**Urszula Król, grupa projektowa 2**

**Sprawozdanie nr 1**

**Temat ćwiczenia:**

Budowa i działanie perceptronu

**1. Cel ćwiczenia**

Celem ćwiczenia było poznanie budowy i działania perceptronu poprzez implementację oraz

uczenie perceptronu realizującego wybraną funkcję logiczną dwóch zmiennych.

**2. Zadania wykonane w ramach ćwiczenia**

a) Implementacja sztucznego neuronu.

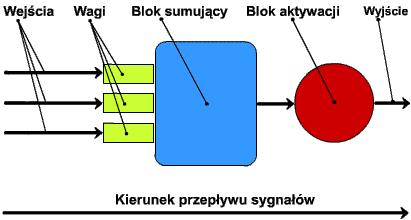
b) Wygenerowanie danych uczących i testujących wybranej funkcji logicznej dwóch zmiennych.

c) Uczenie perceptronu dla różnej liczby danych uczących i różnych współczynników uczenia.

d) Testowanie perceptronu.

**3. Syntetyczny opis budowy oraz wykorzystanego algorytmu uczenia**

Sztuczny neuron to prosty system przetwarzający wartości sygnałów wprowadzanych na jego wejścia w pojedynczą wartość wyjściową, wysyłaną na jego jedynym wyjściu. To podstawowy element [sieci neuronowych](https://pl.wikipedia.org/wiki/Sie%C4%87_neuronowa), która jest jedną z metod [sztucznej inteligencji](https://pl.wikipedia.org/wiki/Sztuczna_inteligencja). Pierwowzorem zbudowania sztucznego neuronu był [biologiczny neuron](https://pl.wikipedia.org/wiki/Neuron).



Sztuczna sieć neuronowa to ogólna nazwa struktur matematycznych i ich programowych lub sprzętowych modeli, realizujących obliczenia lub przetwarzanie sygnałów poprzez rzędy elementów wykonujących pewną podstawową operację na swoim wejściu, zwanych neuronami.

Perceptron to prosty element obliczeniowy. Sumuje ważone sygnały wejściowe oraz porównuje ją z progiem aktywacji. Zależnie od wyniku perceptron może być wzbudzony albo nie.

Do uczenia perceptronu można użyć algorytmu uczenia perceptronu, czyli automatycznego doboru wag na podstawie kolejnych przykładów. W uproszczonym przypadku algorytm wygląda następująco:

* Inicjujemy wagi losowo.
* Dla każdego przykładu uczącego obliczamy odpowiedź perceptronu.
* Jeśli odpowiedź perceptronu jest nieprawidłowa, to modyfikujemy wagi:

w1 += n \* (d-y) \* x1   
w2 += n \* (d-y) \* x2   
b   += n \* (d-y)

gdzie:

n - niewielki współczynnik uczenia (n > 0),

d - oczekiwana odpowiedź,

y - odpowiedź neuronu.

Po wykorzystaniu wszystkich przykładów rozpoczyna się proces uczenia na nowo, do momentu gdy następują jakiekolwiek zmiany wag połączeń.

**4. Informacje o programie:**

-zadaniem perceptronu jest wywnioskowanie, czy suma danych liczb jest liczbą dodatnią;

-wagi ustalane są losowo;

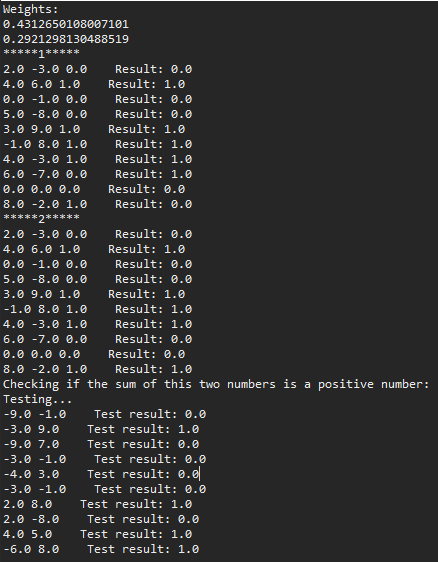
-współczynnik uczenia wynosi 0.1;

-zestaw uczący składa się z 10 par liczb.

**5. Zrzuty ekranu z działania programu:**

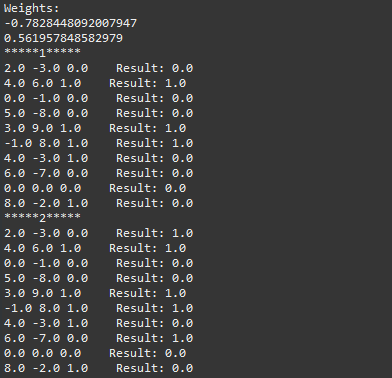
Dla wyżej podanych ustaleń program działa jak poniżej:

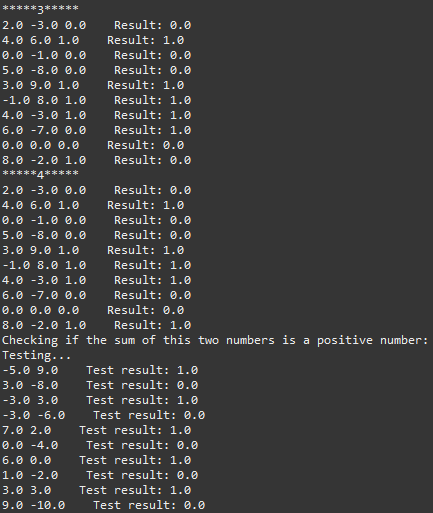
Próba 1:



- Jedynie dwie iteracje

Próba 2:



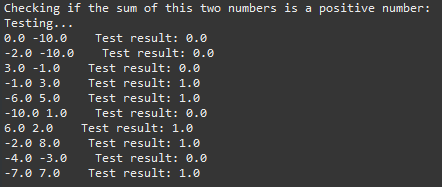


Tym razem liczba iteracji wynosi 4.

Jak widać wagi ustalane są ponownie do momentu aż wynik założony i wynik uzyskany będą się całkowicie zgadzać. W programie założono, iż pętla wykonuje się dopóki błąd jest większy od 0.01. W pierwszej próbie liczba iteracji wyniosła 2, a w drugiej 4. Różnica ta jest spowodowana wylosowanymi wagami.

Próba 3:

Zmniejszono liczbę danych uczących. Ich liczba wynosi teraz 6.



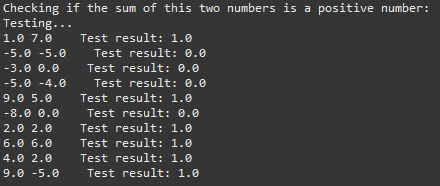
Zaobserwowano, że popełnione zostały błędy:

-6+5=-1, -1 nie jest dodatnie

-7+7=0, 0 nie jest dodatnie

Próba 4:

Zmiana współczynnika uczenia na 0.001



- liczba iteracji wzrosła do 28

- wyniki testu są poprawne

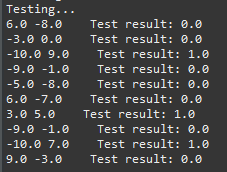
-wagi:

-0.4326586458624204

-0.12495485205577173

Próba 5:

Zmiana współczynnika uczenia na 0.005



- liczba iteracji wyniosła 14

- 3 wyniki testu są błędne, np.: -10+9=-1, -1 nie jest dodatnie

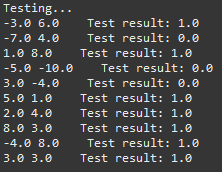
- wagi:

-0.6579374551613637

0.8933155051422343

Próba 6:

Zmiana współczynnika uczenia na 0.01



- liczba iteracji wyniosła 6

- wyniki testu są poprawne

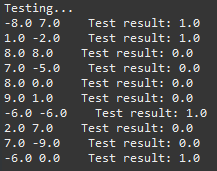
- wagi:

-0.39377267341364064

-0.7073811240580776

Próba 7:

Zwiększenie liczby danych do 20



- otrzymano całkowicie błędne wyniki.

**6. Wnioski:**

- działanie perceptronu jest uzależnione od wielu czynników: współczynnika uczenia, wag oraz liczby danych uczących,

- współczynnik uczenia wpływa na czas i efekt uczenia,

- wraz ze zwiększaniem wartości współczynnika uczenia zmniejsza się liczba iteracji, co ma wpływ na działanie programu i powoduje uzyskanie błędnych wyników,

- wraz ze zmniejszaniem wartości współczynnika uczenia poprawia się działanie programu i otrzymywane wyniki są poprawne, jednak przy zbyt małych wartościach pojawiają się błędy

- dobór wag ma bezpośredni wpływ na działanie perceptronu, w zależności od ich wartości zmienia się poprawność wyników i liczba iteracji,

- dane uczące wpływają na poprawność uczenia perceptronu, przy zbyt małej ilości otrzymujemy błędne wyniki,

- zbyt duża liczba danych uczących powoduje, iż uzyskane wyniki są błędne,

- bardzo ważny jest odpowiedni dobór współczynnika uczenia, wag oraz liczby danych uczących, gdyż mają bezpośredni wpływ na działanie perceptronu.