WSTĘP DO SZTUCZNEJ INTELIGENCJI

Ćwiczenie 6 – Q Learning

MICHAŁ MIZIA 331407

SPIS TREŚCI

Wstęp

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

IMPLEMENTACJA ALGORYTMU

Jedyną decyzją projektową było użycie funkcji liniowej jako decay_epsilon.

```
    def decay_epsilon(self):
        self.epsilon = max(self.epsilon - self.epsilon_decay, 0.01)
```

Główna funkcja update jest zaimplementowana klasycznie

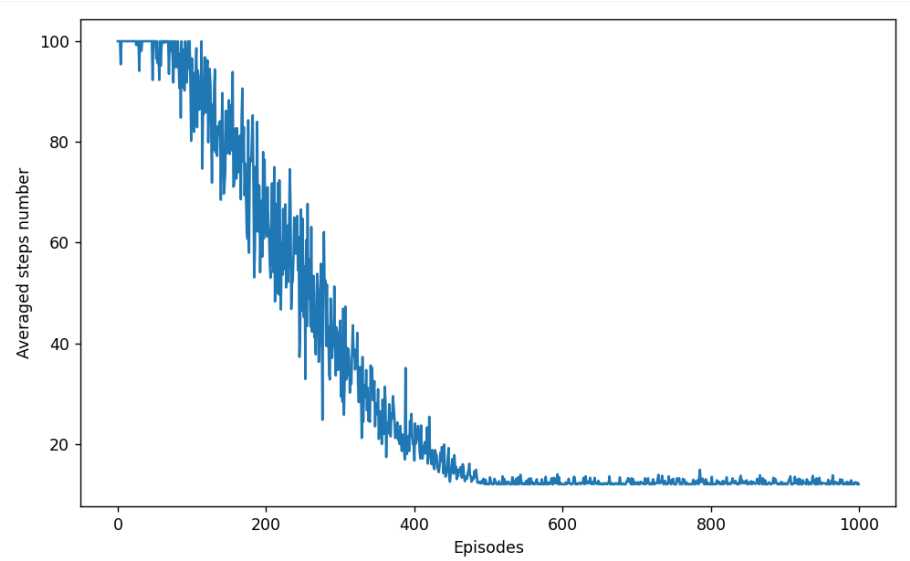
```
def update(self, state, new_state, action, reward) -> None:
    target = reward + self.discount_factor * np.max(self.q_values[new_state])
    error = target - self.q_values[state][action]
    self.training_errors.append(error)
    self.q_values[state][action] += self.lr * error
```

Wyniki

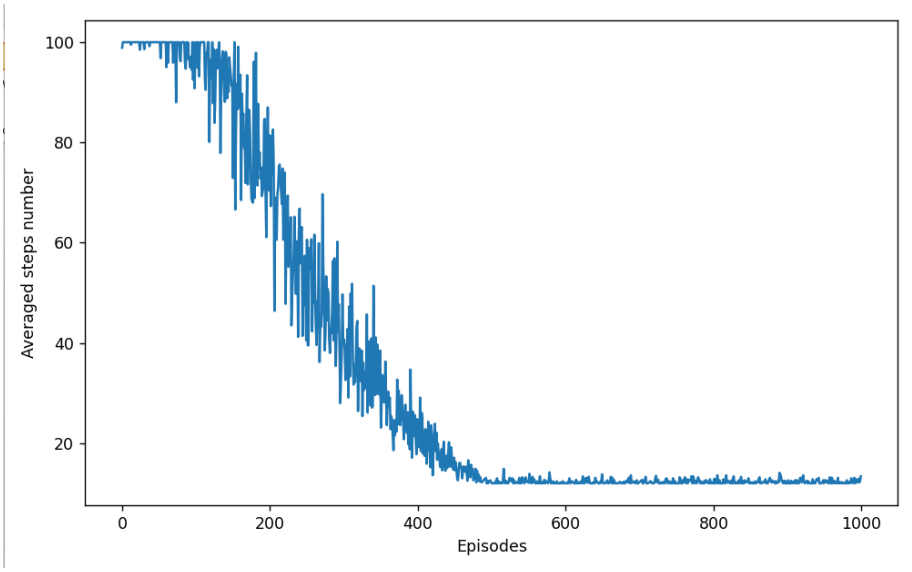
Przy parametrach:

```
agent = Agent(
    env,
    epsilon=1,
    epsilon_decay=(1 / (n_episodes / 2)),
    min_epsilon=0.1,
    learning_rate=0.5,
    discount_factor=0.9,
)
```

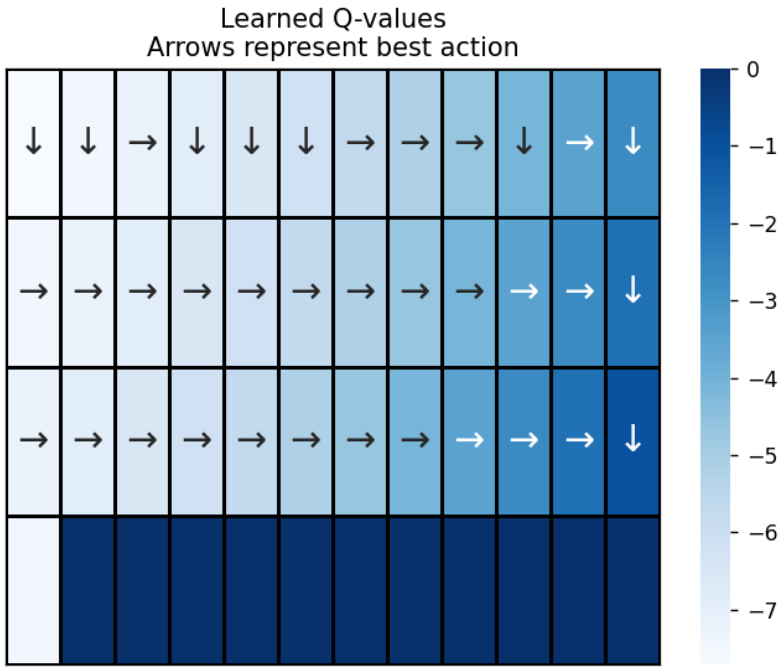
Udawało się znaleźć optymalne rozwiązanie po około 400 epizodach działania algorytmu.



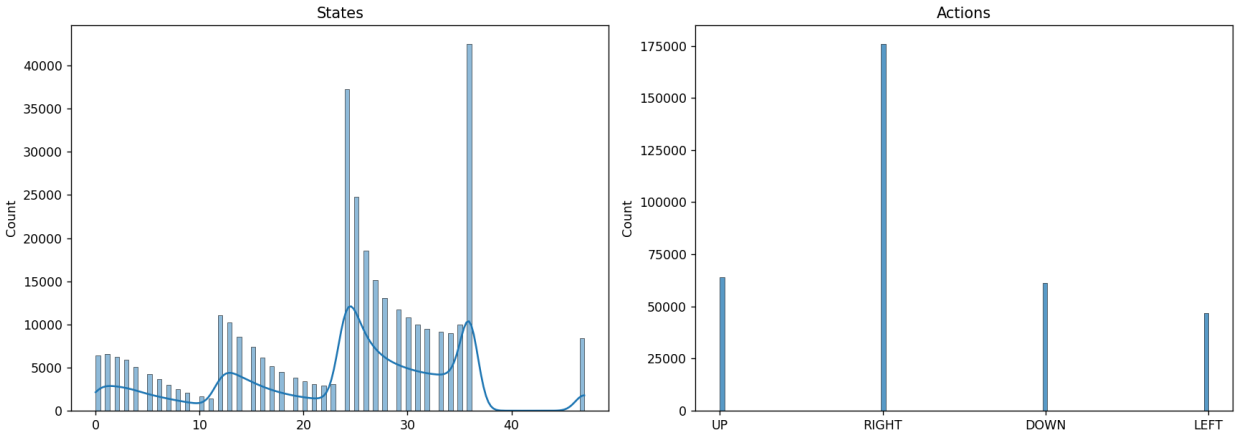
Zmiana learning_rate na 0.1 nie miała dużego wpływu na czas po którym algorytm doszedł do optymalnego rozwiązania:



Wyuczona polityka wygląda następująco:



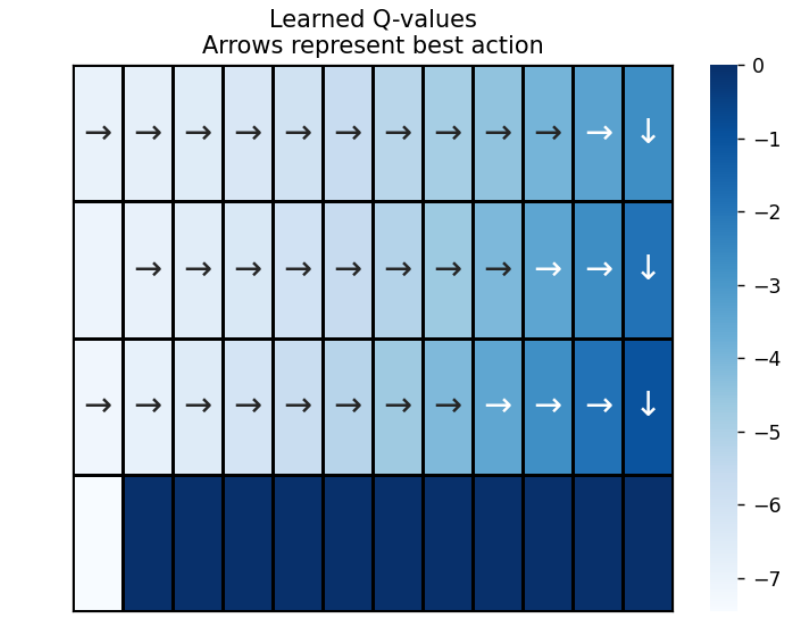
A dystrybucja wszystkich akcji i stanów



Widzimy że najczęściej wybierany jest ruch w prawo, najrzadsze stany w których znajduję się algorytm to stany „klifu” czyli tam gdzie nagroda początkowa wynosi -100, najczęstszy jest stan początkowy czyli 35 oraz stan w górę od niego czyli 23.

Wnioski

Przez to jak łatwy do rozwiązania jest problem Cliff Walkingu, learning_rate nie ma dużego wpływu na ilość epizodów potrzebnych do wytrenowania algorytmu jednak przy niższym learning_rate, w wizualizacji kierunków niektóre rzadko odwiedzane pola nie mają pewności na temat poprawnego kierunku:



Q learning to algorytm dobry do łatwych zadań gdzie łatwo możemy wymyślić politykę uczącą oraz znaleźć deterministyczne rozwiązanie, jednak nie uczy modelu ogólnego rozwiązania a jedynie rozwiązania konkretnego problemu.