

Sprawozdanie 6:

Budowa i działanie sieci Kohonena dla WTA

1. Cel projektu:

Celem ćwiczenia było poznanie budowy i działania sieci Kohonena dla WTA na podstawie programu zrealizowanego w środowisku MatLab. Podajemy cechę kwiatu, a sztuczna sieć neuronowa określa, do którego neuronu jest najbliżej.

2. Pojęcia niezbędne do realizacji projektu:

WTA(Zasada WTA):

[*Winner Takes All*]

Najprościej mówiąc zasada WTA głosi, że tylko jeden neuron może być aktywny, pozostałe są w stanie spoczynkowym.

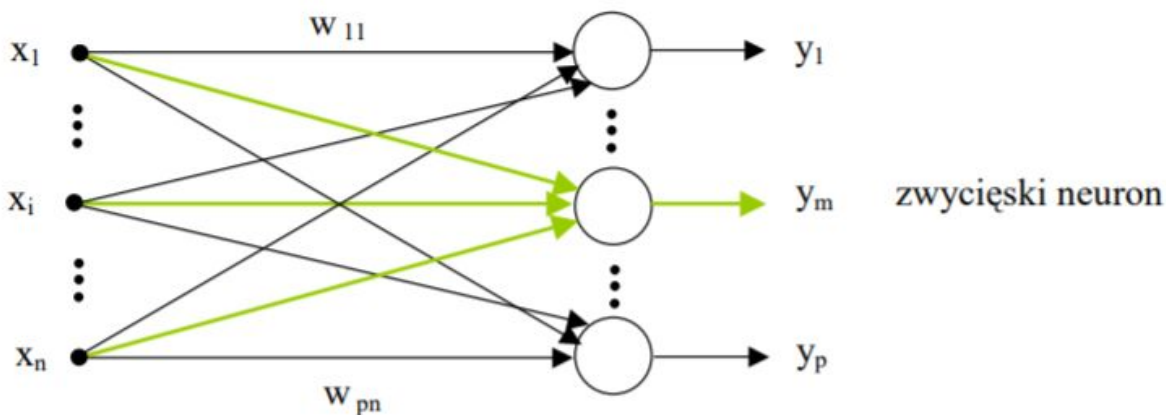
Tylko neuron zwycięski (indeks "c" -> conqueror) uaktualnia swe wagi, (zbliża się do wektora $x(k)$).

Wagi zwycięskiego neuronu mogą zostać uaktualnione np. na podstawie wzoru Kohonena:

Wzór Kohonena:

$$w_c(k+1) = w_c(k) + \eta(k)[x(k) - w_c(k)]$$

Przy podaniu na wejście sieci wielu wektorów zbliżonych do siebie będzie zwyciężać ciągle ten sam neuron, w wyniku czego jego wagi będą odpowiadać uśrednionym wartościom wektorów wejściowych, dla których dany neuron był zwycięzcą. Neurony nie wygrywające nie zmieniają swoich wag – pozostają martwe.



Uczenie metodą WTA:

- Na wstępie przyjmuje się losowe względem 1 wartości wag poszczególnych neuronów.
- Po podaniu pierwszego wektora wejściowego **x** wyłaniany jest zwycięzca o numerze **k**.
- Neurony przegrywające mają na wyjściu stan zero, co blokuje proces aktualizacji ich wag.

3. Kod programu i wyjaśnienia:

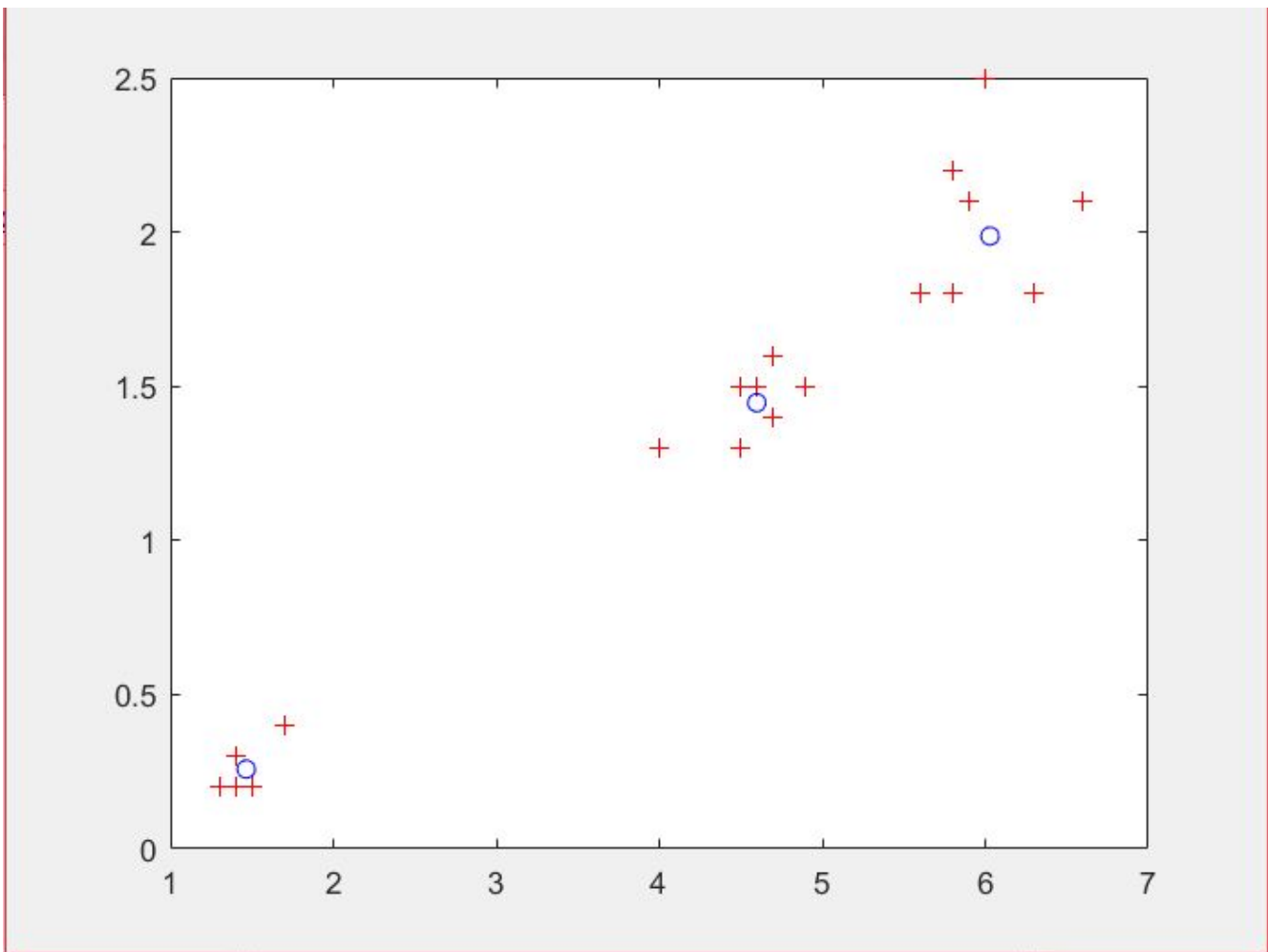
```

8      % Wektor X podany wyżej
9      plot(x(3,:),x(4,:),'+r');
10     title('Input Vectors');
11     xlabel('x(1)');
12     ylabel('x(2)');
13     net = competlayer(3,.2);
14
15     w = eye(4,2)
16     plot(x(3,:),x(4,:),'+r');
17     hold on;
18
19     net.trainParam.epochs = 80;
20     net = train(net,x);
21     w = net.IW{1};
22     plot(w(:,3),w(:,4),'ob');
23     x1 = [0; 0.2;0;0];
24     y = net(x1)

```

W powyższym kodzie tworzę sieć neuronową, która składa się z 3 neuronów. Sieć uczy metodą WTA. Liczba ustalonych przeze mnie iteracji to 80. Wektor X podany jest wyżej i określa właściwości cech charakterystycznych wejścia (w tym przypadku kwiatów).

4. Wyjście naszego programu:



Macierz wag (wyniki zaokrąglone do jednego miejsca po przecinku):

kolumna 1	kolumna 2	kolumna 3	kolumna 4
4.7	3.6	1.2	0.5
6.6	2.8	6.1	1.9
5.9	2.9	4.8	1.5

- pierwsza i druga kolumna przedstawia wartości podobne do siebie
- znaczne zmiany występują przy kolumnie 3 i 4, co oznacza, że cechy istotne kwiatów można określić na podstawie tych właśnie kolumn.
- Algorytm WTA podczas opisu cech kwiatów spisał się dobrze.
- Neurony dopasowują swoje wagi tak, że przy grupowaniu wektorów wejściowych zbliżonych do siebie zwycięża zawsze ten sam neuron - wektor, który zwycięża.