Raport z Ćwiczenia 6

Bartłomiej Rasztabiga 304117

24 stycznia 2022

1 Treść zadania

Zaimplementuj algorytm Q-learning. Następnie, wykorzystując proste środowisko (np. Taxi-v3), zbadaj wpływ hiperparametrów na działanie algorytmu (np. wpływ strategii eksploracji, współczynnik uczenia).

2 Opis implementowanego algorytmu

Wykorzystanym środowiskiem jest „Taxi-v3" z pakiety gym OpenAI. W implementacji zadania wykorzystano algorytm „Epsilon-Greedy" do wyznaczania następnej akcji agenta. Parametryzowanymi wartościami są: liczba epizodów nauki, learning rate, discount rate oraz exploration rate (parametr określający wagę długoterminowej nagrody). Do ewaluacji agenta wykorzystywana jest średnia nagroda ze 100 epizodów.

3 Eksperymenty numeryczne

3.1 Wpływ liczby epizodów

Najpierw porównam wpływ liczby epizodów na średnią nagrodę ze 100 epizodów.

Do eksperymentu użyję poniższych wartości liczby epizodów:
$[1000, 2000, 5000, 10000]$
Pozostałe parametry są ustawione zgodnie z ich optymalnymi wartościami:
exploration rate = 0.5
learning rate = 0.2
discount rate = 1.0

Tablica 1: Porównanie liczby epizodów

liczba epizodów	Nagroda
1000	-18.98
2000	8.03
5000	7.76
10000	7.83

3.2 Wpływ exploration rate

Następnie porównam wpływ exploration rate na średnią nagrodę ze 100 epizodów.

Do eksperymentu użyję poniższych wartości exploration rate:
$[0.5, 0.1, 0.01, 0.001, 0.0001]$
Pozostałe parametry są ustawione zgodnie z ich optymalnymi wartościami:
liczba epizodów = 4000
learning rate = 0.2
discount rate = 1.0

Tablica 2: Porównanie exploration rate

exploration rate	Nagroda
0.5	7.38
0.1	8.28
0.01	7.76
0.001	7.56
0.0001	7.78

3.3 Wpływ learning rate

Następnie porównam wpływ learning rate na średnią nagrodę ze 100 epizodów.

Do eksperymentu użyję poniższych wartości learning rate:
$[1.0, 0.8, 0.6, 0.4, 0.2, 0.1, 0.01]$
Pozostałe parametry są ustawione zgodnie z ich optymalnymi wartościami:
liczba epizodów = 4000
exploration rate = 0.5
discount rate = 1.0

Tablica 3: Porównanie learning rate

learning rate	Nagroda
1.0	8.1
0.8	8.01
0.6	8.06
0.4	8.08
0.2	7.83
0.1	8.19
0.01	-185.08

3.4 Wpływ discount rate

Następnie porównam wpływ discount rate na średnią nagrodę ze 100 epizodów.

Do eksperymentu użyję poniższych wartości discount rate:
$[1.0, 0.8, 0.6, 0.4, 0.2, 0.1, 0.01]$
Pozostałe parametry są ustawione zgodnie z ich optymalnymi wartościami:
liczba epizodów = 4000
exploration rate = 0.5
learning rate = 0.2

Tablica 4: Porównanie discount rate

discount rate	Nagroda
1.0	7.04
0.8	5.79
0.6	-1.94
0.4	-8.34
0.2	-16.47
0.1	-51.72
0.01	-159.78

4 Wnioski z przeprowadzonych badań

4.1 Wpływ liczby epizodów

Oczywistym wnioskiem jest zwiększanie się średniej nagrody wraz ze wzrostem liczby epok wykorzystanych do trenowania agenta. Wzrost nagrody następuje do pewnej wartości liczby epizodów, a następnie oscyluje w okolicach swojego maksimum, wtedy, kiedy agent jest już najlepiej wytrenowany.

4.2 Wpływ exploration rate

Agent najlepiej zachowuje się dla exploration rate równego około 0.1. Wynika to z tego, że przy zadanym exploration rate, agent tylko w 10 % przypadków wybiera akcję losowo. Jest to lepsza strategia, niż wybieranie zawsze akcji na podstawie Q table, ponieważ motywuje eksplorację.

4.3 Wpływ learning rate

Agent najlepiej radzi sobie dla learning rate równego 1.0, chociaż w przedziale [0.1, 1.0] nie widać wyraźnie najlepszej wartości. Natomiast dla learning rate poniżej 0.1, algorytm Q learning nie dopasowuje się do środowiska i nie uzyskuję dobrych wartości średnich nagrody.

4.4 Wpływ discount rate

Agent osiąga najlepsze wyniki dla discount rate w okolicach 1.0. Wartości mniejsze od 1.0 powodują znacząco słabsze zachowanie algorytmu. Oznacza to, że waga przyszłych (długoterminowych nagród) w wybranym środowisku jest ważna dla optymalnego rozwiązania.