

Algorytmy Inspirowane Biologicznie - algorytm genetyczny "Snake"

Piotr Koproń

Mikołaj Leonhardt

Jan Augustyn

Maj 2023

1 Internal Notes - remove when done

evolution.py -i do uczenia wunsza. Zapisuje to best_snake.save() Parametry algorytmu genetycznego są evolution history,statistics (linia 79). Zapisuje na konsolę, zapisuje i Wypluwa wykresy na ekran.

snake/utlils/snake'config.py -i do zmiany ustawień mapy. Asercja że jak pozmieniam to przeżyje.

2 Abstrakt

W tym raporcie rozważamy zastosowania algorytmów genetycznych do szkolenia sztucznej inteligencji realizującej klasyczną grę "Snake".

Struktura gry "Snake" Rzecz dzieje się na kwadratowej planszy 2D o stałej, danej parametrem wielkości. Na tej planszy zawsze znajdują się: **wąż** i **jabłko**. Wąż początkowo zajmuje 3 komórki owej planszy, jabłko - jedną. Wąż będzie przesuwiał się po planszy, unikając ścian planszy oraz swojego ogona, próbując zaś zjeść jak najwięcej jabłek. Przy czym, za każdym razem gdy wąż zje jabłko (przesuwając swoją głowę na pole jabłka), wydłuża się o jedną kratkę. Gdy głowa węża trafi w ścianę bądź w inną część węża, gra się kończy.

Struktura sztucznej inteligencji Celem naszej sztucznej inteligencji będzie wybranie optymalnego kierunku ruchu dla głowy węża, z czterech dostępnych opcji - do góry, w dół, w lewo i w prawo. Wybór podejmie na podstawie **senso-rów** - tego "co wąż widzi" - oraz jego **DNA**, które będzie ustalone na początku gry.

Struktura algorytmu genetycznego Żeby wyszkolić naszą inteligencję, korzystamy z algorytmu genetycznego. Rzecz będzie się działa następująco. Rozpocniemy od stworzenia pewnej **populacji** węży o losowo utworzonym DNA. Następnie każemy im grać, i zapiszemy ich rezultaty. Będzie to **generacja** 1. Zapewne skuteczność SI będzie dość mizerna, ale nie to nas interesuje. Po zapisaniu rezultatów, dokonamy pewnych modyfikacji do naszej populacji, i spróbujemy ponownie. W ten sposób utworzymy **generację** 2. Taką pętlę tworzenia nowych generacji będziemy wykonywać "do skutku" - efektywnie jedynym ograniczeniem jest moc obliczeniowa.

3 Implementacja

Implemetujemy kompleksową symulację, zarówno z samym algorytmem genetycznym, jak i środowiskiem szkoleniowym generującym informacje o jakości danej sztucznej inteligencji.

Struktura DNA Nasza SI węża jest de facto siecią neuronową o stałej ilości neuronów i stałymi połączeniami między nimi. Jedyne co się zmienia to wagi jakie każdy neuron przyjmuje na wejściu, co daje DNA jako ciąg liczb rzeczywistych. Przy czym należy zauważyć, że z perspektywy algorytmu genetycznego nie ma znaczenia, jak DNA jest interpretowane. W naszym przypadku mamy do czynienia z $24 \cdot 32 + 12 \cdot 24 + 4 \cdot 12$ liczbami składającymi się na DNA.

Struktura sensorów SI ma 32 różne sensory, odpowiedzialne za odczytywanie informacji z planszy. 24 z nich są odpowiedzialne za "wzrok" węża, 4 - za informowanie o obecnym kierunku głowy, 4 zaś - informowanie o obecnym kierunku ogona węża.

Wzrok Wąż "widzi" w 8 kierunkach - w osiach kardynalnych oraz po przekątnych.

W każdym kierunku wąż otrzymuje następujące informacje:

1. Odległość od ściany. 2. Odległość od jabłka, jeśli jest w tym kierunku. 3. Odległość od części ciała węża, jeśli jest w tym kierunku.

Kierunek głowy/ogona Sensory kierunku głowy/ogona są implementowane binarnie - odpowiadają na pytanie "Czy (głowa/ogon) jest zwrócona w kierunku tym a tym", gdzie dostępne kierunki to osie kardynalne.

3.1 Charakterystyka algorytmu genetycznego

Podczas przygotowywania każdej generacji algorytm genetyczny dzieli się na trzy fazy: selekcji, krzyżowania i mutacji. TODO: Opisać precyzyjnie

Selekcja Dokonujemy selekcji dwustopniowej - najpierw korzystamy z selekcji elity [BIBLIO], tj. wybieramy 5% najlepszych węży, które przechodzą proces selekcji od razu do następnego pokolenia. Następnie przechodzimy do selekcji "koła ruletki" [biblio] - każdy wąż otrzymuje ilość losów proporcjonalną do jego skuteczności, i losujemy które losy okażą się wygrywające. Wygrywające losy tworzą pary węży, które przejdą do procesu krzyżówki w ilości niezbędnej do odtworzenia wielkości populacji.

Krzyżowanie Proces selekcji dał nam listę par węży, gdzie każde dziecko weźmie połowę DNA od każdego rodzica.

Mutacja Z podanym parametrem prawdopodobieństwem wybieramy w każdym wężu konkretne wartości z DNA które zostaną poddane mutacji. Do tych wartości zostanie dodana wartość losowa z rozkładu normalnego.

4 Rezultaty nauki

Czego się nauczył?

Parametry konfiguracyjne W ramach konfiguracji możemy zmieniać:
Wielkość mapy,
Wielkość populacji w każdej generacji,
Ilość generacji (długość treningu),
funkcję opisującą skuteczność SI,
parametry selekcji/krzyżowania/mutacji.

4.1 Rezultaty z zestawu parametrów 1

Zestaw parametrów 1

Rezultaty nauki Ładne grafy i komentarze tutaj.

4.2 Rezultaty z zestawu parametrów 2

Zestaw parametrów 2

Rezultaty nauki Ładne grafy i komentarze tutaj.

5 Wnioski

Jeżeli na jakieś wpadniemy. Może popatrzeć jak wunsch wyćwiczony na małej mapie radzi sobie na dużej?

6 Bibliografia