Piotr Paturej
Nr albumu 304102

WSI Zadanie 7

Tworzenie algorytmu Q-learning w celu wyznaczenia polityki decyzyjnej dla problemu FrozenLake8x8 zacząłem od zdefiniowania środowiska po jakim może poruszać się agent tzn. stanów, w których może się znajdować, akcji, które może w nich podjąć i nagród za znalezienie się w każdym ze stanów. Zdecydowałem się na ustawienie wartości nagród na -1 dla „bezpiecznych pól" jeziora, -100 dla dziur i 100 dla miejsca docelowego. Stworzyłem funkcje pomocnicze do sprawdzania, czy aktualny stan jest terminalny (czy nagroda za bycie w nim nie wynosi -1) i do wyznaczania następnego ruchu. Funkcja ta z prawdopodobieństwem epsilon wybierała ruch gwarantujący najwyższą nagrodę, a z prawdopodobieństwem 1 – epsilon ruch losowy. Zdecydowałem się na wykorzystanie tego rodzaju funkcji by algorytm zazwyczaj uczył się najlepszej trasy, ale żeby czasem też eksplorował swoje środowisko.
Uczenie się agenta polegało na powtarzaniu następujących kroków przez 1000 iteracji:

1. Ustawienie pozycji startowej.
2. Wykonanie ruchu do nowego stanu
3. Sprawdzenie nagrody za aktualny stan
4. Obliczenie różnicy czasowej za pomocą wzoru:

$$TD = R + D_f \cdot \max(Q_l) - Q_n$$

   gdzie:
   TD – różnica czasowa (temporal difference)
   R – nagroda,
   D_f – stopa dyskontowa (discount factor),
   Q_l – lista Q-wartości dla aktualnej pozycji
   Q_n – Q-wartość dla ostatniego stanu i wykonanej akcji

5. Uaktualnienie odpowiedniej Q-wartości dla ostatniego stanu:

$$Q_{n+1} = Q_n + L_r \cdot TD$$

   gdzie Lr to współczynnik uczenia (learning rate)

6. Jeśli aktualny stan nie jest terminalny, powtórzenie kroków 2 – 6.

Testowanie prawidłowości i szybkości działania algorytmu

Algorytm znalazł dla „jeziora” 8x8 drogę zaznaczoną na zielono:

SFFFFFFF
FFFFFFFF
FFFHFFFF
FFFFFHFF
FFFHFFFF
FHHFFFHF
FHFFHFHF
FFFHFFFG

Czerwonym kolorem zaznaczone są dziury. Jest to jedna z najkrótszych możliwych dróg dla tego jeziora. Dla różnych ziaren generatora liczb pseudolosowych algorytm znajdował też inne drogi, ale wszystkie miały długość 14.

Zmierzyłem czas działania algorytmu w zależności od dobrania wartości współczynnika uczenia się, stopy dyskontowej i wartości epsilon.
Poniżej przedstawiam zmierzone czasy:

Średni czas (sekundy)	Współczynnik uczenia	Stopa dyskontowa	Wartość epsilon
0.29	0.8	0.8	0.8
0.25	0.8	0.8	0.9
0.27	0.8	0.9	0.8
0.23	0.8	0.9	0.9
0.28	0.8	1.0	0.8
0.25	0.8	1.0	0.9
0.28	0.9	0.8	0.8
0.24	0.9	0.8	0.9
0.27	0.9	0.9	0.8
0.24	0.9	0.9	0.9
0.26	0.9	1.0	0.8
0.25	0.9	1.0	0.9

0.27	1.0	0.8	0.8
0.25	1.0	0.8	0.9
0.28	1.0	0.9	0.8
0.25	1.0	0.9	0.9
0.29	1.0	1.0	0.8
0.26	1.0	1.0	0.9

Jak widać różnice w szybkości działania algorytmu są bardzo małe. Największy wpływ zdaje się mieć wartość stałej epsilon, ale jest on pomijalnie mały. Wartości tych stałych nie mają znaczącego wpływu na szybkość działania algorytmu.