

Politechnika Śląska
Wydział Automatyki Elektroniki i Informatyki
Kierunek Automatyka i Robotyka



Metody Sztucznej Inteligencji

Ćwiczenie 3 *Sieci neuronowe Kohonena*

Autorzy
Tomasz Kuś
Karolina Szotek
Grupa 1 TI

Data wykonania ćwiczenia: 28.10.2020

1. Cel ćwiczenia.

Celem ćwiczenia była klasteryzacja danych wygenerowanych samodzielnie, oraz rzeczywistego zbioru danych mikromacierzowych wykorzystując sieci Kohonena.

2. Klasteryzacja danych.

a) dane wygenerowane samodzielnie.

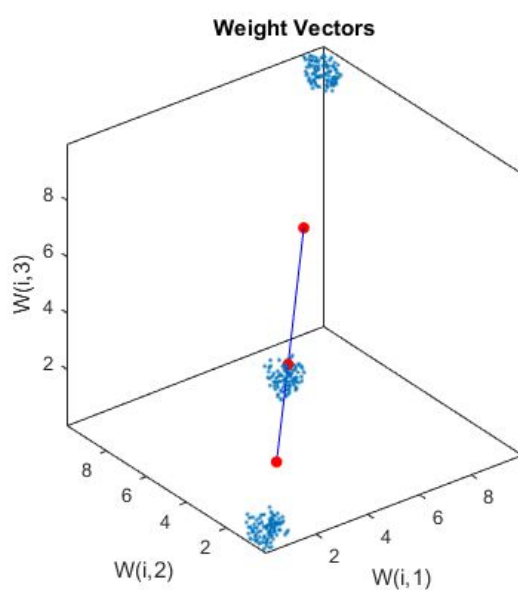
Badanie wpływu parametrów na klasteryzację. Dla każdego z parametrów przeprowadzono dwuetapową analizę danych.

- Liczba neuronów

Wpływ liczby neuronów badano dla funkcji odległości *boxlist*, topologii *randtop* oraz stu epok uczenia.

Liczba neuronów w pierwszym etapie klasteryzacji [1 9]

Liczba neuronów w pierwszym etapie klasteryzacji [1 3]

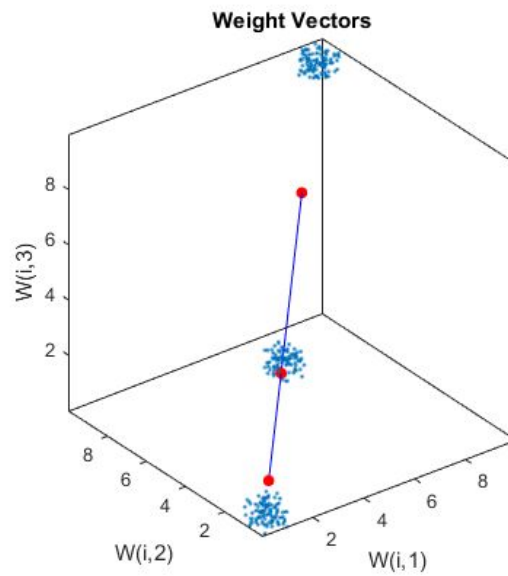


Wynik klasteryzacji :

klasa 1	klasa 2	klasa 3
0,3333	0,3333	0,3333

Liczba neuronów w pierwszym etapie klasteryzacji [1 12]

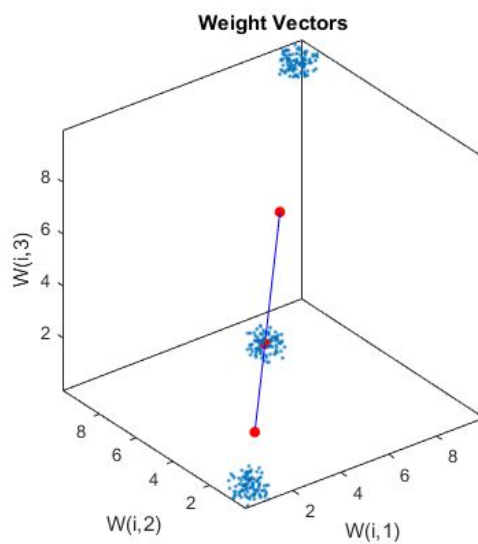
Liczba neuronów w pierwszym etapie klasteryzacji [1 3]



klasa 1	klasa 2	klasa 3
0,3333	0,3333	0,3333

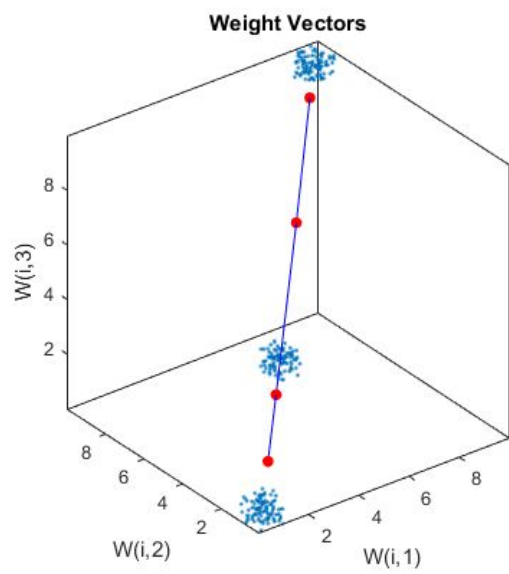
Liczba neuronów w pierwszym etapie klasteryzacji [1 21]

Liczba neuronów w pierwszym etapie klasteryzacji [1 3]



klasa 1	klasa 2	klasa 3
0,3333	0,3333	0,3333

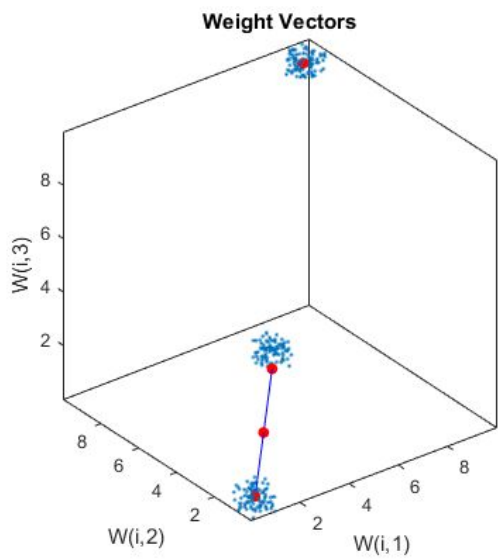
Liczba neuronów w pierwszym etapie klasteryzacji [1 9]
Liczba neuronów w pierwszym etapie klasteryzacji [1 4]



klasa 1	klasa 2	klasa 3	klasa 4
0,3333	0	0,3333	0,3333

Liczba neuronów w pierwszym etapie klasteryzacji [1 12]

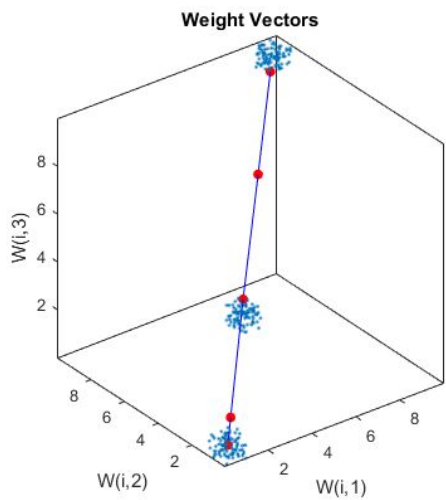
Liczba neuronów w pierwszym etapie klasteryzacji [1 4]



klasa 1	klasa 2	klasa 3	klasa 4
0,3333	0,3333	0	0,3333

Liczba neuronów w pierwszym etapie klasteryzacji [1 21]

Liczba neuronów w pierwszym etapie klasteryzacji [1 5]



klasa 1	klasa 2	klasa 3	klasa 4	klasa 5
0,3233	0,0100	0,3333	0	0

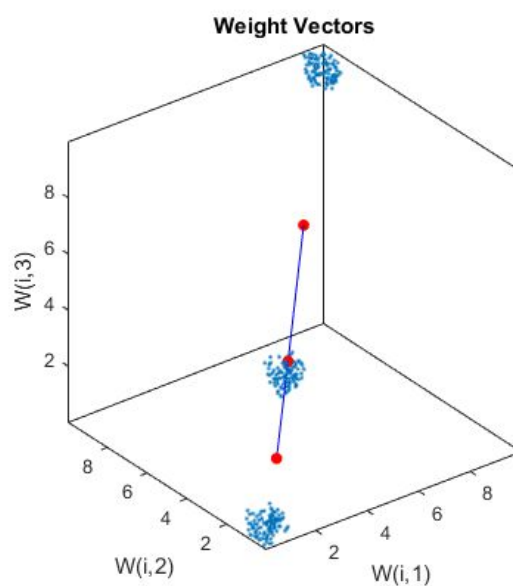
Jak można zauważyć, dla różnej liczby neuronów wyniki klasteryzacji są zbliżone. Gdy w drugim etapie klasteryzacji przypisywaliśmy neurony do liczby klas będących także liczbą zbiorów danych, klasteryzacja została wykonana poprawnie. Jednak w przypadku większej liczby neuronów w drugim etapie, niektóre klasy zostawały puste. Najgorszy wynik osiągnięto dla ostatniej symulacji. Dla dwóch klas nie zostały przyporządkowane punkty, dodatkowo na kolejnych trzech klasa punkty nie zostały przypisane równomiernie.

Dodatkowo można zauważyć, że umiejscowienie neuronów na wykresie zmienia się.

- Topologia sieci

Wpływ topologii sieci badano dla funkcji odległości *boxdist*, stu epok i 3 neuronów w drugim etapie klasteryzacji.

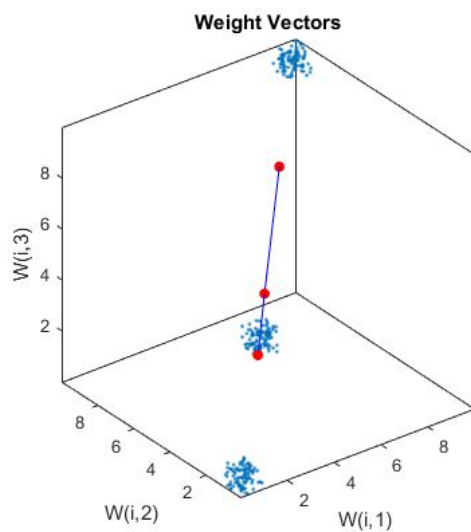
Topologia *randtop*



Wyniki klasteryzacji:

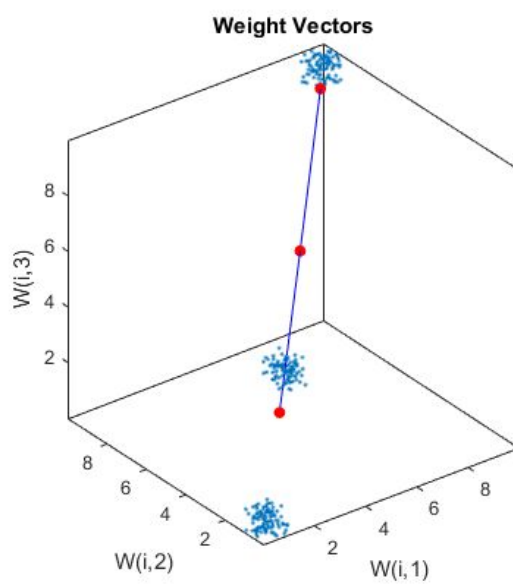
klasa 1	klasa 2	klasa 3
0,3333	0,3333	0,3333

Topologia *hextop*



klasa 1	klasa 2	klasa 3
0,3333	0,0200	0,6467

Topologia *gridtop*



klasa 1	klasa 2	klasa 3
0,3333	0	0,6667

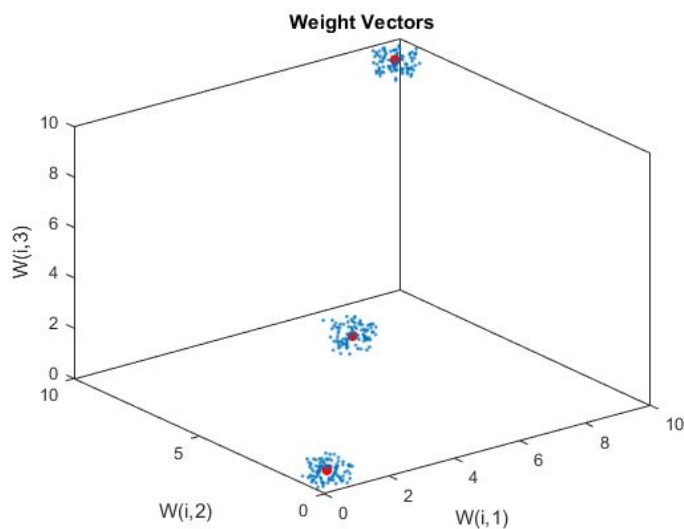
W przypadku topologii sieci, najlepszy wynik uzyskano dla topologii *randtop*. W tym przypadku podział próbek do konkretnych klas jest tak jakiego się spodziewaliśmy. W przypadku topologii *hextop* oraz *gridtop* wyniki nie są zadowalające. Próbkę nie zostały rozdzielone w sposób jakiego się spodziewaliśmy. Najgorzej wypadła metoda *gridtop* w przypadku której do jednej klasy nie została przypisana żadna próbka.

Położenie neuronów na wykresie było różne dla każdej z topologii. Jednak w przypadku topologii *randtop* położenie neuronów jest najlepsze, ponieważ pozwala na poprawną klasteryzację próbek.

- Funkcja odległości

Wpływ funkcji odległości badano dla topologii *randtop*, stu epok i 3 neuronów w drugim etapie klasteryzacji.

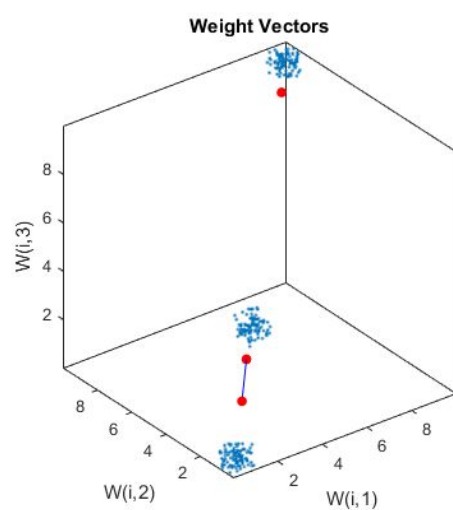
Funkcja *mandist*



Wyniki klasteryzacji:

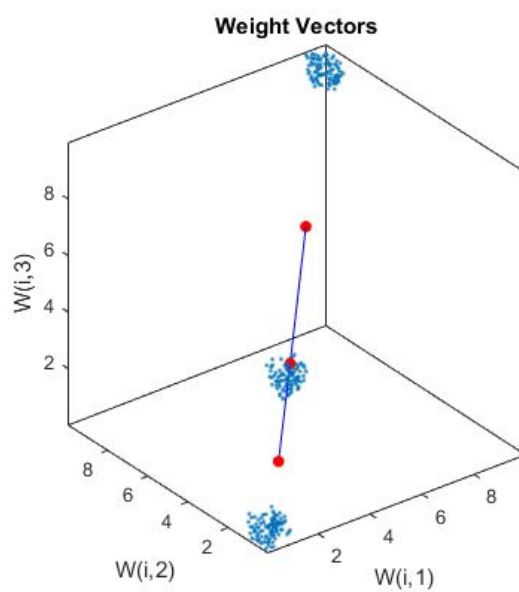
klasa 1	klasa 2	klasa 3
0,3333	0,3333	0,3333

Funkcja *linkdist*



klasa 1	klasa 2	klasa 3
0,3333	0,3333	0,3333

Funkcja *boxlist*



klasa 1	klasa 2	klasa 3
0,3333	0,3333	0,3333

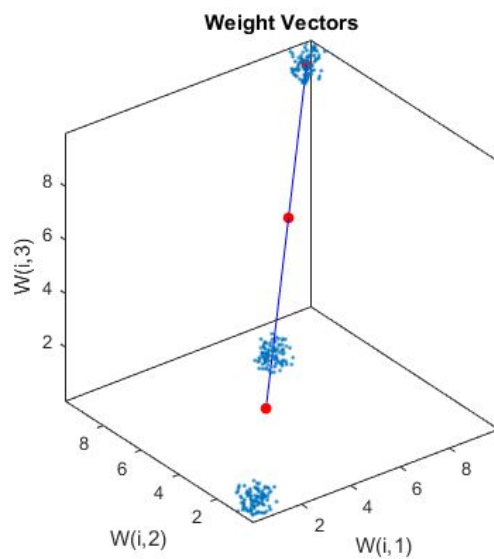
W przypadku funkcji odległości otrzymane wyniki są takie same. Wybór funkcji odległości nie wpływa więc na klasteryzację danych.

W tym przypadku położenie neuronów na wykresie także zależy od funkcji odległości. Można zauważyć, że dla funkcji mandist neurony znajdują się w środku zbiorów z próbkami. Jednak mimo różnego położenia neuronów wszystkie trzy funkcje dają taki sam wynik.

- Liczba epok uczenia

Wpływ liczby epok badano dla funkcji odległości *boxlist*, topologii *randtop* oraz 3 neuronów w drugim etapie klasteryzacji.

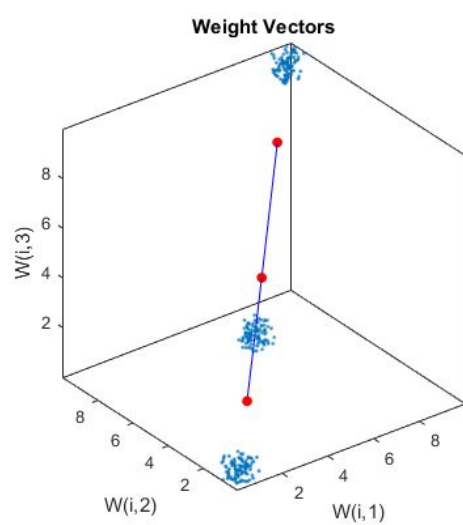
Liczba epok: 50



Wyniki klasteryzacji:

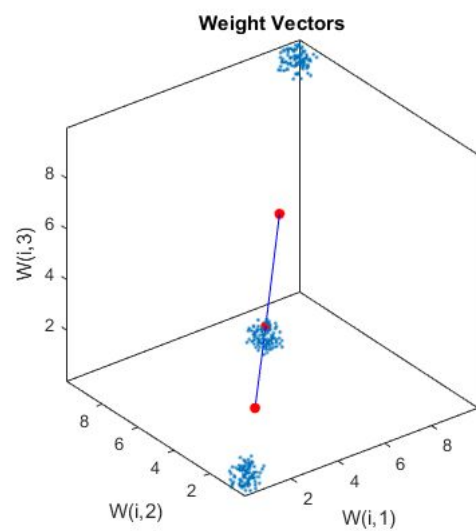
klasa 1	klasa 2	klasa 3
0,3333	0	0,6667

Liczba epok: 75



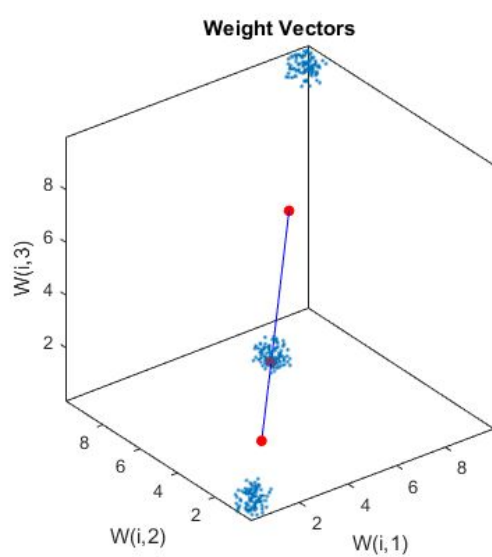
klasa 1	klasa 2	klasa 3
0,3333	0,2333	0,4333

Liczba epok: 100



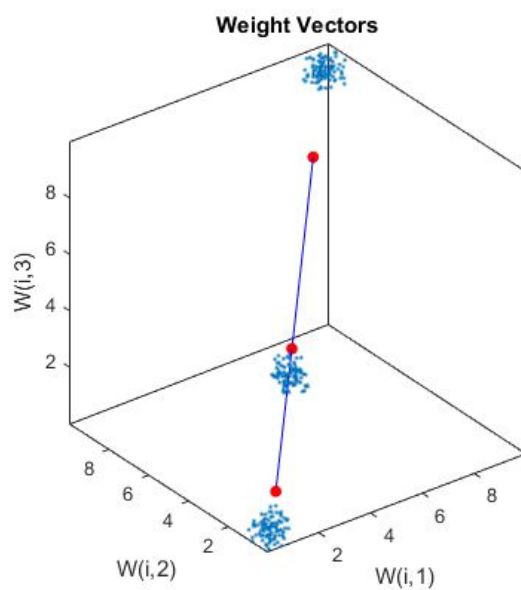
klasa 1	klasa 2	klasa 3
0,3333	0,3333	0,3333

Liczba epok: 200



klasa 1	klasa 2	klasa 3
0,3333	0,3333	0,3333

Liczba epok: 300



klasa 1	klasa 2	klasa 3
0,3333	0,3333	0,3333

Analizując powyższe wyniki, można zauważyć, że dla małej liczby epok uczenia klasteryzacja danych dała niezadowalający wynik. Jednak już dla 100 epok klasteryzacja danych przebiegła poprawnie i zwiększenie liczby epok dało ten sam wynik.

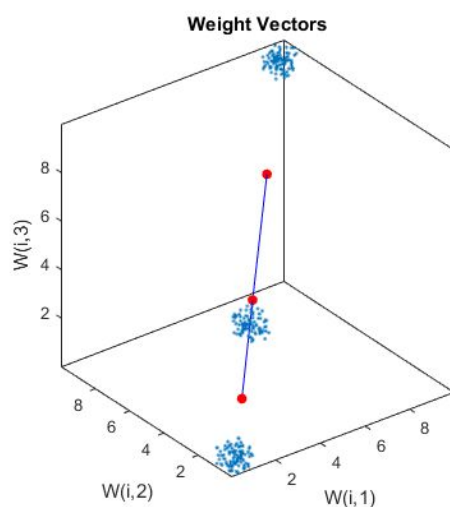
Liczba epok także wpływa na położenie neuronów na wykresie. Im większa jest liczba epok, tym neurony są lepiej umiejscowione. Znajdują się bliżej zbiorów danych.

- Wymiarowość sieci

Badając wpływ wymiarowości sieci na klasteryzację zmieniliśmy ilość warstw sieci dla obu etapów klasteryzacji.

Liczba neuronów w pierwszym etapie klasteryzacji [9 9 1]

Liczba neuronów w pierwszym etapie klasteryzacji [1 3 1]



Wyniki klasteryzacji:

klasa 1	klasa 2	klasa 3
0,3333	0,3333	0,3333

Klasteryzacja dla zmienionej liczby warstw dała taki sam wynik jak klasteryzacja dla sieci dwuwarstwowej.

b) rzeczywisty zbiór danych.

Kolejnym zadaniem była klasteryzacja rzeczywistego zbioru danych.

Jako parametry klasteryzacji przyjęto:

Liczba neuronów w pierwszym etapie klasteryzacji: 9

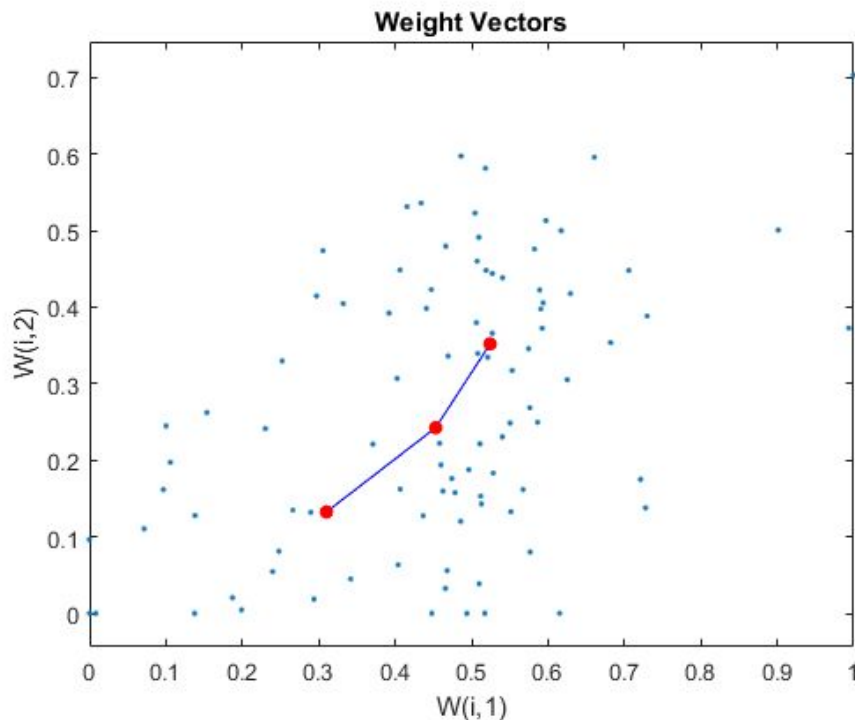
Liczba neuronów w drugim etapie klasteryzacji: 3

Topologia: *randtop*

Funkcja odległości: *boxdist*

Liczba epok: 100

Wartości próbek zostały poddane normalizacji.



Wynik klasteryzacji = 0,5161

Przynależność próbek do danych klas określono z 51.61 procentową pewnością.

3. Wnioski.

Rodzaj topologii wpływa na jakość klasteryzacji, najlepszy wynik otrzymano dla topologii randtop. Funkcja odległości jak i liczba warstw sieci nie mają wpływu na wynik klasteryzacji. Liczba neuronów gdy liczba klastrów i liczba zbiorów danych jest taka sama także nie wpływają na wynik przypisania próbek do klas. Mała liczba epok uczenia daje wyniki niezadowalające, jednak wraz ze zwiększaniem liczby epok wynik klasteryzacji poprawia się.

4. Program.

```
clear all
clc
d1= rand(3, 100);
d2= rand(3, 100) +3;
d3= rand (3,100) +9;

%load ('Data_microarray.mat');
%Data=Data';
%d= Data(1:2,:);
```

```

%for i = 1:2;

%vector = d(i,:);
%range = max(vector) - min(vector);

%for j = 1:length(vector);
%vector(j) = (vector(j)-min(vector))/range;
%range2 =1;

%vector(j) = vector(j)*range2;
%end
%d(i,:) = vector
%end
d= [d1 d2 d3];
plot3 (d(1,:), d(2,:),d(3,:),'', 'MarkerSize', 5);
hold on
box on

top='randtop'; %'hextop'; %'gridtop'; 'randtop';
odl= 'boxdist'; % 'linkdist'; 'mandist'; 'boxdist';

%net=newsom(minmax(d), [3 1]);
net= selforgmap([1 9 1], 100, 3, top, odl);
net.trainParam.epochs= 100;
net= train(net,d);

%plot3(net.IW{1,1})
%gcf
%plotsom(net.IW {1,1},net.layers{1}.distances);
%hold off

X= net.IW{1,1};
X = X';

net= selforgmap([1 3 1], 100, 3, top, odl);
net.trainParam.epochs= 100;
net= train(net,X);

gcf
plotsom(net.IW {1,1},net.layers{1}.distances);
hold off

A= sim(net, d);
A= vec2ind(A);

j = 0;
k= 0;

```

```
l= 0;
```

```
for i=1:length(A)
```

```
    if A(i) ==1;
```

```
        j = j + 1;
```

```
    end
```

```
    if A(i) ==2;
```

```
        k = k + 1;
```

```
    end
```

```
    if A(i) == 3;
```

```
        l = l + 1;
```

```
    end
```

```
end
```

```
result1 = j/length(A)
```

```
result2 = k/length(A)
```

```
result3 = l/length(A)
```