolený problém MUSÍ být jiný než úloha ve cvičení.

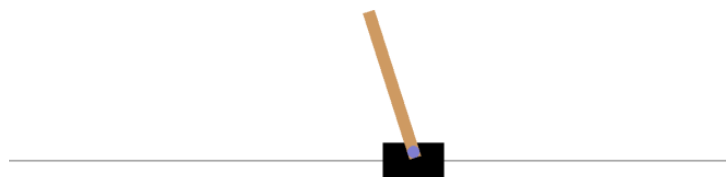
Algoritmus 1 DQN [3]

```

Inicializuj paměť  $D$  s kapacitou  $N$ 
Inicializuj neuronovou síť  $Q$  s náhodnými váhami
for  $epizoda = 1, M$  do
    Inicializuj prostředí a pozoruj počáteční stav  $s_t$ 
    for  $krok = 1, T$  do
        S pravděpodobností  $\epsilon$  vyber náhodnou akci  $a_t$ 
        jinak  $a_t = \operatorname{argmax}(Q(s_t))$ 
        Proveď akci  $a_t$ , pozoruj odměnu  $r_t$  a nový stav  $s_{t+1}$ 
        Ulož vzpomínku  $(s_t, a_t, r_t, s_{t+1})$  do paměti  $D$ 
        Vezmi náhodný vzorek vzpomínek  $(s_i, a_i, r_i, s_{i+1})$  z paměti  $D$ 
        
$$l_i = \begin{cases} r_i & \text{pro stav } s_i, \text{ který je koncový} \\ r_i + \gamma * \max(Q(s_{i+1})) & \text{pro stav } s_i, \text{ který není koncový} \end{cases}$$

        Za použití  $s_i$  jako trénovacích dat a  $l_i$  jako štítků trénuj  $Q$ 
         $s_t = s_{t+1}$ 
    end for
end for

```

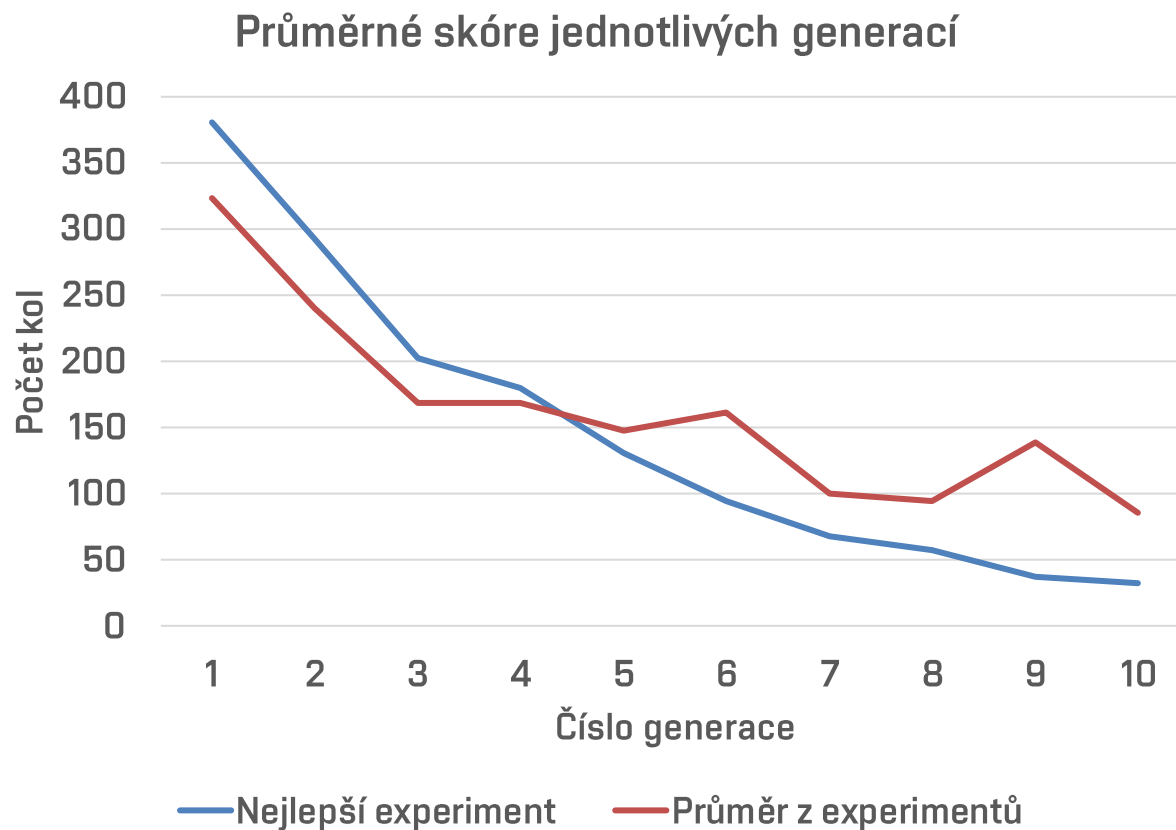


Název prostředí	Odměny	Řešení prostředí
CartPole-v0	+1 za krok s udržením rovnováhy -1 za krok se ztrátou rovnováhy	Dosažení průměrného skóre alespoň 195 ze 100 epizod

Název prostředí	Informace o prostředí	Možné akce	Maximální počet kroků
CartPole-v0	Pozice vozíku Rychlost vozíku Úhel tyče Rychlost tyče na špičce	Pohyb doleva Pohyb doprava	200

```
1 {  
2   "nb_neurons": [4, 8, 16, 32, 64],  
3   "nb_layers": [1, 2, 3, 4],  
4   "activation": ["relu", "elu", "tanh", "sigmoid"],  
5   "optimizer": ["rmsprop", "adam", "sgd", "adagrad",  
6     "adadelata", "adamax", "nadam"]  
7 }
```

nn_parameters.json

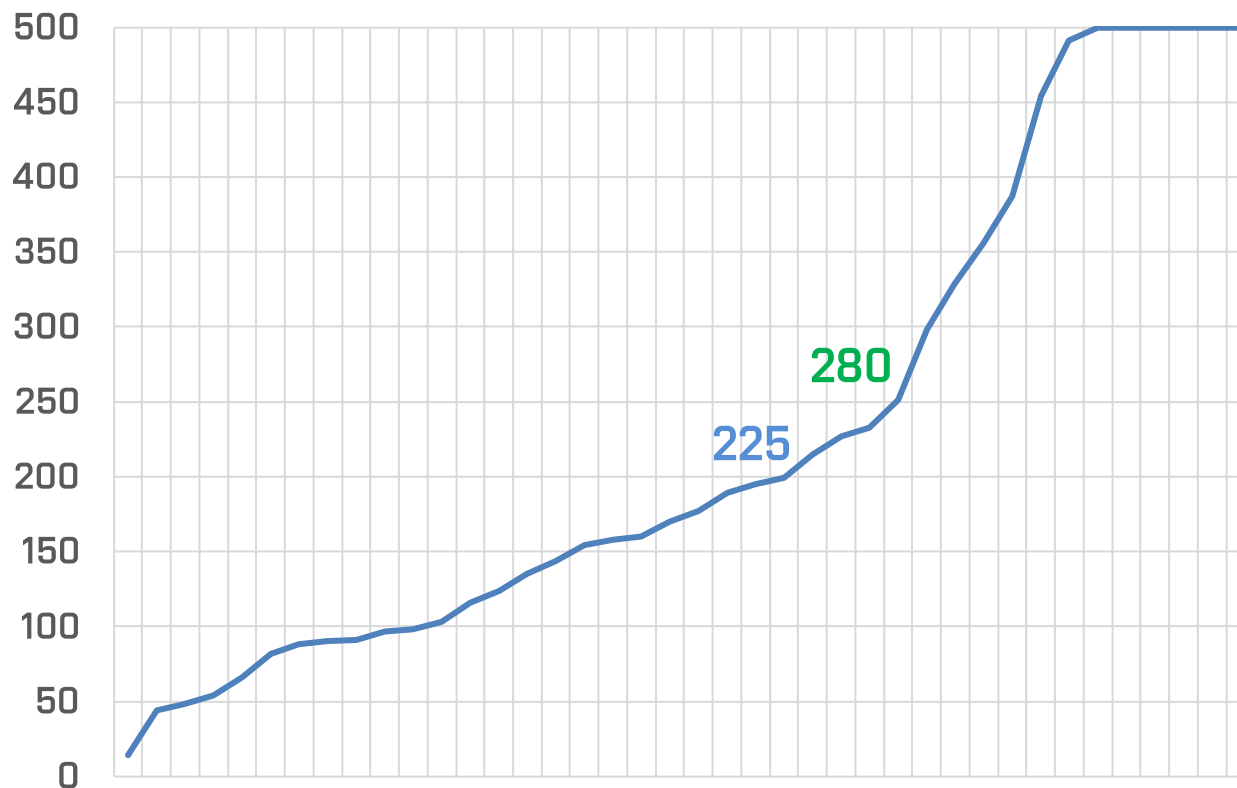


Layer (type)	Output Shape	Param #
dense (Dense)	(None, 16)	80
dense_1 (Dense)	(None, 64)	1088
dense_2 (Dense)	(None, 16)	1040
dense_3 (Dense)	(None, 2)	34
Total params: 2,242		
Trainable params: 2,242		
Non-trainable params: 0		

tahn
elu
sigmoid
linear

adam

Počet her vítězné architektury potřebných ke zvládnutí prostředí CartPole-v0



Balíčky:

- Tensorflow
- Logging
- Argparse
- Numpy
- Gym

`python3 main.py`