Sprawozdanie PSI

**Perceptron**

**Opis budowy:**

Program przyjmuje jednostki uczące, które dla danych wartości punktów (x,y) znana jest przynależność do klasy '1' lub '0'. Chcemy więc znaleźć wagi dla których błąd w stosunku wartości oczekiwanych będzie jak najmniejszy. Za każdą iteracją pętli głównej zmieniana jest waga w zależności od współczynnika uczenia oraz błędu w stosunku do wartości oczekiwanej. Po otrzymania oczekiwanych wyników program przestaje działać.

**Wyniki:**

Dla 100 jednostek uczących są następujące:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ilość iteracji | 4 | 5 | 6 | 4 | 5 | 3 | 3 | 3 | 3 |
| LEARNING\_RATE | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |

Dla 25 :

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ilość iteracji | 9 | 4 | 9 | 5 | 5 | 9 | 5 | 5 | 9 |
| LEARNING\_RATE | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |

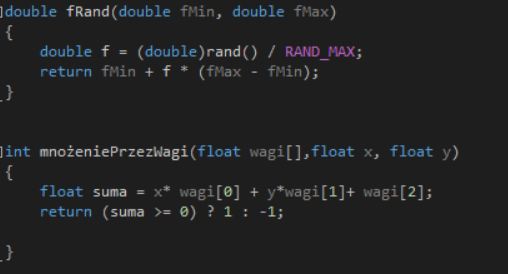
**Wnioski:**

Łatwo można zauważyć, że dla większej ilości jednostek uczących szybciej otrzymamy wyniki.

Dla mniejszej częstotliwości uczenia liczba w przypadku dużej ilości jednostek uczących iteracji potrzebnych do otrzymania oczekiwanych wyników jest większa niż dla wyższych wartości, jednak bezpieczniej jest używać niższych wartości, gdyż zmniejsza ryzyko "przeskoczenia" szukanej wartości.

Wpływ na ilość iteracji ma nie tylko ilość jednostek uczących oraz learning rate, lecz również startowe wartości wag. Dla niektórych wartości o wiele szybciej znajduje szukaną wartość, jednak ciężko znaleźć zależność.

**Listing:**

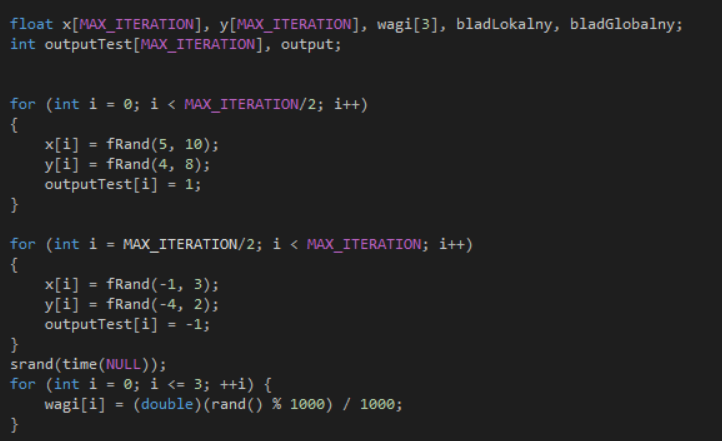


- funkcja losująca z danego przedziału

-funkcja mnożąca przez wagi oraz

funkcja aktywacji

- -tworzenie jednostek uczących



- algorytm uczenia

