WSI: Laboratorium

Ćwiczenie: Uczenie ze wzmocnieniem

Paweł Skierś, 310895

Semestr zimowy 2021/22

ZADANIE

Celem ćwiczenia jest implementacja algorytmu Q-learning. Następnie należy stworzyć agenta rozwiązującego problem FrozenLake.

DOKUMENTACJA ROZWIĄZANIA

Rozwiązanie składa się z dwóch plików: main.py i Qlearning.py. Funkcja implementująca algorytm Q-learning znajduje się w pliku Qlearning.py. Przyjmuje ona następujące parametry:

- env – środowisko w który uczyć będzie się agent
- learning_rate – współczynnik uczenia się
- episodes – liczba epizodów, przez które będzie się agent
- discount_rate – dyskonto
- explotation_rate – współczynnik eksploracji, prawdopodobieństwo wybrania przypadkowej akcji
- max_steps – maksymalna liczba kroków, jaką może wykonać algorytm podczas jednego epizodu
- exploration_decay – współczynnik z jakim zmniejsza się współczynnik eksploracji
- max_exploration – maksymalna wartość współczynnika eksploracji
- min_exploration – minimalna wartość współczynnika eksploracji

Funkcja zwraca wyznaczone wartości funkcji Q agenta oraz historię uzyskanych nagród w każdym epizodzie przez agenta.

EKSPERYMENTY, WYNIKI I WNIOSKI

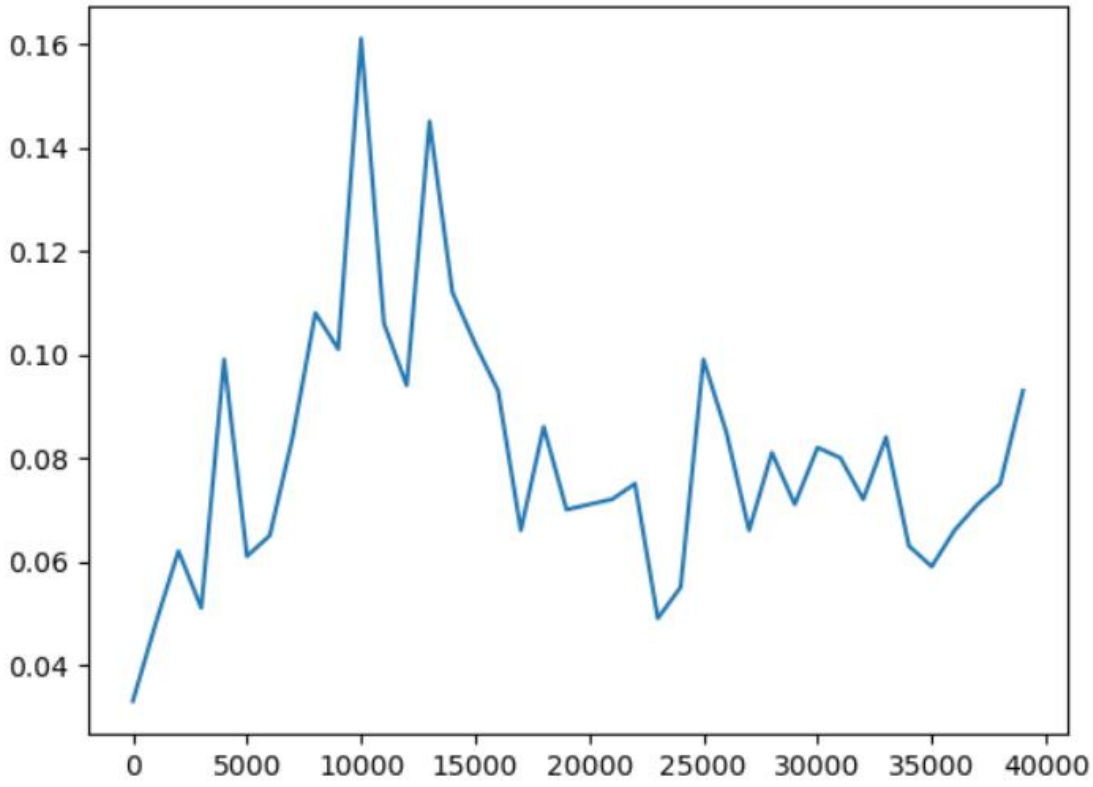
W ramach eksperymentów na stworzonym algorytmie, przeprowadzono eksperymenty, mające na celu sprawdzenie wpływu zadanego algorytmowi dyskonta i współczynnika uczenia. Poniżej widoczne są wyniki i wyciągnięte z nich wnioski.

Wpływ dyskonta

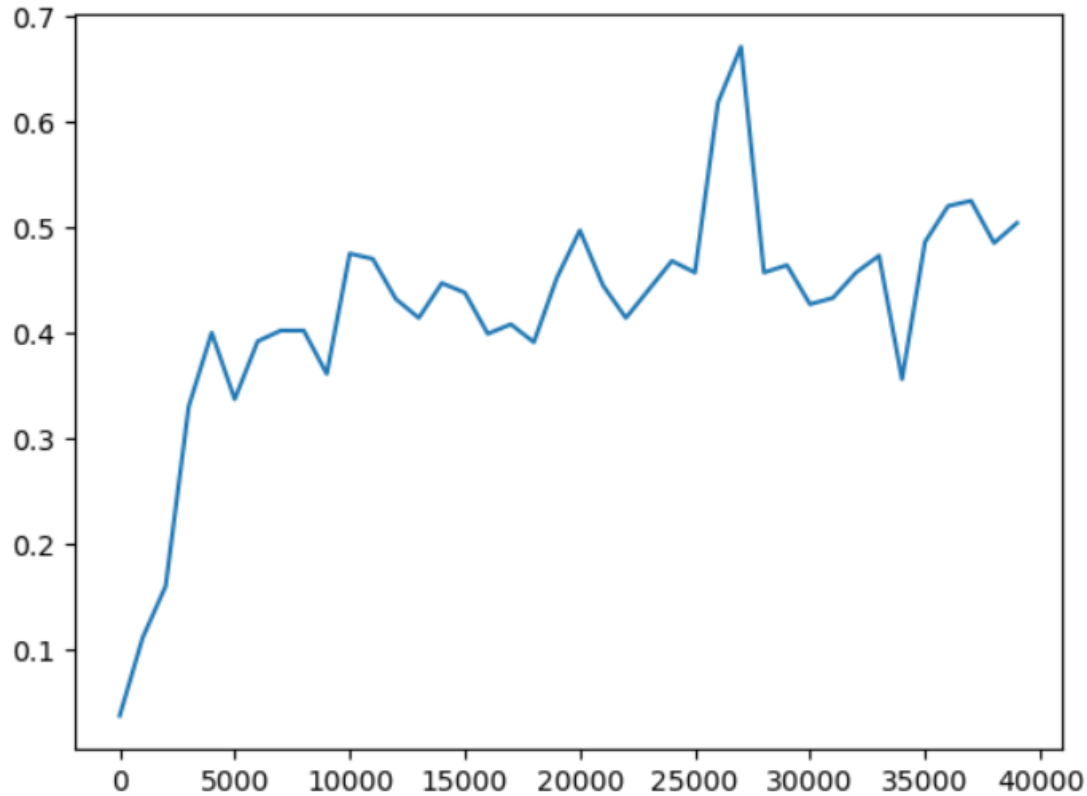
Parametry stałe dla wszystkich prób

```
episodes = 40000
max_steps = 100
learning_rate = 0.1
exploration_rate = 1
exploration_decay = 0.001
max_exploration = 1
min_exploration = 0.0001
```

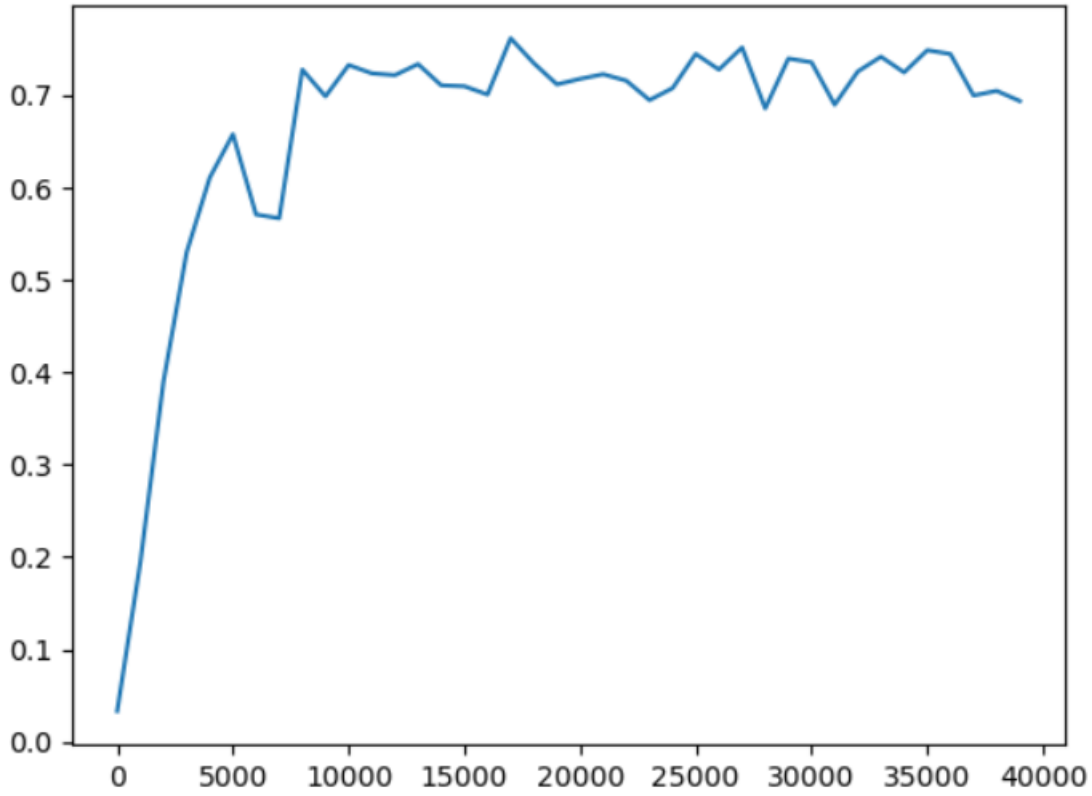
Dyskonto = 0.4



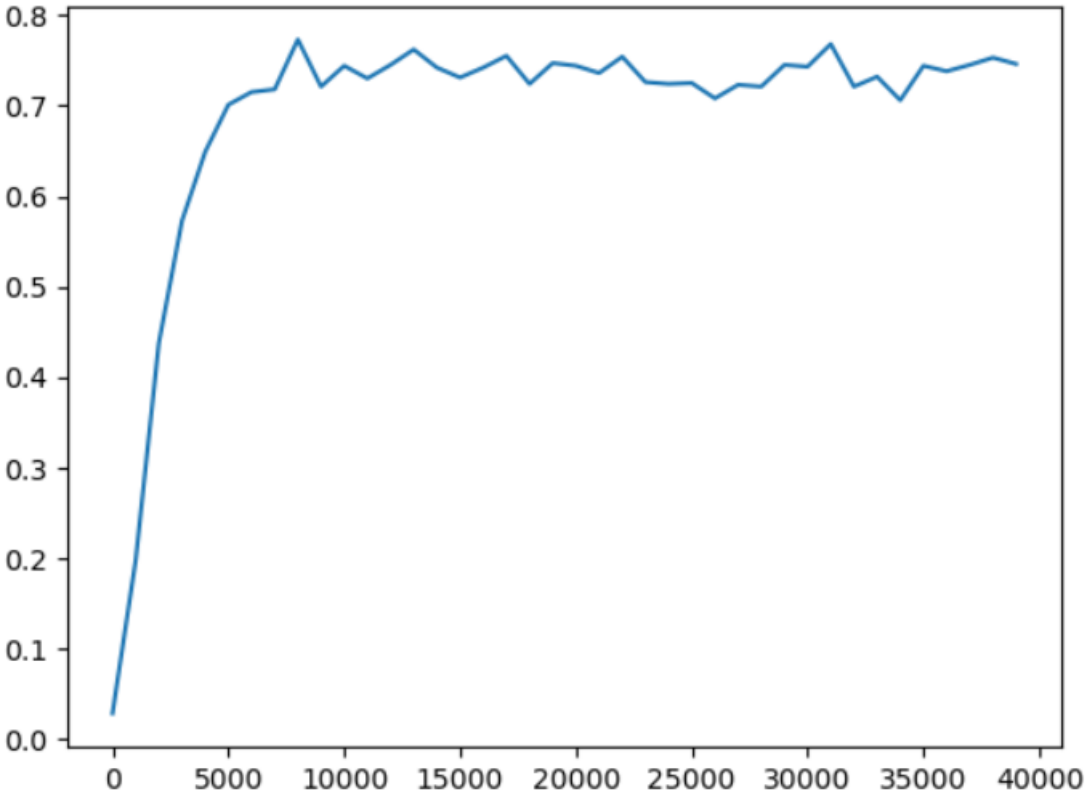
Dyskonto = 0.8



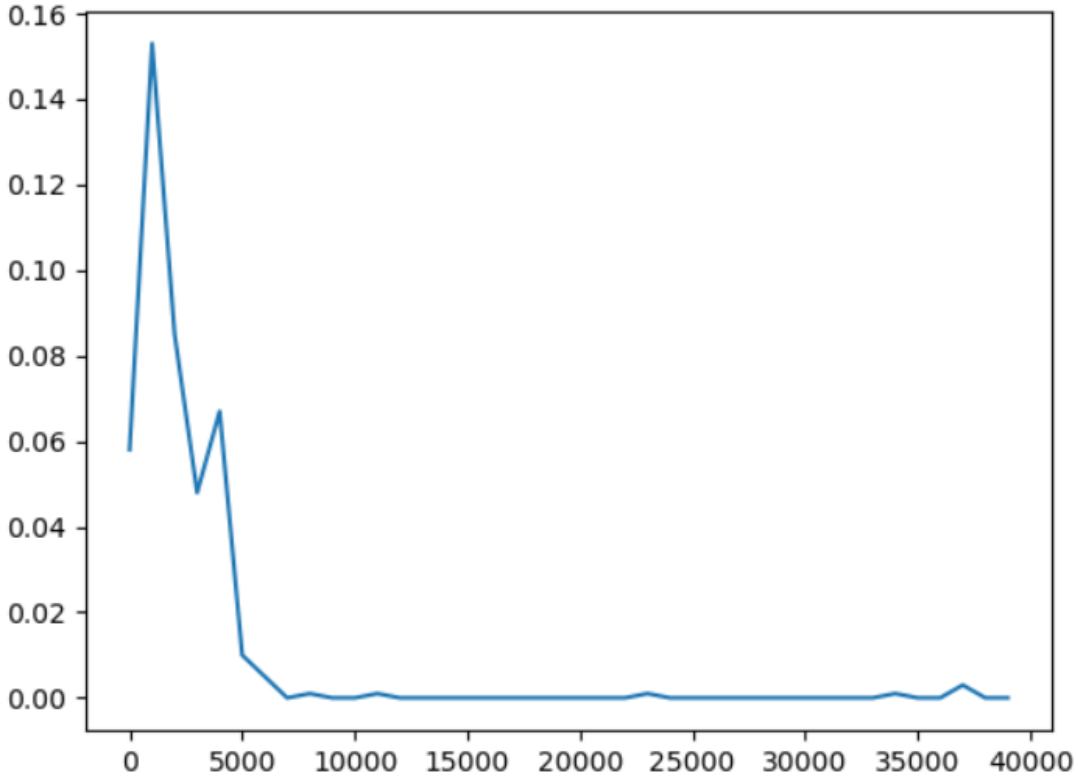
Dyskonto = 0.95



Dyskonto = 0.99



Dyskonto = 0.9999999



Wnioski

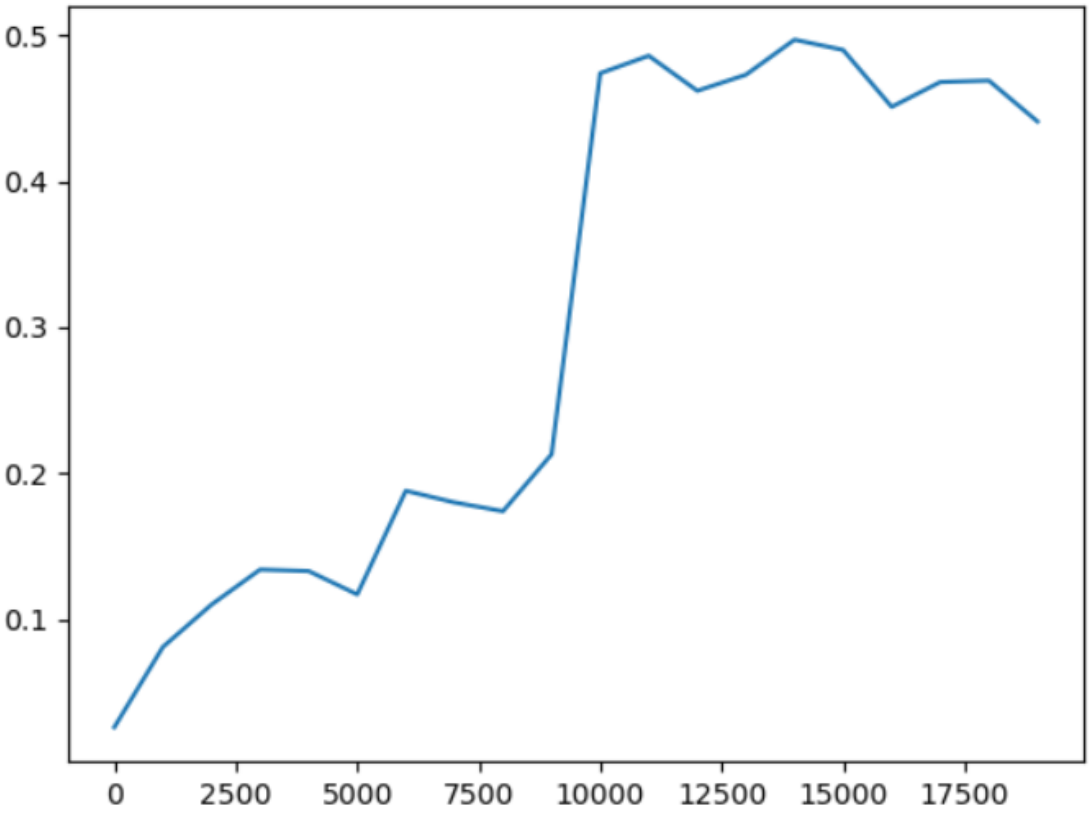
Dyskonto odpowiada za to jak dalekowzroczny jest algorytm. Im bliższe jedność jest dyskonto, tym ważniejsze są dla algorytmu przyszłe nagrody. Jednakże, większe dyskonto utrudnia również wyznaczenie dobrej polityki. Jak widzimy z wyników małe dyskonta sprawiają, że agent nie radzi sobie dobrze. Agent jest zapewne dobry w dochodzenie do celu będąc blisko tego celu, ale raczej słaby w początkowych fazach wędrówki. Zwiększanie dyskonta poprawia wyniki agenta aż do momentu osiągnięcia wartości dyskonta bardzo bliskich 1. Wtedy bowiem znalezienie dobrej polityki staje się za trudne dla agenta. W rezultacie agent ma beznadziejne wyniki.

Wpływ współczynnika uczenia

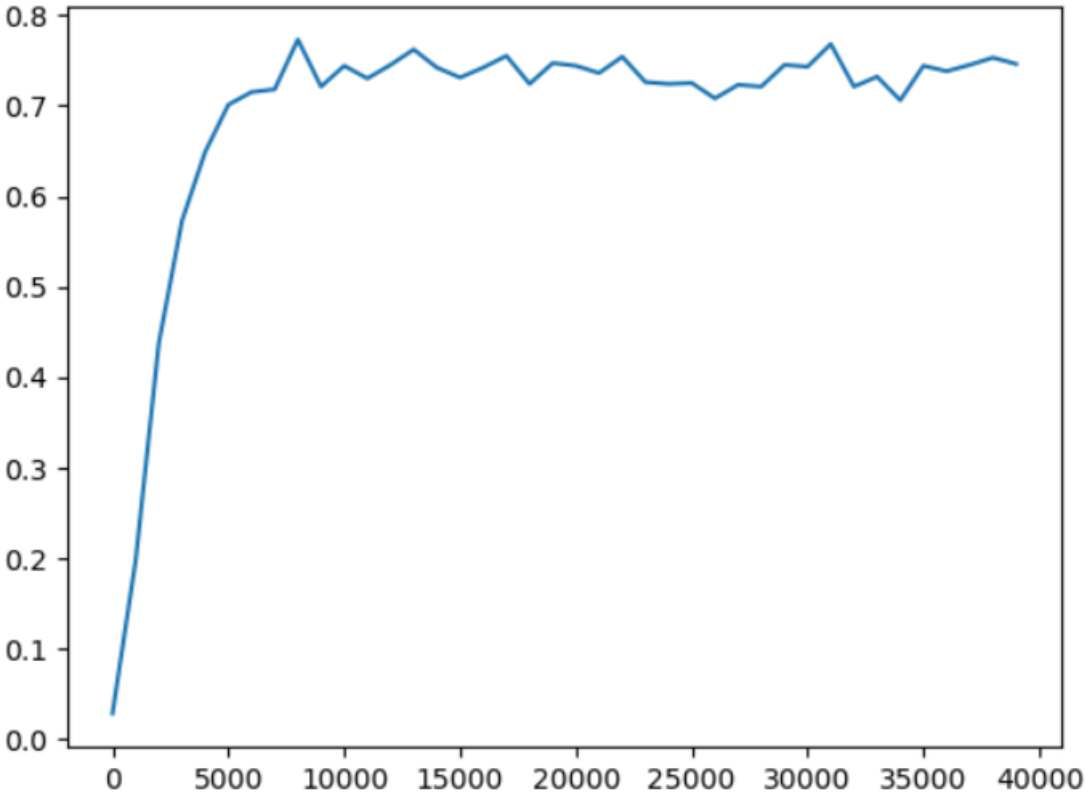
Parametry stałe dla wszystkich prób

```
episodes = 20000
max_steps = 100
discount_rate = 0.99
exploration_rate = 1
exploration_decay = 0.001
max_exploration = 1
min_exploration = 0.0001
```

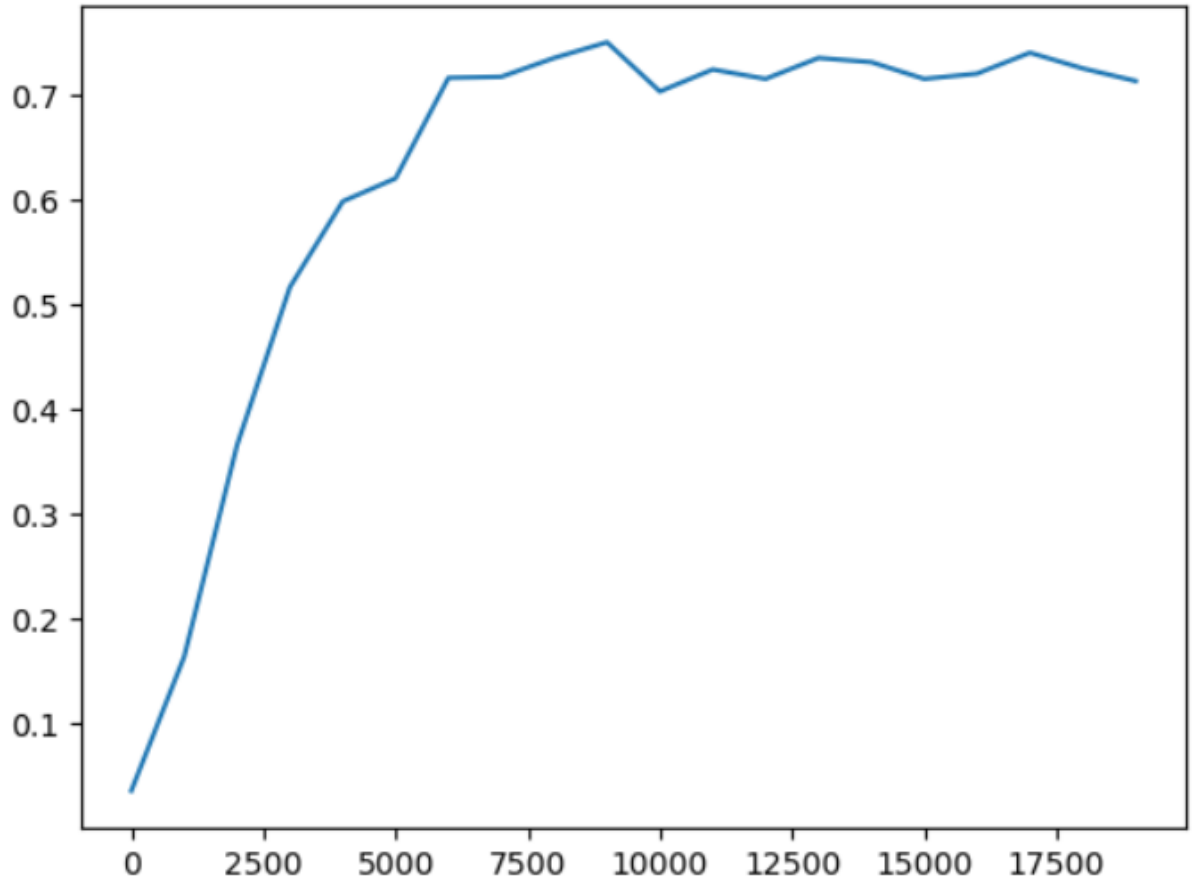
Współczynnik uczenia = 0.02



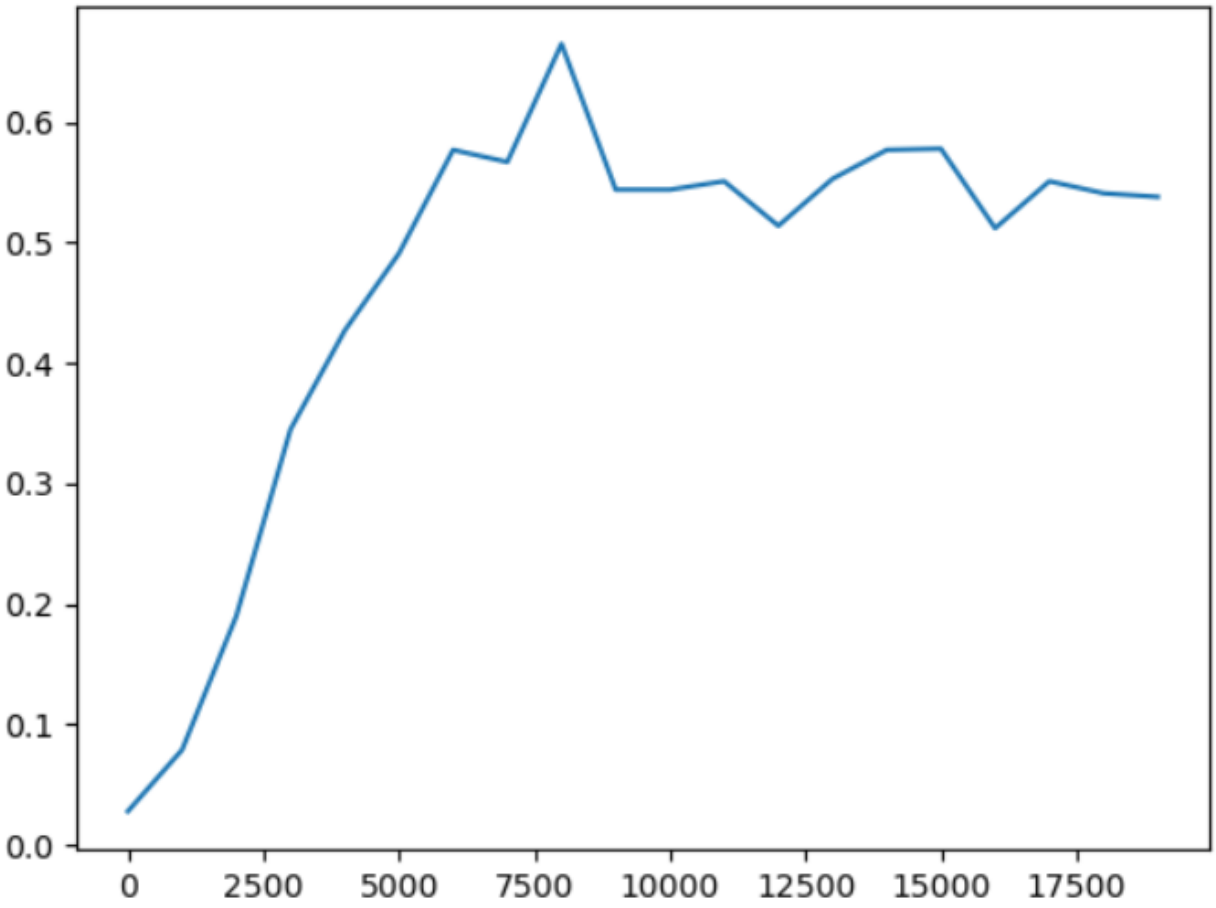
Współczynnik uczenia = 0.1



Współczynnik uczenia = 0.4



Współczynnik uczenia = 0.9



Wnioski

Najlepsze wyniki dają współczynniki o małej wartości tj. ok. 0.1. Dla współczynników znacząco mniejszych wyniki są widocznie gorsze. Dzieje się tak dlatego, że z powodu wielkości spółczynnika uczenia, agent nie jest w stanie zapamiętać/nauczyć się wystarczająco dużo w fazie, w której nastawiony jest na eksplorację. W rezultacie, kiedy algorytm zaczyna eksploatować to nie znajduje najlepszej możliwej do znalezienia polityki. Przy dużym współczynniku uczenia agent również nie osiąga najlepszych rezultatów. Dzieje się tak ponieważ, duży współczynnik uczenia sprawia, że wartości funkcji Q zmieniają się bardzo szybko i często nieco chaotycznie co utrudnia wyznaczenie optymalnej polityki.