WSI Ćwiczenie 6 – Q-learning

Kamil Marszałek 331401

Zadanie polega na nauczeniu agenta, przy pomocy algorytmu Q-learning, jak dojść do celu na planszy FrozenLake 8x8.

Założenia

Eksperymenty zostały przeprowadzone przy ustalonych wartościach hiperparametrów:

- Współczynnik dyskontowania – 0,95 (umożliwia uwzględnienie długoterminowych nagród w procesie uczenia)
- Współczynnik uczenia – 0,2 (pozwala na umiarkowaną aktualizację wartości Q, co ułatwia stopniowe dostosowywanie wartości Q)
- Liczba niezależnych uruchomień – 25
- Liczba epizodów – 1000 dla planszy z wyłączonym poślizgiem, 10000 dla wersji z poślizgiem
- Maksymalna liczba kroków w epizodzie – 200

Wybrane parametry najprawdopodobniej nie są optymalne, ale dają sensowne rezultaty, ponieważ pozwalają agentowi na skuteczne uczenie się w większości przypadków.

Na początku agent jest nastawiony w pełni na eksplorację (epsilon ustawione na 1), a co epizod epsilon jest zmniejszane (przemnażane przez 0.97). Pozwala to na stopniowe przechodzenie od eksploracji do eksploatacji polityki wyboru akcji.

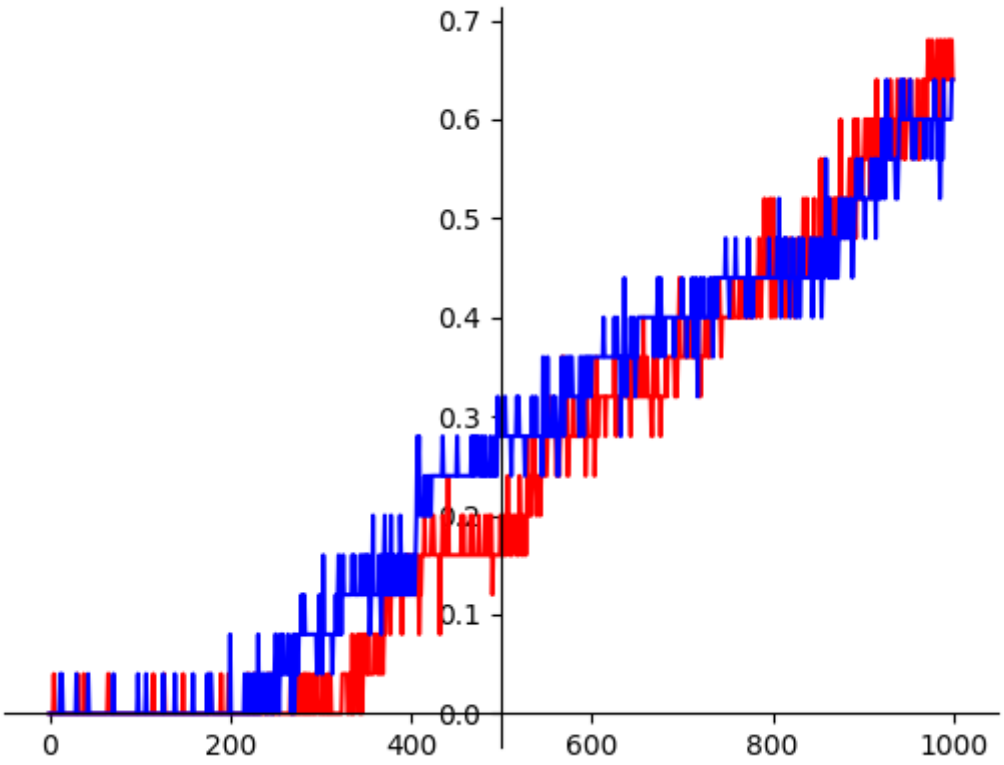
Systemy nagród:

- Standardowy – 1 po dojściu do skarbu, 0 w pozostałych przypadkach
- Karanie za wpadnięcie do dziury – 1 za dojście do skarbu, -1 za wpadnięcie do dziury, 0 w pozostałych przypadkach
- Karanie za wpadnięcie do dziury oraz za stanie w miejscu – 1 za dojście do skarbu, -1 za wpadnięcie do dziury lub zmarnowanie kroku, 0 w pozostałych przypadkach

Do uśrednienia wyników i sporządzenia wykresów został użyty standardowy system nagród, pochodzący ze środowisko gym – u mnie jest nazywany standardowym.

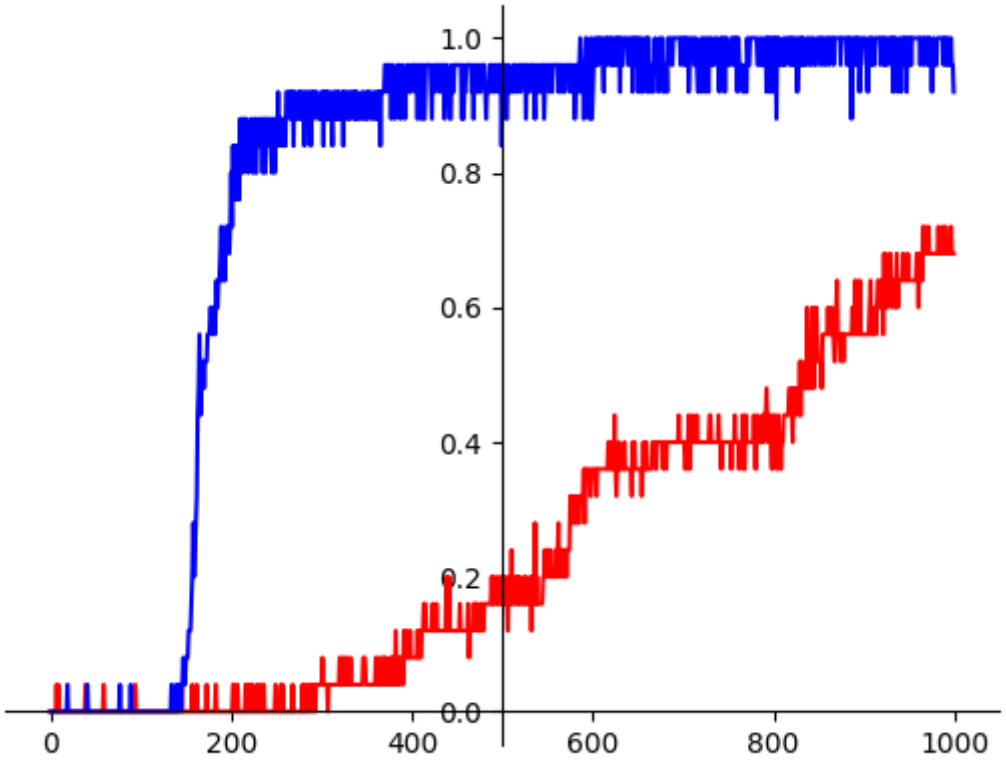
Wyniki eksperymentów

Na początku przedstawię wykres, gdzie zarówno czerwonym, jak i niebieskim został narysowany wykres wartości średnich nagród pochodzących ze środowiska gym, na początku wersja bez poślizgu:



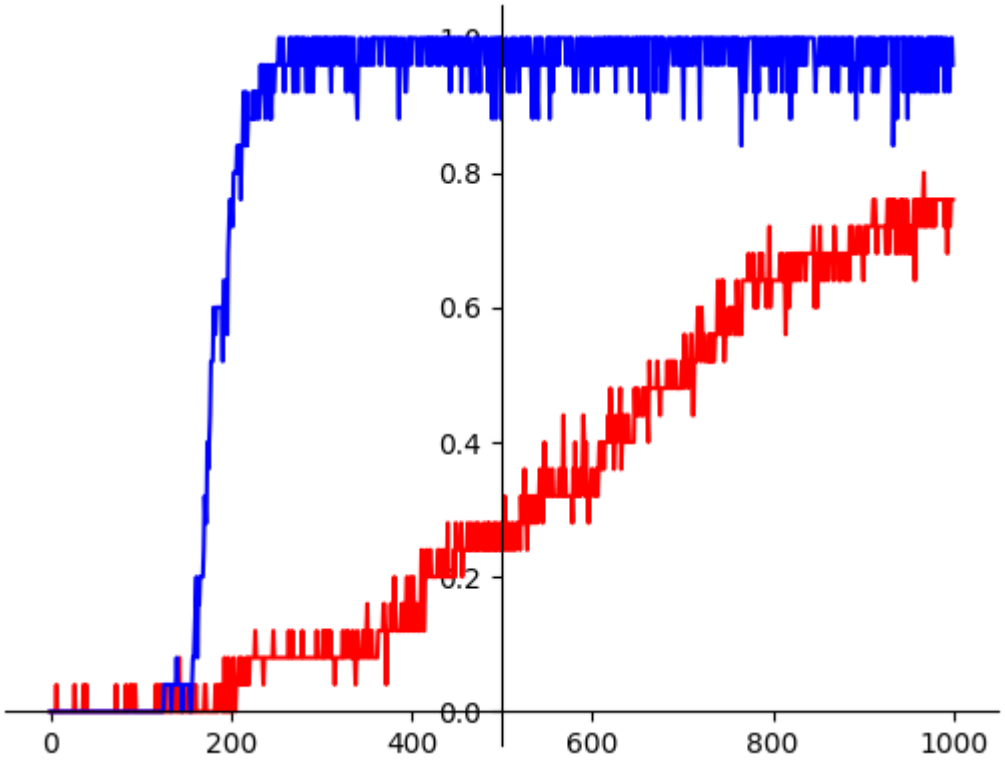
Z wykresu widzimy, że algorytm Q-learning jest sobie w stanie łatwo poradzić z problemem FrozenLake8x8, kiedy nie występuje poślizg. W czasie tysiąca epizodów jest w stanie nauczyć się na tyle, że nawet w 60-70% uruchomień, dla końcowych epizodów dociera do mety. Widać również, że algorytm jest silnie uzależniony od wylosowanych danych, gdyż dla tych samych parametrów uzyskujemy duże odchylenia. Na początku algorytm błądzi, ale w miarę wzrostu liczby epizodów rośnie średnia nagroda.

Następnie został sporządzony wykres porównawczy dla standardowego systemu nagród oraz systemu nagród, gdzie wpadnięcie dziury jest karane – przypadek bez poślizgu.



Linia niebieska obrazuje nam średnią nagrodę dla nowego systemu nagród, natomiast czerwona to standardowy system nagród. Dla przypadku bez poślizgu karanie za wpadnięcie do dziury znacząco poprawia skuteczność docierania do skarbu, agent bardzo szybko uczy się odpowiednich ruchów i już od około 200 epizodu osiąga rezultaty bliskie 1. Wraz z dalszym wzrostem epizodów średnia nagroda przybliża się powoli do wartości 1.

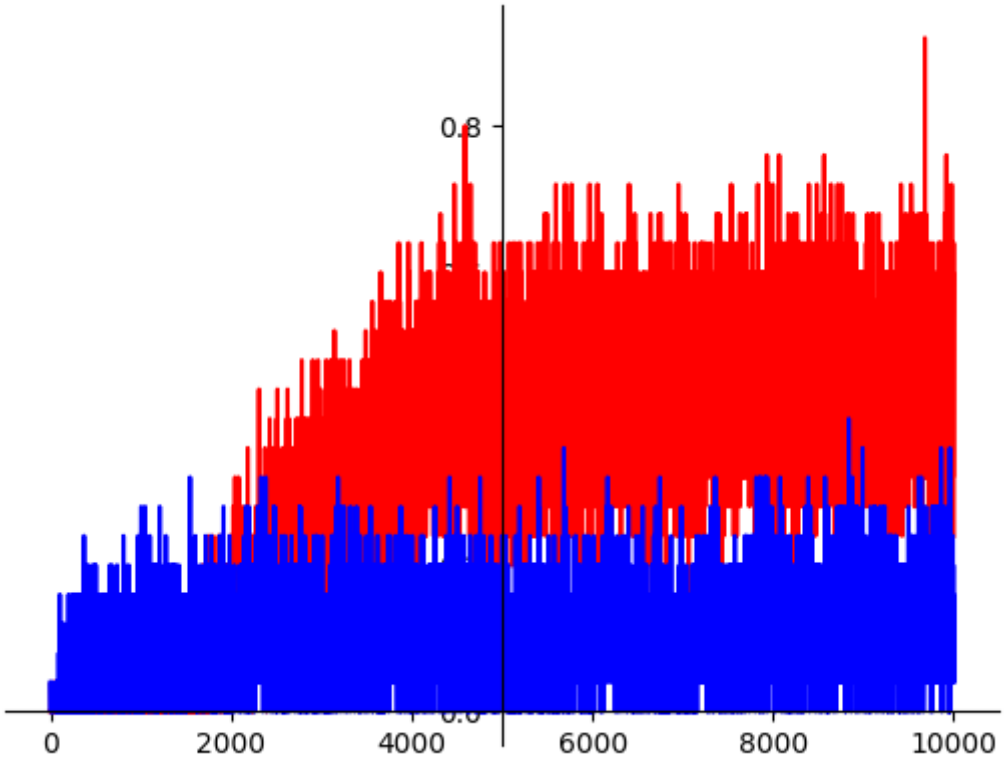
Kolejny wykres porównuje standardowy system nagród z systemem nagród, gdzie karane jest wpadanie w dziury oraz stanie w miejscu – środowisko bez poślizgu.



Linia niebieska oznacza system z karaniem za wpadanie w dziury oraz stanie w miejscu, natomiast czerwona to standardowy system nagród. Można zauważyć, że podobnie jak w poprzednim eksperymencie, począwszy od około 200 epizodu, średnia nagroda jest bliska wartości 1, zbieganie do 1 jest w tym przypadku silniejsze, co może wynikać z większej ilości informacji, które są podawane agentowi.

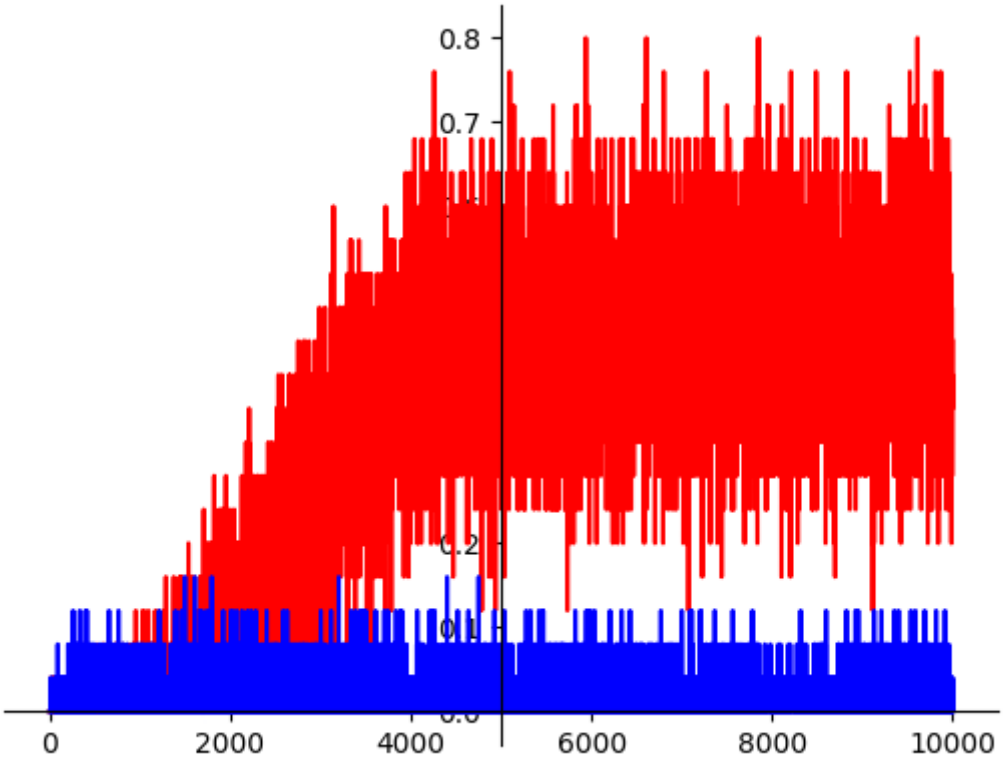
Następnie przeprowadzone zostały badania dla środowiska z włączonym poślizgiem, liczba epizodów została zwiększona do 10000.

Na początku wykres ukazujący porównanie standardowego systemu nagród z systemem z dodatkowym karaniem za wpadnięcie w dziurę.



Wykres dla danych z poślizgiem znacząco różni się od tego uzyskanego bez poślizgu. Dla wersji bez poślizgu 1000 epizodów w zupełności wystarczało do nauczenia agenta odpowiedniej strategii, dla środowiska z włączonym poślizgiem potrzebne jest znacznie więcej epizodów. Zdecydowanie lepiej radzi sobie standardowy sposób nagradzania. Już od około 4000 epizodów uzyskuje on dobrą średnią nagrodę. Agent, który uczy się poprzez zmodyfikowany system nagród nie jest w stanie uzyskać średniej nagrody wynoszącej powyżej 0,4 w żadnym z 10000 epizodów, natomiast dla standardowego systemu uzyskujemy około dwukrotnie lepsze rezultaty.

Następnie sporządzony został wykres przedstawiający porównanie standardowego systemu nagród z systemem, gdzie agent jest karany za wpadnięcie w dziurę oraz za stanie w miejscu.



Widzimy, że wyniki są jeszcze gorsze dla tego systemu nagród. W żadnym z 10000 epizodów nie uzyskaliśmy wyższej średniej nagrody niż 0,2. Prawdopodobnie nadmiar informacji prowadzi do nauczenia agenta błędnych praktyk. Może się to brać z faktu, że wykonanie wytypowanego ruchu jest niedeterministyczne (wytypowany ruch zostaje wykonany średnio co trzy akcje). Standardowy system nagród uzyskuje znacznie lepszą skuteczność niż ten najbardziej ubogacony.

Wnioski

Można stwierdzić, że algorytm Q-Learning radzi sobie dobrze z wyznaczaniem strategii postępowania dla agenta w środowisku FrozenLake.

W przypadku wersji bez poślizgu, agent uczy się stosunkowo szybko (1000 epizodów całkowicie wystarcza). Najlepiej sprawdził się najbardziej złożony system nagród z karaniem za stanie w miejscu oraz wpadnięcie do dziury, najgorzej natomiast najprostszy, standardowy system nagród. Deterministyczność środowiska sprawia, że nagrody są bardziej przewidywalne, co ułatwia proces uczenia. Wzbogacenie systemu nagród (karanie za stanie w miejscu i wpadnięcie do dziury) działa korzystnie, gdyż dostarcza agentowi więcej informacji o skutkach jego działań.

Natomiast dla środowiska z włączonym poślizgiem (niedeterministycznym) potrzebne jest zdecydowanie więcej epizodów zanim osiągniemy średnią wygraną porównywalną z tą, którą uzyskujemy dla środowiska bez poślizgu, aczkolwiek jedynie dla standardowego systemu nagród. Systemy nagród dające więcej informacji działają znacznie gorzej niż standardowy system nagród. Może to wynikać z tego, że agent dostaje informacje o akcji, która niekoniecznie musiała się wydarzyć, przez co wartości zawarte w qtable mogą zostać zaburzone. Agent dostaję informacje, które mogą go wprowadzać bardziej w błąd niż dawać sensowne wskazówki. Najlepiej radzi sobie standardowy system nagród (najprostszy), a najgorzej ten najbardziej złożony (karanie za stanie w miejscu i za wpadnięcie w dziurę.