Głębokie uczenie ze wzmocnieniem

Piotr Duch

pduch@iis.p.lodz.pl
Instytut Informatyki Stosowanej
Politechnika Łódzka

Lato 2023

Plan wykładu

1 Wprowadzenie

2 Głębokie uczenie ze wzmocnieniem

3 Artykuły

Informacje ogólne:

- Materiały wykładowe oraz laboratoryjne dostępne są na stronie (pduch.iis.p.lodz.pl).
- Literatura podstawowa:
  - Richard S. Sutton, and Andrew G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
  - Morales, Miguel. *Grokking deep reinforcement learning*. Simon and Schuster, 2020.
- Wykłady uzupełniające:
  - RL Course by David Silver - https://www.youtube.com
  - CS 188: Artificial Intelligence by Pieter Abbeel (wykład 10 i 11)- https://www.youtube.com/watch?v=lXuHxkpO5E8
- Materiały dodatkowe:
  - Practical RL Course by Yandex School of Data Analysis - https://github.com/yandexdataschool/Practical_RL
  - CS 188: Introduction to Artificial Intelligence by Berkeley University of California - https://inst.eecs.berkeley.edu/ cs188/fa19/project3/

Informacje ogólne - zaliczenie:

- Laboratorium:
  - Zadania do wykonania w Pythonie (a jednak 5) notebooki pythonowe),
  - Ocena końcowa - 66% - 73% ocena 3, 73% - 80% ocena 3.5, 80% - 87% ocena 4, 87% - 94% ocena 4.5, 94% i wyżej - 5.
- Wykład - prezentacja artykułów.
- Projekt - VizDoom - multi-player mode.
- Kontakt:
  - poprzez platformę MS Teams na chacie indywidualnym,
  - mailowo: pduch@iis.p.lodz.pl.

Informacje szczegółowe - plan działania:

- *Deep Q-Networks* - Notebook Pythonowy 4 (Pliki dodatkowe do notebooków) 15.03.2024.
- *Double Deep Q-Networks* - Notebook Pythonowy 5 29.03.2024.
- *REINFORCE* - Notebook Pythonowy 6 12.04.2024.
- *Actor-Critic* - Notebook Pythonowy 7 26.04.2024.

Wprowadzenie

Uczenie ze wzmocnieniem - wprowadzenie

Metody

- Uczenie pasywne:
  - Ocena strategii (ang. *Policy Evaluation*)
  - Polepszanie strategii (ang. *Policy Improvement*)
  - Iteracyjne doskonalenie strategii (ang. *Policy Iteration*)
  - Iteracyjne obliczanie funkcji wartości (ang. *Value Iteration*)

Uczenie ze wzmocnieniem - wprowadzenie

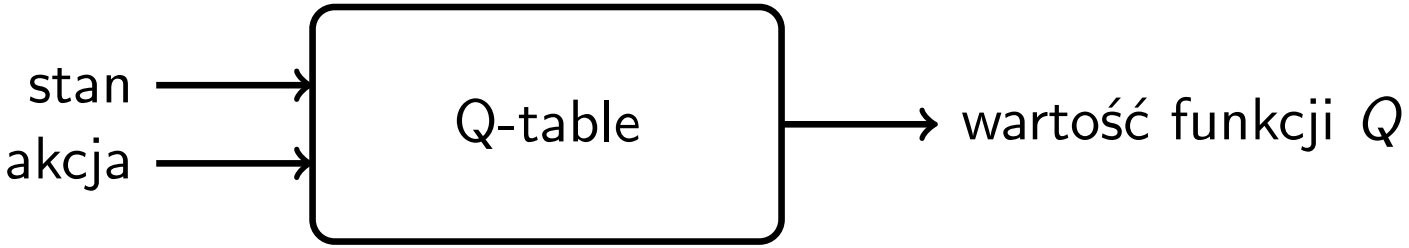
Metody cd.

- Uczenie aktywne:
  - Metody różnic czasowych (ang. *Temporal Difference Learning*)
    - Monte Carlo
    - Q-Learning
    - SARSA
  - Metody aproksymacyjne
    - Aproksymacja funkcji wartości (ang. *Approximate Q-Learning*)
  - Metody wykorzystujące głebokie sieci neuronowe
    - Deep Q-Learning
    - Double Q-Learning
    - Actor-Critic
    - REINFORCE
    - Policy Gradient

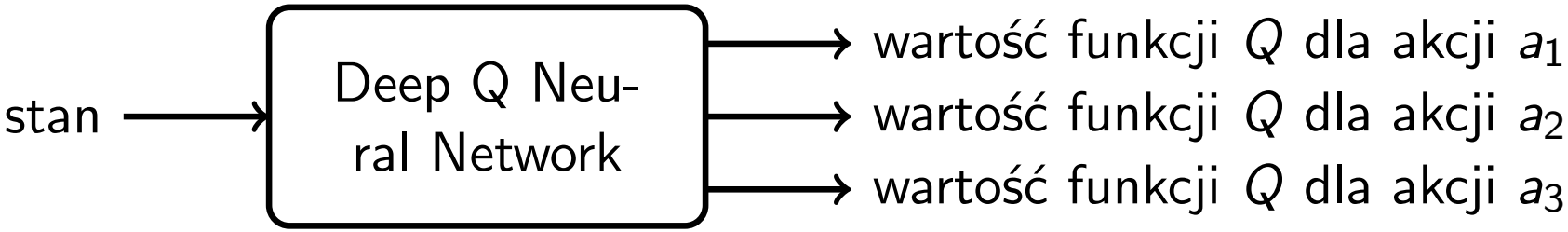
Głębokie uczenie ze wzmocnieniem

Głębokie uczenie ze wzmocnieniem

Deep Q Neural Network



Rysunek 1: Q-Learning



Rysunek 2: Deep Q Neural Network

Głębokie uczenie ze wzmocnieniem

Deep Q Neural Network

Sposób działania:

- na wejście sieci podawany jest aktualny stan,
- następna akcja jest wybierana na podstawie odpowiedzi sieci (akcja o największej wartości),
- funkcją kosztu jest błąd średniokwadratowy z wartości przewidzianej przez sieć, a tej docelowej (wyznaczonej za pomocą równania Bellmana) - $R + \gamma max_a(Q(s', a))$,
- jako wartość oczekiwaną funkcji *predict* podajemy wektor wartości Q zwrócony przez sieć dla danego stanu, z zaktualizowaną wartością dla wybranej akcji.

Głębokie uczenie ze wzmocnieniem

Deep Q Neural Network

Problemy z podejściem DQN:

- sekwencyjnie skorelowane dane - może wpływać na zbieżność i wydajność sieci,
- niestabilność dystrybucji danych z powodu zmian strategii - strategia może nie być zbieżna lub oscylować.

Głębokie uczenie ze wzmocnieniem

Deep Q Neural Network - Experience Replay

Rozwiązanie problemu sekwencyjnie skorelowanych danych - zapamiętanie całej serii wykonanych działań, to znaczy stanu (s), akcji podjętej w tym stanie (a), otrzymanej nagrody (r) oraz kolejnego stanu (s'). Uczenie sieci na podstawie losowo dobranych próbek z zebranych danych.

Głębokie uczenie ze wzmocnieniem

Deep Q Neural Network - Experience Replay

Sposób działania:

1. zebranie zbioru danych w postaci krotek - (s, a, r, s') na podstawie akcji podjętych przez agenta (sieć), korzystając z algorytmu ϵ-zachłannego,
2. wybór losowych krotek z zebranego zbioru,
3. trenowanie modelu sieci na podstawie wybranych danych,
4. podejmowanie nowych akcji na podstawie zaktualizowanego modelu i algorytmu ϵ-zachłannego,
5. powrót do punktu 2.

Głębokie uczenie ze wzmocnieniem

Deep Q Neural Network - Prioritized Experience Replay

Sposób działania:

1. wybierz akcję a w stanie s i zapamiętaj nowy stan (s') oraz otrzymaną nagrodę (r),
2. wyznacz za pomocą sieci wartość dla aktualnego stanu $Q(s, a)$,
3. wyznacz za pomocą sieci oczekiwaną wartość dla aktualnego stanu $target = r + \gamma * max'_a Q(s', a')$,
4. wyznacz wartość absoltuną błędu dla analizowanego stanu $atd_err = abs(Q(s, a) - target)$,
5. dodaj krotkę (s, a, r, s', atd_err) do bufora,
6. wykorzystaj dane z bufora do uczenia sieci.

Głębokie uczenie ze wzmocnieniem

Deep Q Neural Network - Prioritized Experience Replay

Opcje priorytetyzacji:

1. *Proportional prioritization*,
2. *Rank-based prioritization*,
3. *Weighted importance-sampling*.

Głębokie uczenie ze wzmocnieniem

Deep Q Neural Network - Prioritized Experience Replay

Proportional prioritization:

$$p_i = |\delta_i| + \epsilon \tag{1}$$

$$P(i) = \frac{p_i^\alpha}{\sum_k p_k^\alpha} \tag{2}$$

Głębokie uczenie ze wzmocnieniem

Deep Q Neural Network - Prioritized Experience Replay

Rank-based prioritization:

$$\theta = \theta - \alpha \frac{dloss}{d\theta} w \tag{3}$$

$$w(i) = (\frac{1}{N} * \frac{1}{P(i)})^b \tag{4}$$

Głębokie uczenie ze wzmocnieniem

Deep Q Neural Network - Prioritized Experience Replay

Weighted importance-sampling:

$$p(i) = \frac{1}{rank(1)} \tag{5}$$

$$P(i) = \frac{p_i^\alpha}{\sum_k p_k^\alpha} \tag{6}$$

Głębokie uczenie ze wzmocnieniem

Double Deep Q Neural Network

Rozwiązanie problemu niestabilności dystrybucji danych z powodu zmian strategii, ocena wybranej akcji odbywa się za pomocą innej sieci, niż sam wybór akcji.

Głębokie uczenie ze wzmocnieniem

Double Deep Q Neural Network

Korzystanie z dwóch sieci:

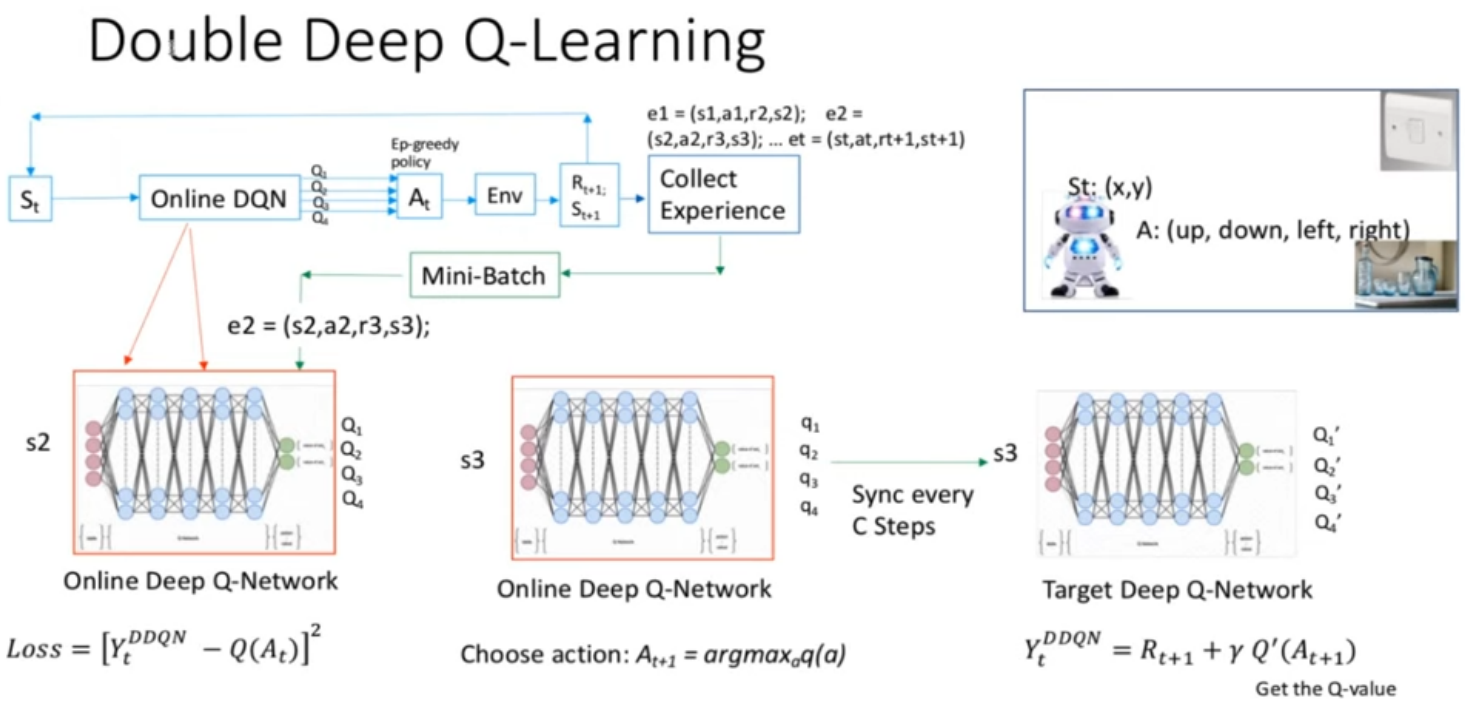
- sieć Q' - wykorzystywana do wyboru akcji podejmowanych przez agenta,
- sieć Q - wykorzystywana do oszacowania wybranej akcji.

Funkcją kosztu jest błąd średniokwadratowy z wartości przewidzianej przez sieć Q', a tej oszacowanej prze sieć Q (wyznaczonej za pomocą równania Bellmana):

$$Q^*(s_t, a_t) \approx r_t + \gamma Q(s_{t+1}, argmax_{a'} Q'(s_{t+1}, a'))$$

Głębokie uczenie ze wzmocnieniem

Double Deep Q Neural Network



Rysunek 3: Double Deep Q-Learning

[DDQN] Deep Reinforcement Learning with Double Q-learning — TDLS Foundational)

Uczenie aktywne

Double Deep Q Neural Network

Double Q-Learning

Initialize primary network Q_θ, target network Q'_θ, replay buffer D, $\tau << 1$.
For each iteration do:
    For each environment step do:
        Observe state s and select action a using network Q_θ:
        Take action a, observe r, s'
        Store (s, a, r, s') in replay buffer D
    For each update step do:
        Sample $e = (s, a, r, s)$ from D
        Compute target Q value:

$$Q^*(s, a) \approx r + \gamma Q(s', argmax_{a'} Q'_\theta(s', a'))$$

        Perform gradient descent step on $(Q^*(s, a) - Q_\theta(s, a))^2$
        Update target network parameters:

$$\theta' \leftarrow \tau * \theta + (1 - \tau) * \theta'$$

Mnih, V., Kavukcuoglu, K., Silver, D. et al. Human-level control through deep reinforcement learning. Nature 518, 529–533 (2015).

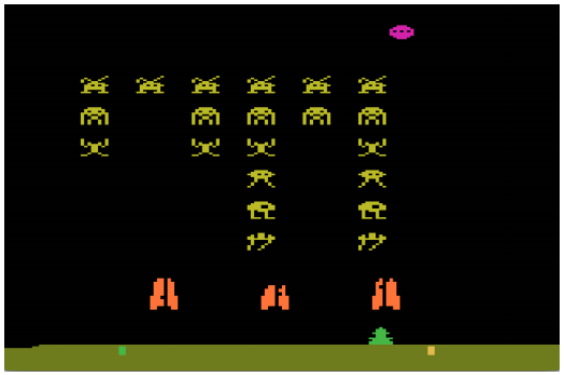
Głębokie uczenie ze wzmocnieniem

Materiały uzupełniające

- Książka *Grokking Deep Reinforcement Learning*, Miguel Morales, October 2020.
  - Rozdziały 9 - *More stable value-based methods* - DQN, DDQN, Experience Replay.
  - Rozdziały 10 - *Sample-efficient value-based methods* - Prioritized Experience Replay.

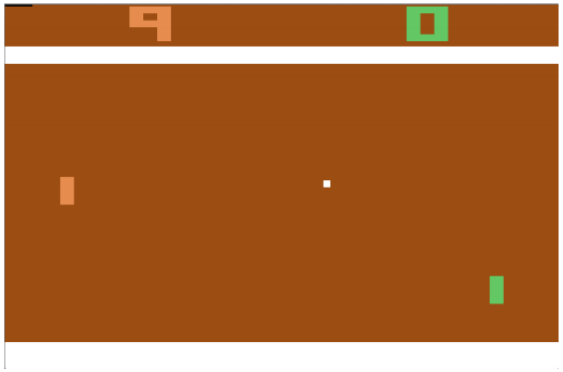
Głębokie uczenie ze wzmocnieniem

Double Deep Q Neural Network



(a) Space Invaders

(b) Beam Rider



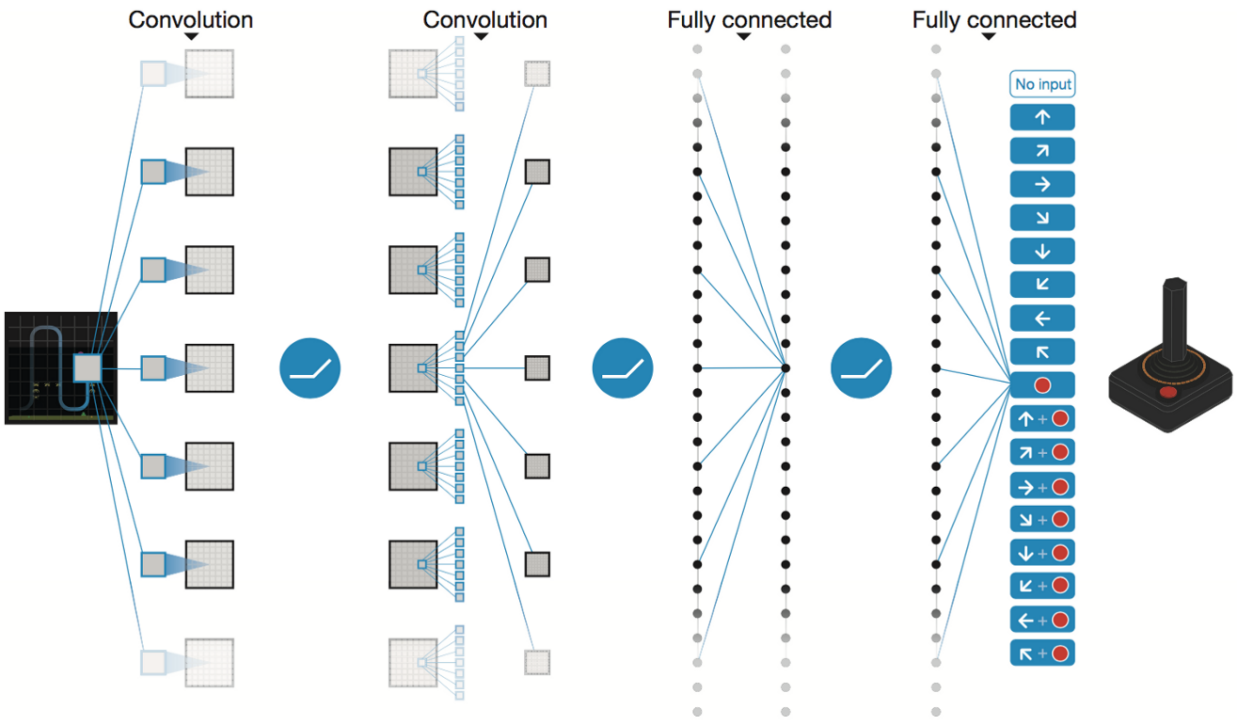
(c) Pong

(d) Breakout

Rysunek 4: Mnih, Volodymyr, et al. "Playing atari with deep reinforcement learning." (2013).

Głębokie uczenie ze wzmocnieniem

Double Deep Q Neural Network

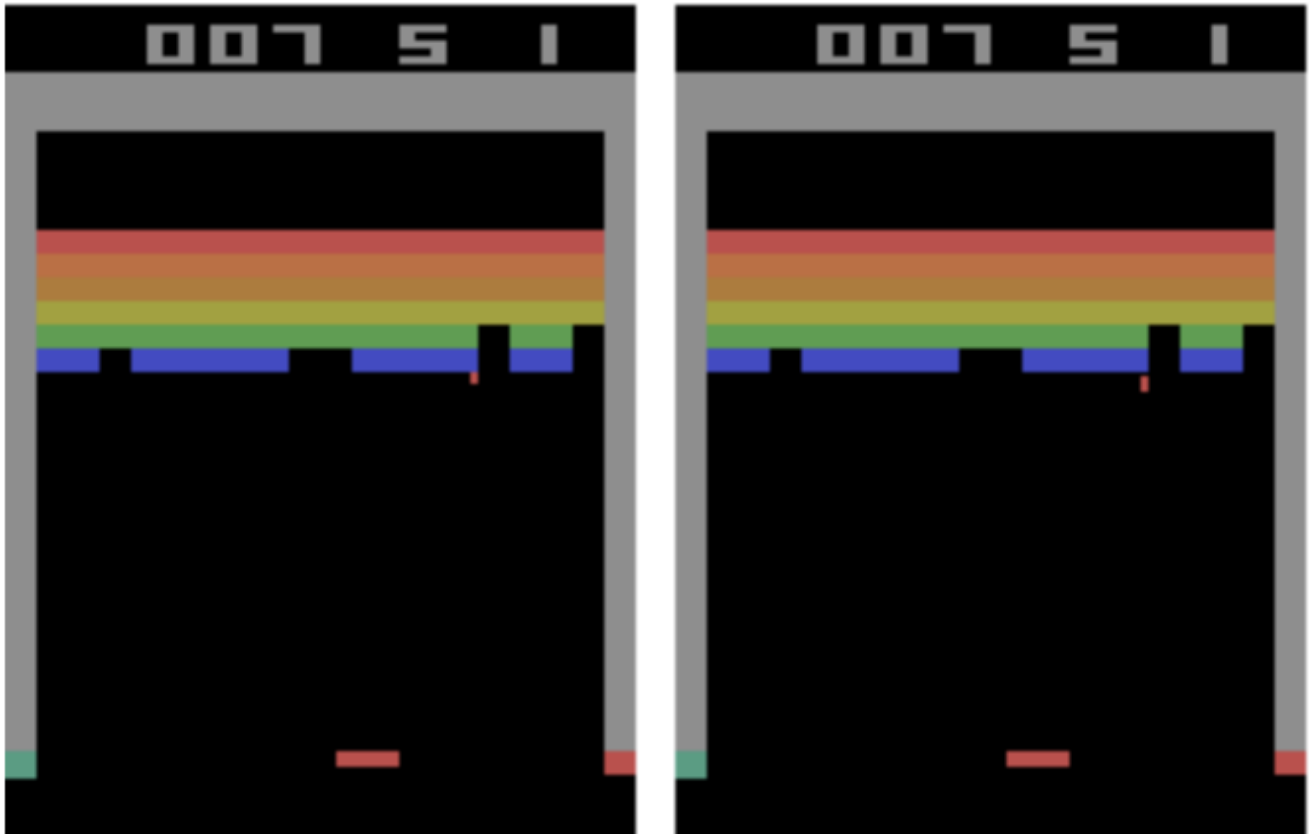


Rysunek 5: Network Architecture

Mnih, V., Kavukcuoglu, K., Silver, D. et al. Human-level control through deep reinforcement learning. Nature 518, 529–533 (2015).

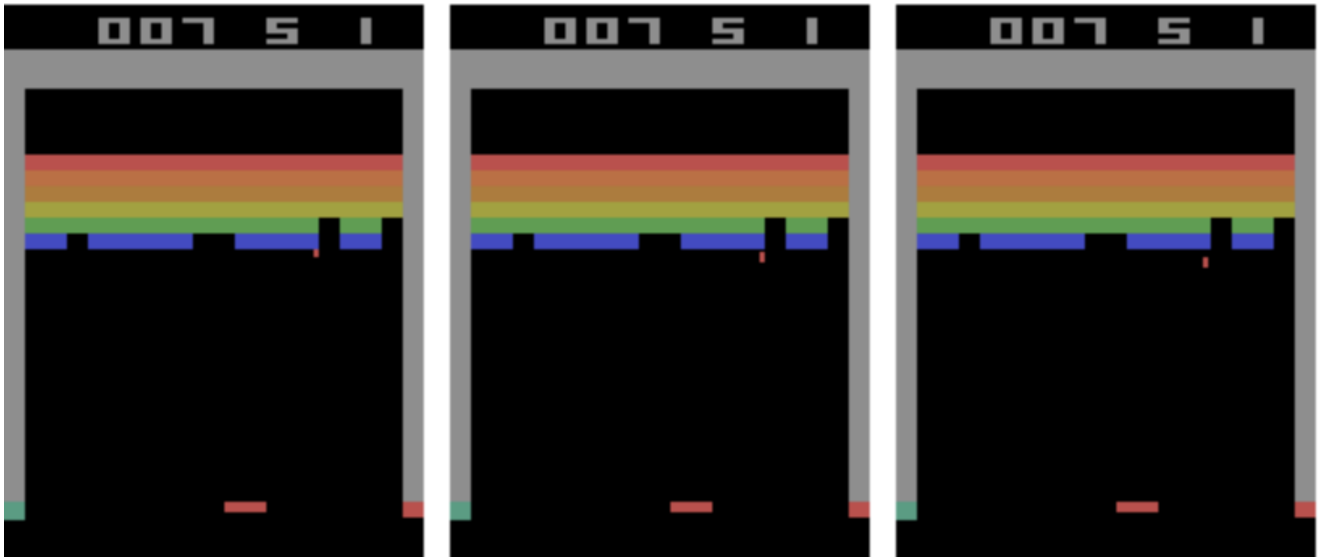
Głębokie uczenie ze wzmocnieniem

Double Deep Q Neural Network



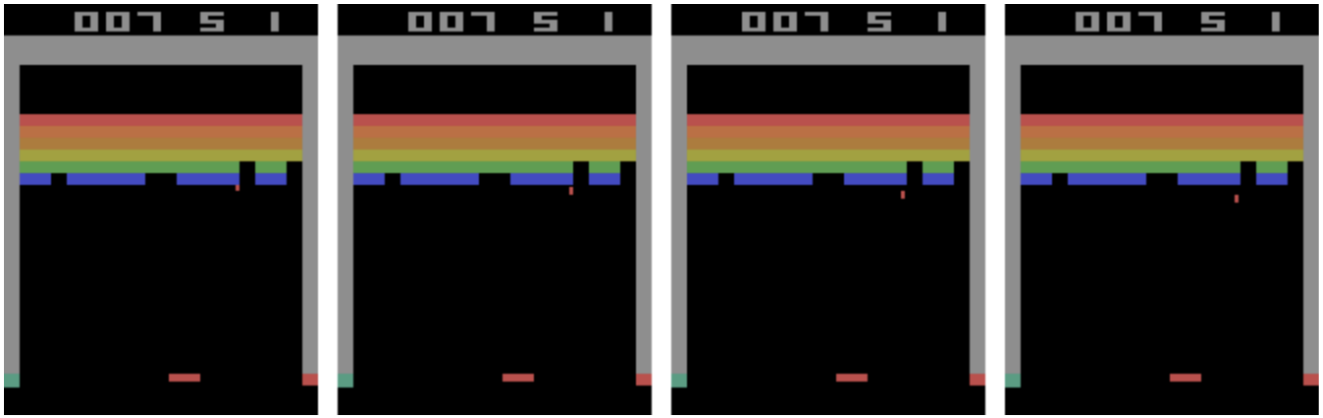
Głębokie uczenie ze wzmocnieniem

Double Deep Q Neural Network



Głębokie uczenie ze wzmocnieniem

Double Deep Q Neural Network



Głębokie uczenie ze wzmocnieniem

Double Deep Q Neural Network

Przetwarzania wstępne obrazów:

- wycięcie istotnego fragmentu obrazu,
- zamiana na odcienie szarości,
- zmniejszenie obrazu.

Głębokie uczenie ze wzmocnieniem

Double Deep Q Neural Network

Hyperparameter	Value	Description
minibatch size	32	Number of training cases over which each stochastic gradient descent (SGD) update is computed.
replay memory size	1000000	SGD updates are sampled from this number of most recent frames.
agent history length	4	The number of most recent frames experienced by the agent that are given as input to the Q network.
discount factor	0.99	Discount factor gamma used in the Q-learning update.
action repeat	4	Repeat each action selected by the agent this many times. Using a value of 4 results in the agent seeing only every 4th input frame.
update frequency	4	The number of actions selected by the agent between successive SGD updates. Using a value of 4 results in the agent selecting 4 actions between each pair of successive updates.
learning rate	0.00025	The learning rate used by RMSProp.
gradient momentum	0.95	Gradient momentum used by RMSProp.
squared gradient momentum	0.95	Squared gradient (denominator) momentum used by RMSProp.
min squared gradient	0.01	Constant added to the squared gradient in the denominator of the RMSProp update.
initial exploration	1	Initial value of ε in ε-greedy exploration.
final exploration	0.1	Final value of ε in ε-greedy exploration.
final exploration frame	1000000	The number of frames over which the initial value of ε is linearly annealed to its final value.
replay start size	50000	A uniform random policy is run for this number of frames before learning starts and the resulting experience is used to populate the replay memory.
no-op max	30	Maximum number of "do nothing" actions to be performed by the agent at the start of an episode.

Rysunek 6: Przykładowe parametry sieci

Mnih, V., Kavukcuoglu, K., Silver, D. et al. Human-level control through deep reinforcement learning. Nature 518, 529–533 (2015).

Głębokie uczenie ze wzmocnieniem

Krótkie podsumowanie

- Uczenie pasywne (ang. *model-based learning*):
  - Iteracyjne doskonalenie strategii (ang. *Policy Iteration*).
  - Iteracyjne obliczanie funkcji wartości (ang. *Value Iteration*).

Głębokie uczenie ze wzmocnieniem

Krótkie podsumowanie

- Uczenie pasywne (ang. *model-based learning*):
  - Iteracyjne doskonalenie strategii (ang. *Policy Iteration*).
  - Iteracyjne obliczanie funkcji wartości (ang. *Value Iteration*).
- Uczenie aktywne (ang. *model-free learning*):
  - Algorytmy bazujące na wyznaczaniu funkcji wartości:
    - Q-Learning (*off-policy*).
    - SARSA (*on-policy*).
    - Metody aproksymacyjne (Liniowa kombinacja cech, *Deep Q Networks*).

Głębokie uczenie ze wzmocnieniem

Krótkie podsumowanie

- Uczenie pasywne (ang. *model-based learning*):
  - Iteracyjne doskonalenie strategii (ang. *Policy Iteration*).
  - Iteracyjne obliczanie funkcji wartości (ang. *Value Iteration*).
- Uczenie aktywne (ang. *model-free learning*):
  - Algorytmy bazujące na wyznaczaniu funkcji wartości:
    - Q-Learning (*off-policy*).
    - SARSA (*on-policy*).
    - Metody aproksymacyjne (Liniowa kombinacja cech, *Deep Q Networks*).
  - Algorytmy bazujące na wyznaczaniu strategii (ang. *REINFORCE*).

Głębokie uczenie ze wzmocnieniem

Policy Gradients

Dotychczasowe podejście:

- Aproksymacja funkcji wartości:

$$v_*(s) \approx v_\pi(s)$$

$$q_*(s, a) \approx q_\pi(s, a)$$

- Wyznaczanie strategii na podstawie tych funkcji (algorytm zachłanny, ϵ-zachłanny).

Nowe podejście:

- Wyznaczanie strategii bezpośrednio:

$$\pi_\theta(s) = p(a|s, \theta)$$

Głębokie uczenie ze wzmocnieniem

Policy Gradients

Zalety optymalizacji strategii:

- Strategia wyznaczana przez metody Q-Learning jest zawsze deterministyczna. Metody te nie mogą opracować strategii stochastycznej, co może być przydatne w niektórych środowiskach. W takim przypadku wykorzystywane są algorytmy, np. ϵ-zachłanne, ale to podejście nie zawsze jest wystarczające i optymalne.
- Metody należące do grupy optymalizujących strategie mogą być zastosowane w przypadku środowisk, gdzie akcje mają charakter ciągły.
- W sposób ciągły poprawiana jest strategia, a w przypadku metod Q-Learning aproksymowana jest funkcja wartości, a dopiero na jej podstawie wyznaczana jest strategia.

Głębokie uczenie ze wzmocnieniem

Policy Gradients

- Algorytmy optymalizacji strategii należą do dziedziny algorytmów optymalizacyjnych.
- Celem jest znalezienie takich wartości parametrów θ, które będą dawały najwyższy wynik funkcji $J(\theta)$.
- Metody nie wykorzystujące gradientów (np. algorytmy genetyczne).
- Metody wykorzystujące gradient (np. metoda gradientu prostego).

Głębokie uczenie ze wzmocnieniem

Policy Gradients

$$\theta_{t+1} = \theta_t + \alpha \nabla \pi_{\theta_t}(s, a^*)$$

Głębokie uczenie ze wzmocnieniem

Policy Gradients

$$\theta_{t+1} = \theta_t + \alpha \nabla \pi_{\theta_t}(s, a^*)$$

$$\theta_{t+1} = \theta_t + \alpha \hat{Q}(s, a) \nabla \pi_{\theta_t}(s, a)$$

Głębokie uczenie ze wzmocnieniem

Policy Gradients

$$\theta_{t+1} = \theta_t + \alpha \nabla \pi_{\theta_t}(s, a^*)$$

$$\theta_{t+1} = \theta_t + \alpha \hat{Q}(s, a) \nabla \pi_{\theta_t}(s, a)$$

$$\theta_{t+1} = \theta_t + \alpha \frac{\hat{Q}(s, a) \nabla \pi_{\theta_t}(s, a)}{\pi_\theta(s, a)}$$

Głębokie uczenie ze wzmocnieniem

Policy Gradients

$$\theta_{t+1} = \theta_t + \alpha \nabla \pi_{\theta_t}(s, a^*)$$

$$\theta_{t+1} = \theta_t + \alpha \hat{Q}(s, a) \nabla \pi_{\theta_t}(s, a)$$

$$\theta_{t+1} = \theta_t + \alpha \frac{\hat{Q}(s, a) \nabla \pi_{\theta_t}(s, a)}{\pi_\theta(s, a)}$$

$$\theta_{t+1} = \theta_t + \alpha \hat{Q}(s, a) \nabla_\theta log \pi_{\theta_t}(s, a)$$

Głębokie uczenie ze wzmocnieniem

Monte-Carlo Policy Gradient (REINFORCE)

Aktualizacja parametrów metodą gradientu prostego (ang. *stochastic gradient ascent*).

Wykorzystanie nagrody skumulowanej jako wartości funkcji $Q(\hat{s}, a)$.

$$\Delta\theta_t = \alpha G_t \nabla_\theta log \pi_{\theta_t}(s, a)$$

Głębokie uczenie ze wzmocnieniem

Monte-Carlo Policy Gradient (REINFORCE)

REINFORCE: Monte-Carlo Policy-Gradient Control (episodic) for π_*

Input: a differentiable policy parameterization $\pi(a|s,\theta)$.
Algorithm parameter: step size $\alpha > 0$.
Initialize policy parameter $\theta \in \mathbb{R}$.
For each episode:

Generate an episode $s_0, a_0, r_1, ..., s_{t-1}, a_{t-1}, r_t$, following $\pi(.|.,\theta)$.
Loop for each step of the episode $t = 0, 1, ..., T-1$:

$$G \leftarrow \sum_{k=t+1}^{T} \gamma^{k-t-1} r_k$$

$$\theta \leftarrow \theta + \alpha G_t \nabla_\theta log \pi_\theta(a_t, s_t|\theta)$$

Richard S. Sutton and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Głębokie uczenie ze wzmocnieniem

REINFORCE with Baseline

$$\theta_{t+1} = \theta_t + \alpha(G_t - b(s_t))\nabla_\theta log\pi_{\theta_t}(s_t, a_t) \tag{7}$$

Metody wyznaczania wartości odniesienia:

- *whitening*:

$$G_t^* = \frac{G_t - \mu_G}{\sigma_G} \tag{8}$$

- *learned baseline*:

$$\theta_{t+1} = \theta_t + \alpha(G_t - \hat{v}(s_t, w))\nabla_\theta log\pi_{\theta_t}(s_t, a_t) \tag{9}$$

$$w_{t+1} = w_t + \beta(G_t - \hat{v}(s_t, w))\nabla_v(s_t, w) \tag{10}$$

Głębokie uczenie ze wzmocnieniem

REINFORCE with Baseline

REINFORCE with Baseline (episodic), for estimating $\pi_\theta \approx \pi_*$

Input: a differentiable policy parameterization $\pi(a|s,\theta)$.
Input: a differentiable state-value function parameterization $\upsilon(s,w)$.
Algorithm parameter: step sizes $\alpha > 0$ and $\beta > 0$.
Initialize policy parameter $\theta \in \mathbb{R}$ and state-value weights $w \in \mathbb{R}$.
For each episode:

Generate an episode $s_0, a_0, r_1, ..., s_{t-1}, a_{t-1}, r_t$, following $\pi(.|.,\theta)$.
Loop for each step of the episode $t = 0, 1, ..., T-1$:

$G \leftarrow \sum_{k=t+1}^{T} \gamma^{k-t-1} r_k$
$\delta \leftarrow G - \hat{\upsilon}(s_t, w)$
$w \leftarrow w + \beta\delta\nabla_w \upsilon(s_t, w)$
$\theta \leftarrow \theta + \alpha\delta\nabla_\theta log\pi_\theta(a_t, s_t|\theta)$

Richard S. Sutton and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Głębokie uczenie ze wzmocnieniem

Actor-Critic

- uczy się strategii - $\pi_\theta(s, a)$,
- uczy się funkcji oceny stanu - $V_\theta(s)$,
- wykorzystać $V_\theta(s)$, żeby szybciej nauczyć się strategii $\pi_\theta(s, a)$.

Głębokie uczenie ze wzmocnieniem

Actor-Critic

$$\theta_{t+1} = \theta_t + \alpha(G_{t:t+1} - \hat{v}(s_t, w))\nabla_\theta log\pi_\theta(a_t, s_t|\theta) \tag{11}$$

$$= \theta_t + \alpha(r_{t+1} + \gamma\hat{v}(s_{t+1}, w) - \hat{v}(s_t, w))\nabla_\theta log\pi_\theta(a_t, s_t|\theta) \tag{12}$$

$$= \theta_t + \alpha\delta_t\nabla_\theta log\pi_\theta(a_t, s_t|\theta) \tag{13}$$

Głębokie uczenie ze wzmocnieniem

Actor-Critic

One-step Actor-Critic, for estimating $\pi_\theta \approx \pi_*$

Input: a differentiable policy parameterization $\pi(a|s,\theta)$.
Input: a differentiable state-value function parameterization $\upsilon(s,w)$.
Algorithm parameter: step sizes $\alpha > 0$ and $\beta > 0$.
Initialize policy parameter $\theta \in \mathbb{R}$ and state-value weights $w \in \mathbb{R}$.
For each episode:
    Initialize s (first step of episode).
    Loop while s is not terminal (for each time step):
        Choose action a according to the policy $\pi_\theta(.,s,\theta)$,
        Take action a, observe s', r,
        $\delta \leftarrow r + \gamma\hat{\upsilon}(s',w) - \hat{\upsilon}(s,w)$.
        $w \leftarrow w + \beta\delta\nabla_w\upsilon(s',w)$.
        $\theta \leftarrow \theta + \alpha\delta\nabla_\theta log\pi_\theta(a,s|\theta)$.
        $s \leftarrow s'$.

Richard S. Sutton and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018.

Głębokie uczenie ze wzmocnieniem

Materiały uzupełniające

- Książka *Reinforcement Learning: An Introduction*, Richard S. Sutton and Andrew G. Barto, wydanie drugie, 2018.
  - Rozdziały 13.1 - 13.4 - *Policy Gradient Methods*.
  - Rozdziały 13.5 - *Actor–Critic Methods*.
- Video *Hado Van Hasselt, Advanced Deep Learning & Reinforcement Learning Lectures. Reinforcement Learning 6: Policy Gradients and Actor Critics*.
- Video *RL Course by David Silver - Lecture 7: Policy Gradient Methods*.

Głębokie uczenie ze wzmocnieniem

Model środowiska

Frozen Lake:

S	F	F	F
F	H	F	H
F	F	F	H
H	F	F	G

Stan jest reprezentowany za pomocą pojedynczej wartości z przedziału $< 0, 15 >$ oznaczającej pozycję agenta na planszy.
W przypadku sieci wejściem będzie wektor 16 elementowy, gdzie wartością 1 będzie zaznaczony aktualny stan - pozostałe wartości będą równe 0.

Głębokie uczenie ze wzmocnieniem

Model środowiska

Frozen Lake Extended:

S	F	F	F
F	H	F	H
F	F	F	H
H	F	F	G

Stan jest reprezentowany za pomocą trzech tablic - w pierwszej zaznaczone cyfrą 1 są dziury, w drugiej pozycja gracza, a w trzeciej cel.
W przypadku sieci wejściem będzie wektor złożony z trzech tablic.

Głębokie uczenie ze wzmocnieniem

Model środowiska

OpenAI Gym - CartPole:

Akcje:

- ruch wózka w lewo,
- ruch wózka w prawo.

Stan:

- pozycja wózka,
- prędkość wózka,
- kąt słupa,
- prędkość końcówki słupa.

Nagrody:

- 1 - za każdy krok.

Uczenie ze wzmocnieniem

Kursy online

- Reinforcement Learning:
  - *Practical Reinforcement Learning* - Yandex School of Data Analysis,
  - *Deep Reinforcement Learning* - CS 285 at UC Berkeley.
- Deep Learning:
  - *MIT 6.S191 Introduction to Deep Learning* - Massachusetts Institute of Technology,
  - *Natural Language Processing with Deep Learning* - CS224n at Stanford.

Uczenie ze wzmocnieniem

Wykłady

- *Hado Van Hasselt, Advanced Deep Learning & Reinforcement Learning*.
- *RL Course by David Silver*.
- *Deep Learning State of the Art* - MIT Deep Learning Series.

Artykuły

Artykuły

- AlphaTensor
- Hanabi
- Zabawa w chowanego
- AlphaFold
- Dota2
- InstructGPT
- Elektronika
- EfficientZero