WSI – Raport z zadania 6

Michał Pędziwiatr

Treść polecenia

Zaimplementować algorytm Q-learning, a następnie użyć go do wytrenowania agenta rozwiązującego problem Cliff Walking: https://gymnasium.farama.org/environments/toy_text/cliff_walking/

Stworzyć wizualizację wyuczonej polityki i umieścić ją w sprawozdaniu. Wzór wizualizacji: https://gymnasium.farama.org/tutorials/training_agents/FrozenLake_tuto/#visualization

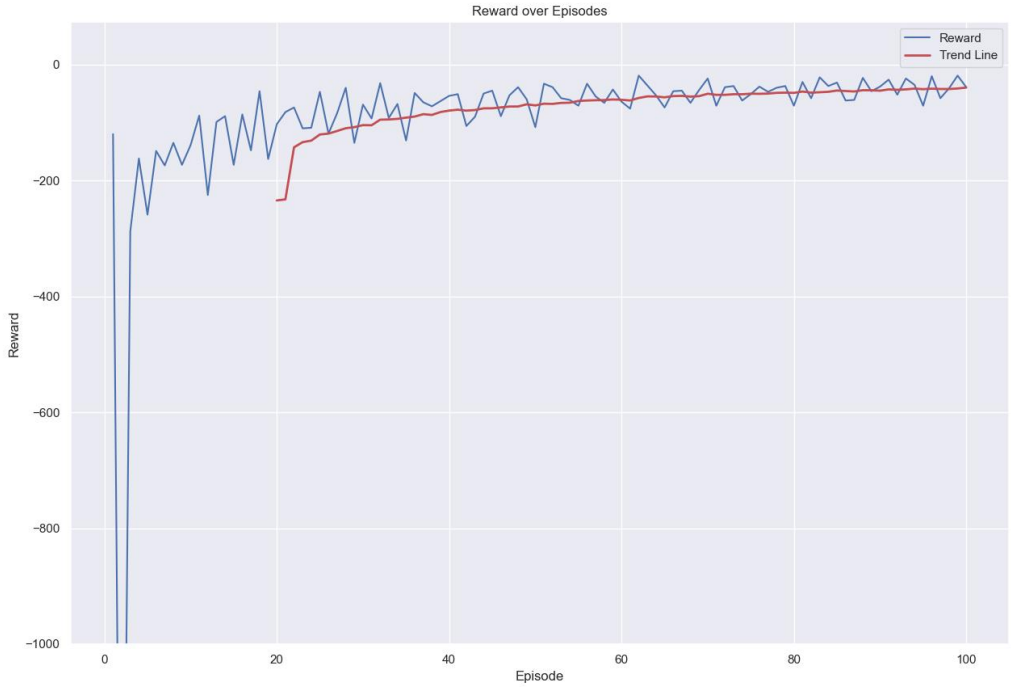
Nie należy implementować środowiska samemu - tworzymy je wywołaniem gymnasium.make("CliffWalking-v0").

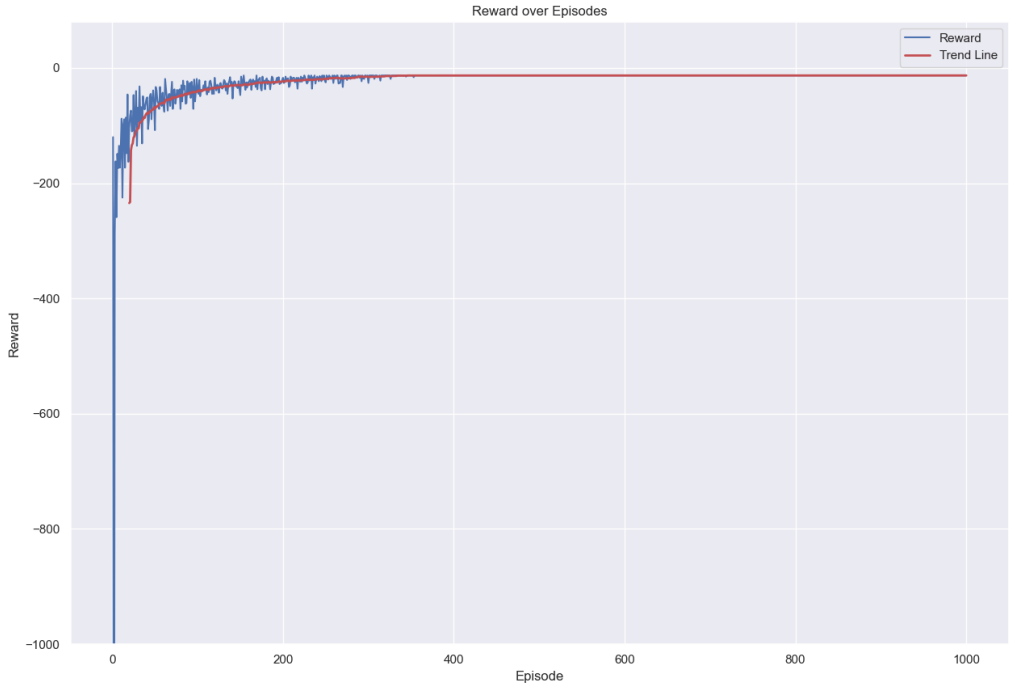
Argumenty wywołania skryptu

- episodes <int> - liczba epizodów, czyli podejść uczących algorytm q-learning.
- max_steps <int> - maksymalna liczba kroków jakie algorytm może wykonać w pojedynczym epizodzie.
- epsilon <float> – współczynnik eksploracji, czyli szansa na wybranie przez algorytm nowego, nieeksplorowanego dotychczas ruchu, zamiast eksploatacji już poznanych akcji.
- beta <float> – współczynnik uczenia algorytmu, określający jak duży wpływ na wartość Q mieć będzie różnica (delta) nagrody uzyskanej od nagrody przewidywanej.
- gamma <float> – współczynnik dyskontowania, narzucająca zaufanie oraz wagę jaką przykładamy do naszych przewidywań, a także i preferencje algorytmu pomiędzy nagrodą krótkoterminową oraz długoterminową.

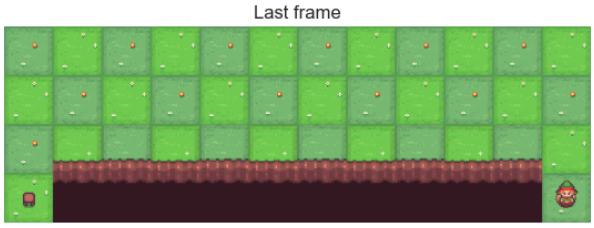
Rezultaty działania algorytmu:

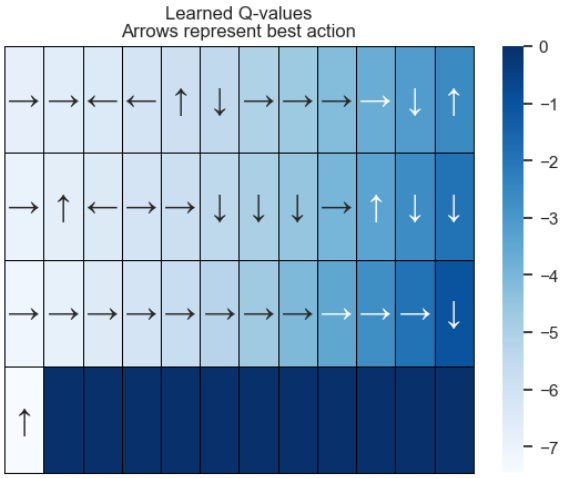
Wykres nagrody osiąganej przez algorytm w danym epizodzie:



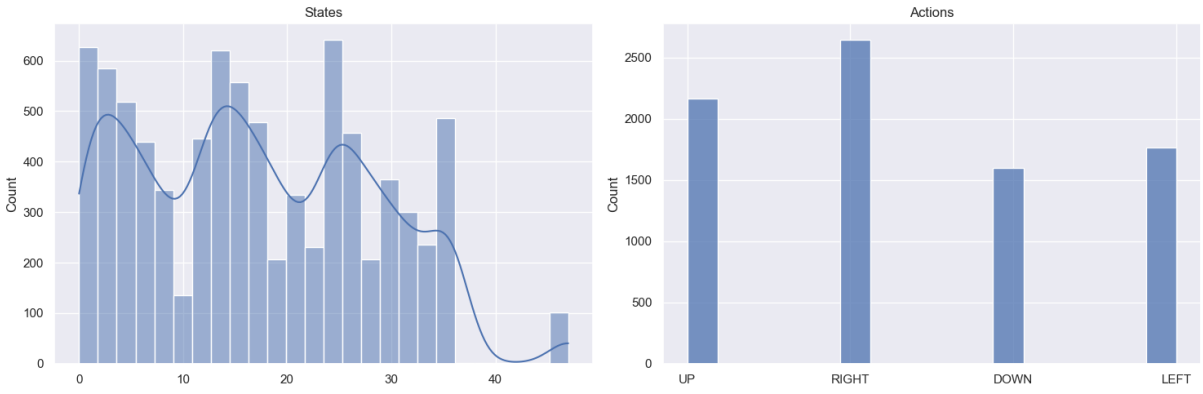


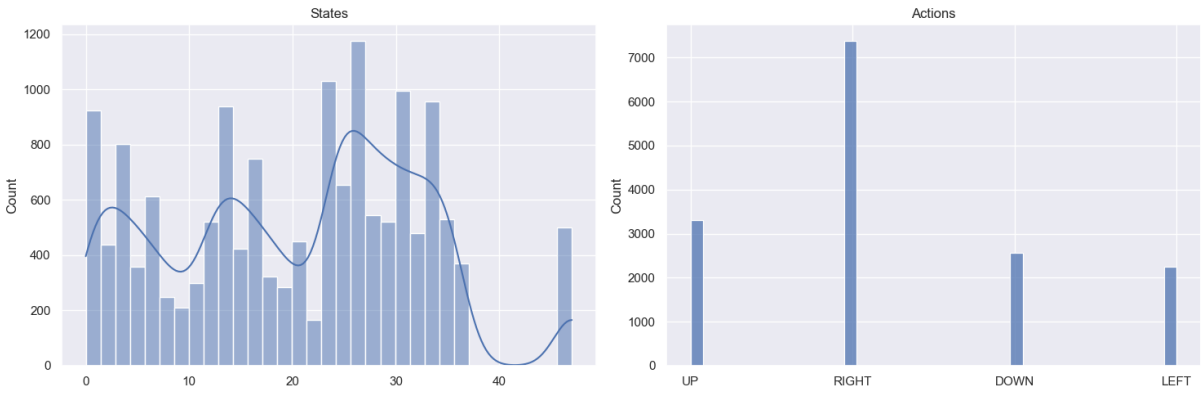
Końcowa pozycja agenta oraz mapa najlepszych poznanych ruchów dla każdej odkrytej pozycji:





Wykres częstotliwości odwiedzania danych stanów oraz wykonywania określonych ruchów dla odpowiednio 100 i 500 epizodów:



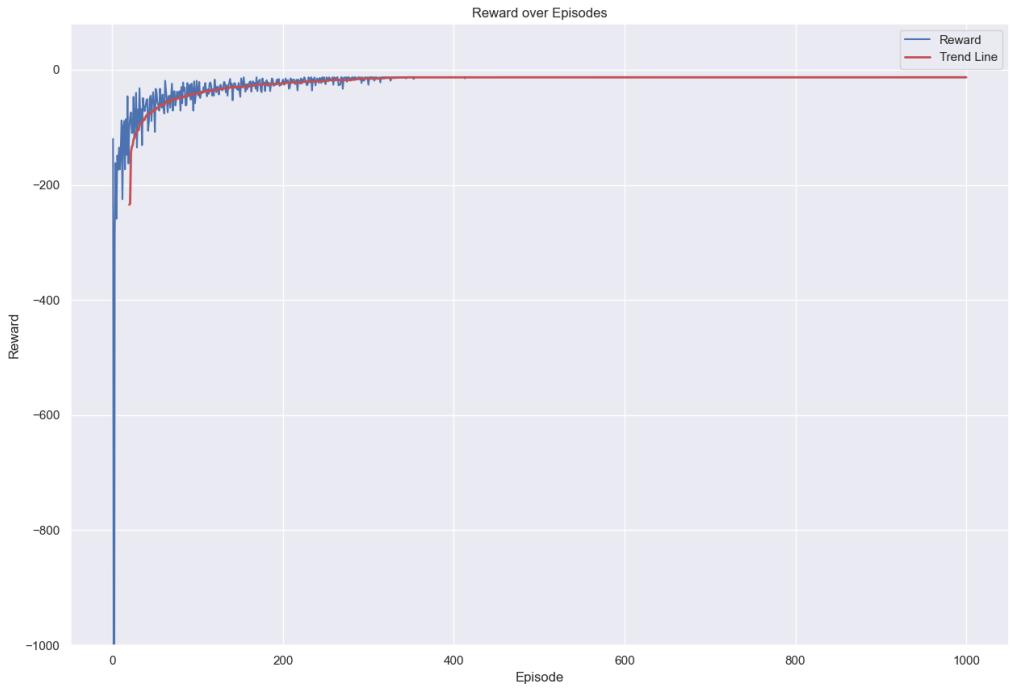


Eksperymenty

Eksperyment 1 - Wpływ współczynnika eksploracji (epsilonu) na optymalizację algorytmu

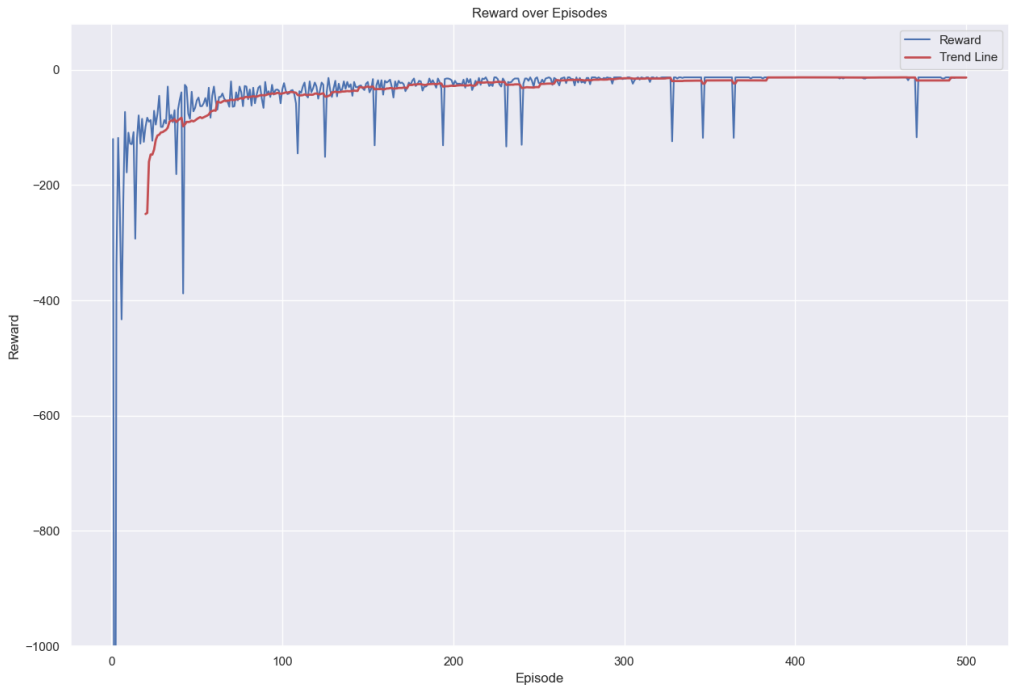
Współczynnik eksploracji: 0.0001

```
>======================< FINISHED >======================<
> Params:
    Episodes: 1000
    Max steps per episode: 1000
    Exploration rate: 0.001
    Learning rate: 0.1
    Discount factor: 0.9
>-------------------------------------------------------<
> Results:
    Average reward: -23.407
    Average reward in last 100 episodes: -13.000
    Best reward: -13  (Best possible: -13)
>=======================================================<
```



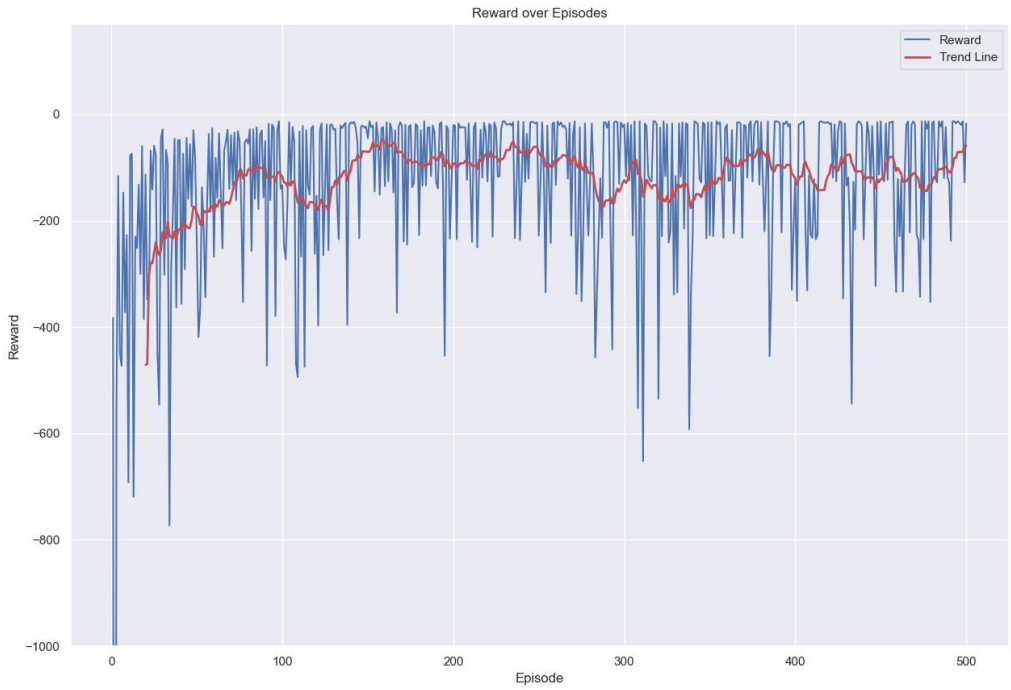
Współczynnik eksploracji: 0.01

```
>======================< FINISHED >======================<
> Params:
    Episodes: 500
    Max steps per episode: 1000
    Exploration rate: 0.01
    Learning rate: 0.1
    Discount factor: 0.9
>-------------------------------------------------------<
> Results:
    Average reward: -36.670
    Average reward in last 100 episodes: -14.240
    Best reward: -13  (Best possible: -13)
>=======================================================<
```



Współczynnik eksploracji: 0.25

```
>======================< FINISHED >======================<
> Params:
    Episodes: 500
    Max steps per episode: 1000
    Exploration rate: 0.2
    Learning rate: 0.1
    Discount factor: 0.9
>-------------------------------------------------------<
> Results:
    Average reward: -127.806
    Average reward in last 100 episodes: -103.790
    Best reward: -13  (Best possible: -13)
>=======================================================<
```



Eksperyment 1 – Wnioski

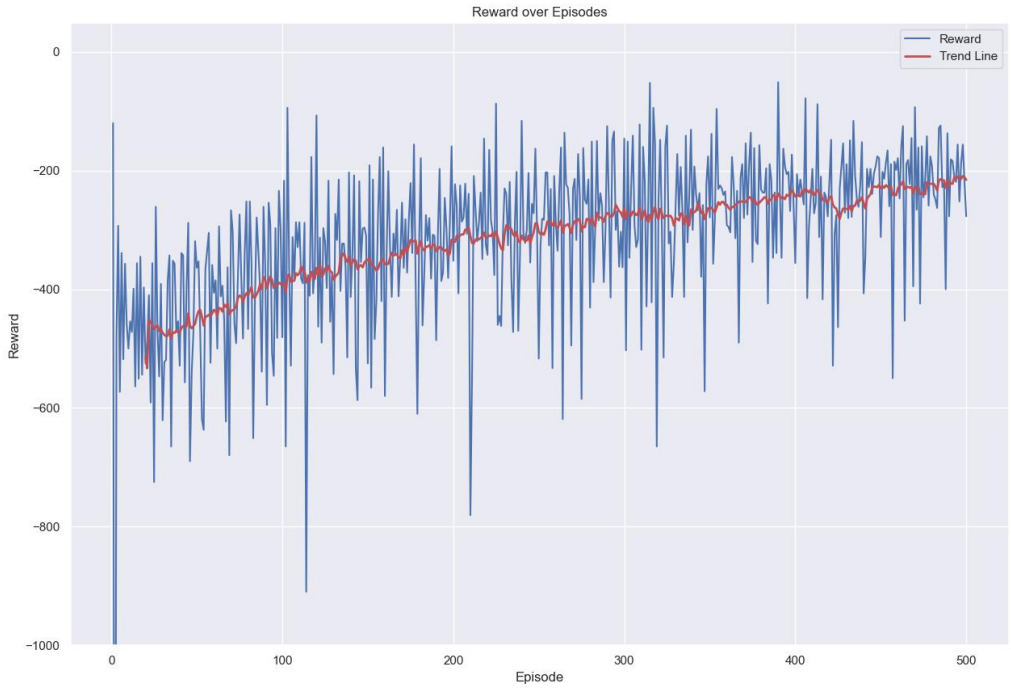
Jak pokazały przeprowadzone pomiary, dobranie odpowiedniego współczynnika eksploracji w stosunku do eksploatacji odkrytych akcji, jest kluczowe dla optymalnego działania algorytmu Q-learn. Widoczne na wykresach „skoki" wartości, mimo braku zmian w innych parametrach skryptu oraz wartości najlepszej osiągniętej nagrody jasno wskazuje na fakt, iż wzrost tego parametru bezpośrednio wpływa na nieprzewidywalność oraz losowość wyników. Jest to uzasadnione, ponieważ parametr ten z założenia, decyduje o częstotliwości ignorowania dotychczas zdobytych przez program danych i wykonywania ruchów losowych. Takie działanie, mimo chaotyczności w swej naturze jest kluczowe dla poprawnego działania algorytmu Q-learn, bowiem umożliwia wyprowadzenie algorytmu z potencjalnych „pułapek" (takich jak lokalne minimum) oraz zmusza go do eksploracji nowych, potencjalnie lepszych rozwiązań.

Warto jednak pamiętać, że w naszym przypadku, gdzie każdy ruch wiąże się z obniżeniem nagrody i nie osiąga ona nigdy wartości nieujemnych przy jednoczesnej inicjalizacji tablicy wartości Q jako macierz zerową, wpływ ten będzie dużo mniejszy. Dzieję się tak, ponieważ niezależnie od jakości danego ruchu, zawsze będzie on w oczach algorytmu gorszy od innego niezbadanego, dla którego bilans nagrody wynosi 0.

Eksperyment 2 - Wpływ współczynnika uczenia (bety) na optymalizację algorytmu

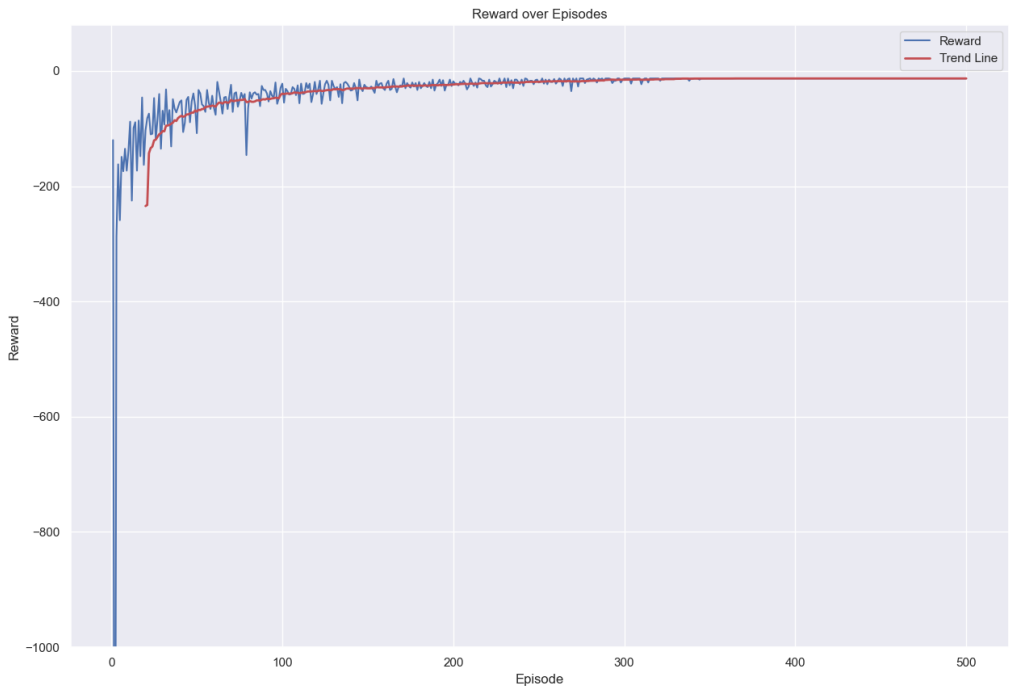
Współczynnik uczenia: 0.001

```
>======================< FINISHED >======================<
> Params:
    Episodes: 500
    Max steps per episode: 1000
    Exploration rate: 0.0001
    Learning rate: 0.001
    Discount factor: 0.9
>-------------------------------------------------------<
> Results:
    Average reward: -315.644
    Average reward in last 100 episodes: -233.660
    Best reward: -51  (Best possible: -13)
>=======================================================<
```



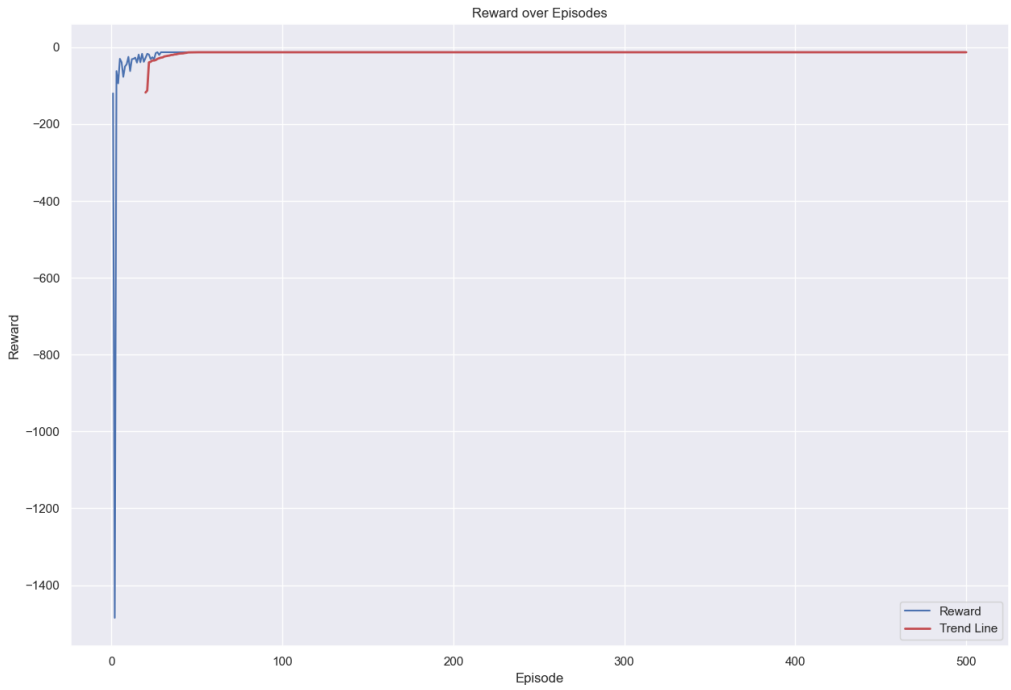
Współczynnik uczenia: 0.01

```
>======================< FINISHED >======================<
> Params:
    Episodes: 500
    Max steps per episode: 1000
    Exploration rate: 0.0001
    Learning rate: 0.1
    Discount factor: 0.9
>-------------------------------------------------------<
> Results:
    Average reward: -33.374
    Average reward in last 100 episodes: -13.000
    Best reward: -13  (Best possible: -13)
>=======================================================<
```



Współczynnik uczenia: 0.9

```
>======================< FINISHED >======================<
> Params:
    Episodes: 500
    Max steps per episode: 1000
    Exploration rate: 0.0001
    Learning rate: 0.9
    Discount factor: 0.9
>------------------------------------------------------<
> Results:
    Average reward: -17.330
    Average reward in last 100 episodes: -13.000
    Best reward: -13  (Best possible: -13)
>======================================================<
```



Eksperyment 2 – Wnioski

Wykonane pomiary, podobnie jak w ostatnim eksperymencie, bezsprzecznie ukazują nam jak wielką wagę ma dobór odpowiednich wartości badanego parametru w kontekście optymalizacji naszego algorytmu. Parametr Beta – współczynnik uczenia, w algorytmie Q-learning oznacza wpływ, jaki różnica pomiędzy oczekiwaną wartością, a wartością uzyskaną ma na aktualizacje wartości Q. Jest on także w pewnym sensie więc prędkością zmiany tychże wartości.

Zgodnie z przewidywaniami, niskie wartości bety, takie jak ukazane powyżej 0,001, doprowadziły do zbyt powolnej aktualizacji danych algorytmu. Mimo tego, iż prawdopodobnie dla bardziej zaawansowanych problemów, te rzędy wielkości mogłyby skutkować zwiększeniem dokładności oraz jakości rezultatów osiąganych przez algorytm, w naszym przypadku „Cliff walking" okazały się one nieoptymalne.

Przechodząc do wartości, dla których algorytm zwraca wartości zadowalające, takie jak 0,9 lub 0,01, widzimy kontynuacje trendu zwiększenia się ostrości wykresu nagrody wraz z rosnącą wartością bety. Świadczy to o wspomnianym już wcześniej tempie zmian wartości, które w naszym, dość prostym przypadku zdaje się sprawować najlepiej przy bardziej dynamicznym działaniu.

Mimo tego, iż pomiar dla wartości współczynnika uczenia równego 0,9 zdaje się górować nad tego dla wartości 0,01 w kontekście średniej nagrody, warto pamiętać, że to, który okaże się optymalniejszy zależy także od reszty parametrów i celu. Ostatecznie, oba parametry dochodzą do tych samych wartości nagrody (-13), różni się jedynie szybkość w jakim tego dokonują. Można więc stwierdzić, że w naszym przypadku zwiększenie wartości bety, będzie korzystne w przypadku niskiej liczby epizodów, takiej jak np. 100 lub 200, gdzie potrzebujemy dojść do zadowalających wartości jak najszybciej. Jeśli jednak w przyszłości chcielibyśmy rozszerzyć nasze środowisko i sprawić, że byłoby ono bardziej skomplikowane, prawdopodobnie niższe jej wartości w połączeniu z wysoką liczbą epizodów okazałyby się bardziej dokładne.