

Zadanie domowe nr 3

Sieci neuronowe i czołgi

Dawid Ćwik

Informatyka II

gr. 1

238137

1. Trenowanie sieci

Język R umożliwia korzystanie z biblioteki o nazwie *neuralnet*. Dzięki podaniu odpowiednich danych oraz sposobu ich interpretacji jesteśmy w stanie wygenerować informacje na temat sieci neuronowej.

Moja sieć korzystała z 7 kolumn wejściowych oraz 4 wyjściowych. Sieć posiadała 5 warstw ukrytych.

INPUT	OUTPUT
myTank.x	myTank.controls.turnLeft
myTank.y	
myTank.rotation	myTank.controls.turnRight
myTank.ShootCooldown	
enemyTank.x	myTank.controls.goForward
nemyTank.Y	
currentGameTime	myTank.controls.shoot

Ważnymi parametrami podczas budowania sieci były zmienne

- *Stepmax* – Maksymalna ilość kroków możliwa do obliczenia neuronu. w moim przypadku wynosiła $1e+06$. Często podczas testowania różnych wariacji kolumn wejściowych oraz ilości danych, zbyt mały parametr tworzył błąd uniemożliwiający dalsze obliczanie.
- *Threshold* – parametr odpowiedzialny za eliminowaniu neuronów obciążonych podanym błędem. W moim przypadku wynosił 0.01. W niektórych przypadkach testowanych przy tym parametrze sieć tworzyła się kilka godzin, kiedy po zmianie na 0.1 czas skracał się do kilkunastu minut.

Ciekawym faktem okazał się że dodanie kolejnej kolumny wejściowej może wpłynąć na ograniczenie czasu powstawania sieci. Początkowo tworzyłem sieć dla 6 inputów, bez zmiennej *currentGameTime*. Przy parametrach podanych powyżej, sieć ta powstawała kilkanaście godzin. Po dodaniu jednej kolumny czas wyniósł ok. 30 sekund(!).

Dodatkowo zaobserwowałem pozytywny wpływ normalizacji danych na czas działania.

Zdecydowałem się na takie kolumny z powodu sposobu generowania danych, opisane w następnym punkcie.

2. Dane

Dane generowałem sterując czołgiem manualnie. Moim zamysłem było dostanie się do środka cały czas strzelając. Po dostaniu się w centrum planszy czołg rotował się w lewo lub w prawo w kierunku przeciwnika. Ostatnie nie interesowały mnie kolumny takie jak jazda do tyłu ponieważ, nigdy nie korzystałem z tej możliwości. Tak samo rotacja lufą niezależnie od czołgu została pominięta przy graniu, więc nie brałem jej pod uwagę. Skupiłem się na mniejszym celu, a nie dodatkowo unikania pocisków.

Rozegrałem kilkanaście gier gdzie moim przeciwnikiem były skrypty:

- czołg nie rusza się z miejsca startu
- czołg robi kółka
- czołg jedzie do przodu
- czołg jedzie do przodu i kręci się raz w prawo, raz w lewo

Ostatecznie wygenerowałem około 10MB danych. Do wczytania ich jako zmiennej w projekcie potrzebowałem je przerobić komentami linux „awk” oraz „cut”. Kopiowanie danych z konsoli miało zapisane miejsce ich tworzenia „main.js” oraz co druga linijka była pusta.

```
Main.js:462 194.551,293.202,301.5,301.5,0.491,-0.812,0.026
Main.js:462 195.042,292.39,301.5,301.5,0.491,-0.812,0.026,
Main.js:462 195.533,291.578,301.5,301.5,0.491,-0.812,0.026
Main.js:462 196.025,290.766,301.5,301.5,0.491,-0.812,0.026
Main.js:462 196.516,289.955,301.5,301.5,0.492,-0.812,0.026
Main.js:462 197.008,289.143,301.5,301.5,0.492,-0.812,0.026
Main.js:462 197.5,288.331,301.5,301.5,0.492,-0.811,0.026,-
```

```
awk 'NF' inf.ug.edu.pl-1546543741136.log > test
cut -c13- test > test2
```

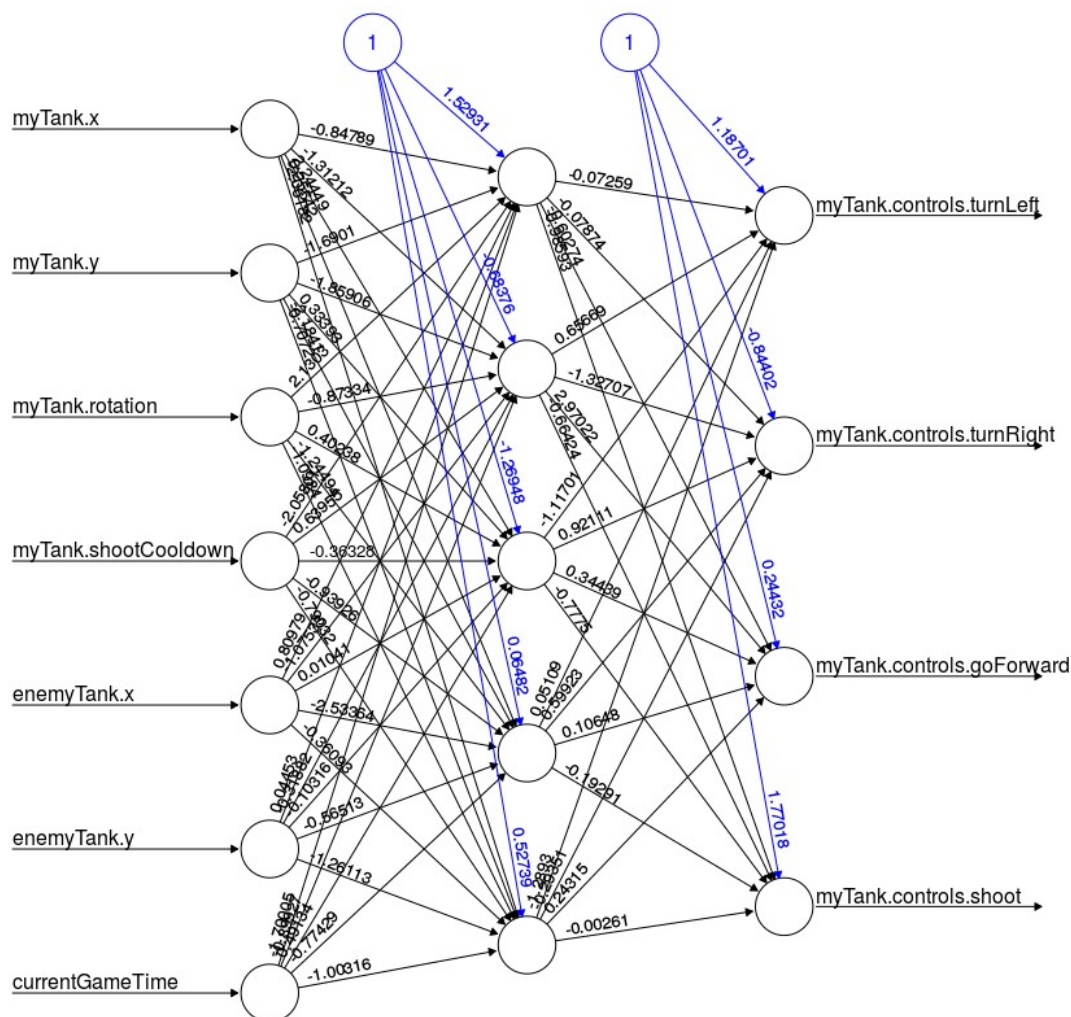
[illegible]

W języku R obróbka danych polegała na wybraniu kolumn które mnie interesują, było ich 11. Następnie normalizacja danych oraz podział na zbiory testowy oraz treningowy.

```
#WCZYTANIE DANYCH
dane <- read_csv("dane.csv")
dane <- data.frame(dane[1:3], dane[9:12], dane[14], dane[29:30], dane[50])
#NORMALIZACJA DANYCH
dane.norm <- norm(dane)|
#PODZIAL NA ZBIORY TRENINGOWY I TESTOWY
set.seed(1234)
ind <- sample(2, nrow(dane), replace=TRUE, prob=c(0.67, 0.33))
dane.train <- dane[ind==1, 1:length(dane)]
dane.test <- dane[ind==2, 1:length(dane)]
```

3. Podsumowanie

Ostatnie wygenerowana sieć prezentuje się następująco



Error: 8107.127464 Steps: 51

Zmienne wagowe oraz bias:

```
> toString(bias1)
[1] "1.52931647304191, -0.683762815022971, -1.26947564545938, 0.0648237748558423, 0.527393151442628"
> toString(bias2)
[1] "1.18700958259448, -0.844024517760241, 0.244329356391131, 1.77019334724042"
> toString(weights1)
[1] "-0.847886847134174, -1.69009844513508, 3.06082122429591, -2.05805572880234, 0.809794106538265, 0.0445363817138832,
-1.70003623705589, -1.31212426358934, -1.85906380427056, -0.873337728989258, 0.639547584631172, -1.07532897238582, -0.31
8821131529731, -0.899273884722628, 2.2444870795631, 0.333932778767224, 0.402384948980428, -0.363276866287009, 0.01040526
65773477, -0.103158796317718, 0.491342451426235, 0.255418201458929, -2.18410165377388, -1.24493598306807, -0.93925664539
3268, -2.53364383615537, -0.565125759473863, -0.774293306778966, -2.06181715891916, -0.707202235714628, -1.0983962005076
8, -0.790318649767228, -0.360932941048802, -1.26113428214927, -1.00315666162677"
> toString(weights2)
[1] "-0.0715219724498689, 0.656687898123871, -1.11701070787621, 0.0510900443252304, -1.28929716883776, -0.07875775052738
48, -1.32706630039177, 0.921111283344343, 0.599225576139466, -0.293513662563517, -0.601571880151595, 2.97021718898804, 0
.344393829236086, 0.106479198779836, 0.243149041412368, -0.984868534275229, -0.664241851521269, -0.777490857196657, -0.1
92908636340889, -0.00261295932656624"
```

Sam projekt bardzo ciekawy ze względu na formę rywalizacji oraz wprowadzenie gry. Jest to rzadkość w poleceniach projektów uczelnianych. Niestety jak tytuł wskazuje miał on dotyczyć sieci neuronowych, a ostatecznie moje odczucia są mieszane i bardziej skłaniałbym się do tematu sprawdzenia naszego logicznego myślenia i wyciągania wniosków na podstawie obserwacji danych. Ich Sama sieć neuronowa i jej tworzenie było chaotyczne, raczej metodą prób i błędów. Pomimo zrozumienia sposobu działania metody neuralnet, zgłębienia tematu warstw ukrytych, biasów, wejść i wyjść w sieciach neuronowych, efekt mojej pracy jest mało zadowalający, a sam czołg nie jest w stanie wykonywać choć trochę przemyślanych ruchów.

Być może moje dane z powodu tworzenia ich manualnie i interpretowania mojej myśli jako dobrej, komputer nie mógł zrozumieć zamysłu jechania do środka i strzelania w stronę czołgu wroga.

Pomimo 10MB danych, obliczaniu sieci neuronowej na kilkanastu sposobów i czekaniu kilkanastu godzin na wyniki które ostatecznie kończyły się niepowodzeniem, mój czołg nie jest w stanie robić niczego poza jeżdżeniem w kółko i przypadkowym strzelaniem.