Katarzyna Kanicka

WSI Sprawozdanie 6 – Uczenie ze wzmocnieniem

Treść polecenia:

1. Zaimplementować algorytm Q-learning, a następnie użyć go do wytrenowania agenta rozwiązującego problem Cliff Walking https://gymnasium.farama.org/environments/toy_text/cliff_walking/
2. Stworzyć wizualizację wyuczonej polityki i umieścić ją w sprawozdaniu. Wzór wizualizacji https://gymnasium.farama.org/tutorials/training_agents/FrozenLake_tuto/#visualization

Instrukcja do używania skryptu

Po pobraniu repozytorium należy:

- przejść do głownego katalogu – lab3

```
$ cd lab3
```

- stworzyć środowisko wirtualne i pobrać requirements.txt

```
$ python3 -m venv .venv
$ source .venv/bin/activate
$ pip install -r requirements.txt
```

- uruchomić plik main.py, w terminalu za pomocą:

```
$ python3 main.py [argumenty]
```

Możliwe argumenty:

- **learning_rate** – krok uczący (domyślnie 0.9)
- **discount_factor** – współczynnik gamma (domyślnie 0.9)
- **episodes** – liczba epizodów (domyślnie 1000)
- **training** - flaga, która decyduje o tym, czy obecne uruchomienie ma być na nowo trenowane (w przeciwnym wypadku załadowuje Qtable z pliku CliffWalking_model.pkl)
- **render** – flaga, która decyduje czy tworzony env ma ustawić render = ‘human’, czyli ma wyświetlać graficznie symulacje epizodów

Środowisko ‘Cliff Walking’

Opis

- Gdy gra rozpoczyna się, gracz znajduje się na pozycji [3, 0] na planszy 4x12, a cel znajduje się w miejscu [3, 11]. Jeśli gracz osiągnie cel, epizod się kończy.
- Na pozycjach [3, 1..10] znajduje się klif. Jeśli gracz znajdzie się w lokalizacji klifu, powróci do lokalizacji początkowej.

Akcje

- 0 – ruch do góry
- 1 – ruch w prawo
- 2 – ruch w dół
- 3 – ruch w lewo

Nagrody

- Każdy krok: -1
- Wpadnięcie w klif: -100
- Osiągnięcie celu: 0

Działanie algorytmu – Q-learning

Wartość Qtable(s, a) przedstawia wartość nagrody, jaką spodziewamy się dostać po podjęciu akcji *a* w stanie *s*. Jest ona aktualizowana podczas uczenia na podstawie wzoru:

$$Q^{new}(S_t, A_t) \leftarrow (1 - \underbrace{\alpha}_{\text{learning rate}}) \cdot \underbrace{Q(S_t, A_t)}_{\text{current value}} + \underbrace{\alpha}_{\text{learning rate}} \cdot \underbrace{\left(\underbrace{R_{t+1}}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \cdot \underbrace{\max_a Q(S_{t+1}, a)}_{\text{estimate of optimal future value}} \right)}_{\text{new value (temporal difference target)}}$$

Została użyta strategia **epsilon-greedy**:

Aby zachować równowagę pomiędzy ulepszaniem dotychczasowych, a odkrywaniem nowych zachowań agenta (czyli równowagę pomiędzy uczeniem a eksploracją) akcje są początkowo wybierane losowo, a w miarę trwania uczenia zostaje wzmiejszany współczynnik losowych akcji (epsilon). Z czasem agent zacznie częściej wybierać akcje na podstawie Qtable.

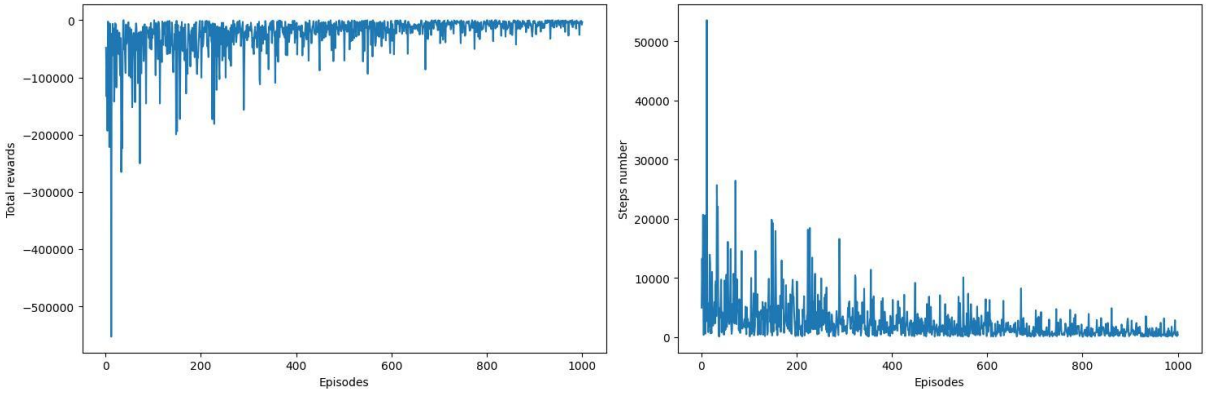
Na początku epsilon ma wartość 1, i z każdym epizodem zmienia się na max(epsilon – 0.0001, 0).

Wyniki programu

Proces uczenia

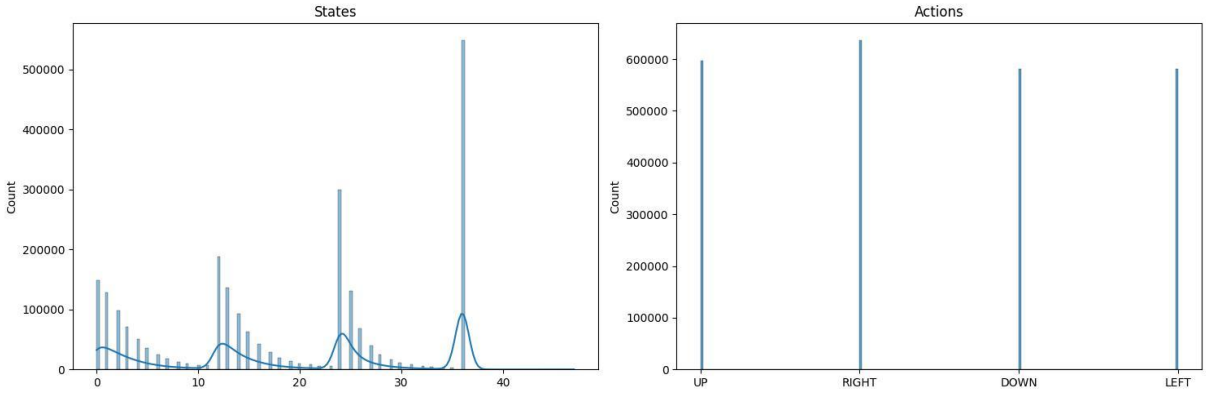
Liczba epizodów: 1000

Zmiana wielkości nagród i kroków na przestrzeni epizodów



- Na początku widać intensywną eksplorację z dużymi wahaniami w nagrodach i liczbie kroków.
- W miarę spadku epsilonu algorytm przechodzi w fazę eksploatacji.

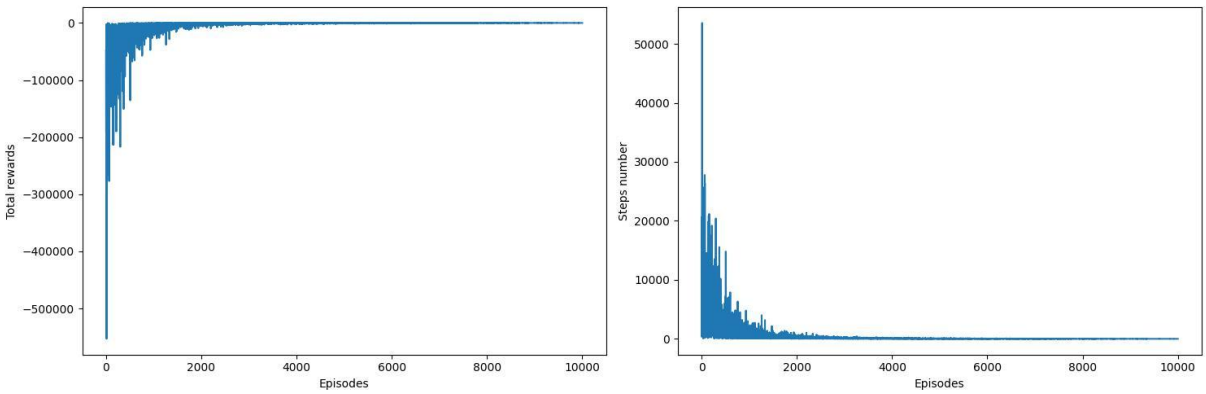
Rozkład stanów i akcji



- Najczęstsze stany znajdują się wzdłuż lewego brzegu planszy (w tym stan początkowy – 36).
- Najczęściej wybieraną akcją jest „RIGHT”, co zgadza się z lokalizacją gracza i celu.

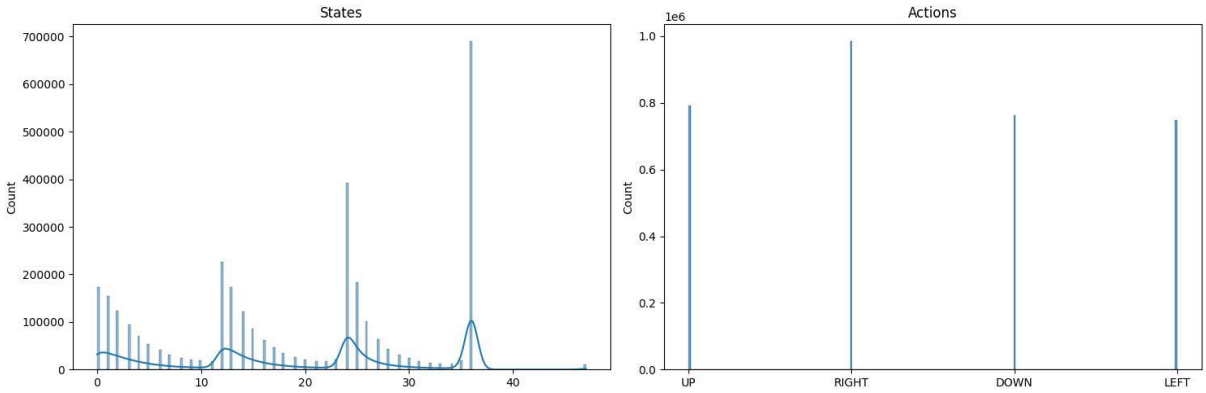
Liczba epizodów: 10000

Zmiana wielkości nagród i kroków na przestrzeni epizodów



- Przy większej liczbie epizodów nagrody i kroki zbliżają się do minimalnych wartości: nagrody równe -13, kroki równe 13 (zgodnie z tym, że każdy krok ma nagrodę -1, a minimalna liczba kroków to 13).

Rozkład stanów i akcji



- Większa liczba epizodów zwiększa dokładność decyzji. Wyraźniej widoczna jest przewaga wyboru akcji „RIGHT”.

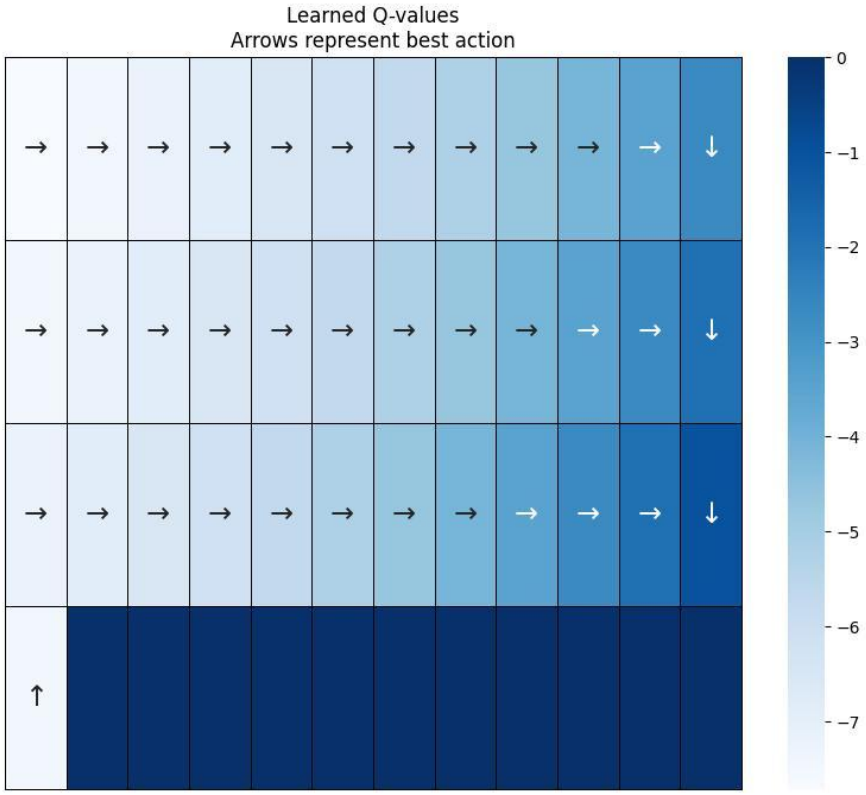
Qtable

	UP	RIGHT	DOWN	LEFT
0	-7.9411	-7.7123	-7.7123	-7.9411
1	-7.7123	-7.4581	-7.4581	-7.9411
2	-7.4581	-7.1757	-7.1757	-7.7123
3	-7.1757	-6.8619	-6.8619	-7.4581
4	-6.8619	-6.5132	-6.5132	-7.1757

5	-6.5132	-6.1258	-6.1258	-6.8619
6	-6.1258	-5.6953	-5.6953	-6.5132
7	-5.6953	-5.217	-5.217	-6.1258
8	-5.217	-4.6856	-4.6856	-5.6953
9	-4.6856	-4.0951	-4.0951	-5.217
10	-4.0951	-3.439	-3.439	-4.6856
11	-3.439	-3.439	-2.71	-4.0951
12	-7.9411	-7.4581	-7.4581	-7.7123
13	-7.7123	-7.1757	-7.1757	-7.7123
14	-7.4581	-6.8619	-6.8619	-7.4581
15	-7.1757	-6.5132	-6.5132	-7.1757
16	-6.8619	-6.1258	-6.1258	-6.8619
17	-6.5132	-5.6953	-5.6953	-6.5132
18	-6.1258	-5.217	-5.217	-6.1258
19	-5.6953	-4.6856	-4.6856	-5.6953
20	-5.217	-4.0951	-4.0951	-5.217
21	-4.6856	-3.439	-3.439	-4.6856
22	-4.0951	-2.71	-2.71	-4.0951
23	-3.439	-2.71	-1.9	-3.439
24	-7.7123	-7.1757	-7.7123	-7.4581
25	-7.4581	-6.8619	-106.7123	-7.4581
26	-7.1757	-6.5132	-106.7123	-7.1757
27	-6.8619	-6.1258	-106.7123	-6.8619
28	-6.5132	-5.6953	-106.7123	-6.5132
29	-6.1258	-5.217	-106.7123	-6.1258
30	-5.6953	-4.6856	-106.7123	-5.6953
31	-5.217	-4.0951	-106.7123	-5.217
32	-4.6856	-3.439	-106.7123	-4.6856
33	-4.0951	-2.71	-106.7123	-4.0951
34	-3.439	-1.9	-106.7123	-3.439
35	-2.71	-1.9	-1.0	-2.71
36	-7.4581	-106.7123	-7.7123	-7.7123
37	0.0	0.0	0.0	0.0
38	0.0	0.0	0.0	0.0
39	0.0	0.0	0.0	0.0
40	0.0	0.0	0.0	0.0
41	0.0	0.0	0.0	0.0
42	0.0	0.0	0.0	0.0
43	0.0	0.0	0.0	0.0
44	0.0	0.0	0.0	0.0
45	0.0	0.0	0.0	0.0
46	0.0	0.0	0.0	0.0
47	0.0	0.0	0.0	0.0

- Dla stanów obok klifu jedna z akcji ma wartość -106.7123, co eliminuje jej wybór przez agenta. Wartości dla stanów z klifu lub końcowych (37-47) wynoszą 0, ponieważ z tych stanów nie można kontynuować.

Polityka



- Polityka prowadzi do stanu końcowego (prawy dolny róg). Im bliżej celu, tym potencjalna nagroda jest wyższa, co przedstawia ciemniejszy kolor na wizualizacji.