

Maciej Krzywda,
Inżynieria Obliczeniowa, IMiIP
Podstawy Sztucznej Inteligencji
nr albumu: 293102

Sprawozdanie 6

Tytuł projektu:

Budowa i działanie sieci Kohonena dla WTM

- **Cel projektu:**

Celem ćwiczenia jest poznanie budowy i działania sieci Kohonena przy wykorzystaniu reguły WTM do odwzorowywania istotnych cech liter alfabetu.

- **Przebieg ćwiczenia**

1. Wygenerowanie danych uczących i testujących, zawierających 20 dużych liter dowolnie wybranego alfabetu w postaci dwuwymiarowej tablicy np. 4x5 pikseli dla jednej litery.
przykładowe znaki: <http://www.ai.c-labtech.net/sn/litery.html>
2. Przygotowanie (implementacja lub wykorzystanie gotowych narzędzi) sieci Kohonena i algorytmu uczenia opartego o regułę Winner Takes Most (WTM).
3. Uczenie sieci dla różnych współczynników uczenia.
4. Testowanie sieci.

- **Część Teoretyczna**

Sieć neuronowa – Nazwa struktur matematycznych i ich programowych lub sprzętowych modeli. Wykonuje obliczenia za pomocą powiązanych ze sobą elementów (neuronów). Sieć neuronowa została zaprojektowana w taki sposób by przypominać działanie ludzkiego mózgu złożonego z naturalnych neuronów i łączących je synaps.

Sieć jednokierunkowa – Jest to sieć neuronowa, w której nie występuje zjawisko sprzężenia zwrotnego oznacza to, iż każdy sygnał przechodzi przez każdy neuron dokładnie raz w swoim cyklu.

Sieć Kohonena – Jest podstawowym typem sieci samoorganizującym się. Metodyka nauki sieci samoorganizującej różni się tym, iż nie podajemy jej wzorca do jakiego dążymy, a suche dane, które musi interpretować sama. Uczenia polega na wybieraniu w drodze konkurencji zwycięskiego neuronu, który najlepiej odpowiada wektorowi wejściowemu. System ten w zbliżeniu bardzo przypomina uczenie się przez ludzi.

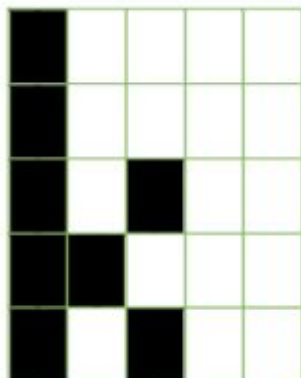
Etapy działania takiej sieci to.:

- Konstrukcja
- Uczenie
- Rozpoznawanie

Reguła WTM (Winner Takes Most) - Opiera się na takiej samej zasadzie rywalizacji jak WTA czyli neurony konkurują ze sobą, gdzie neuron o najlepszej odpowiedzi na dane wejściowe zostaje zwycięzcą, jednak w tym przypadku (WTM) wprowadza się sąsiedztwo zmieniając tym samym zasadę WTA w „zwycięzca bierze większość”.

• Przebieg ćwiczenia

Przebieg ćwiczenia.: - Wygenerowanie danych wejściowych, w tym przypadku 20 pierwszych liter alfabetu z pominięciem litery 'M', 'V', 'W', oraz 'X' z uwagi, na trudność rozpoznania dla danej matrycy - Przyjęta została matryca (tablica) 4x5, w której odpowiednie pola oznaczone 1 odpowiadały kolorowi czarnemu, natomiast zerem dla koloru białego, przykładowo.:

	<table><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td></tr></table>	1	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	1	0	0	0	1	0	1	0	0
1	0	0	0	0																						
1	0	0	0	0																						
1	0	1	0	0																						
1	1	0	0	0																						
1	0	1	0	0																						

Matryce wykorzystałem już wcześniej w projekcie pt. **PSI. Budowa i działanie sieci jednowarstwowej**

Wszystkie dane wejściowe zostały przedstawione w tabeli „iData”, w której każdy wektor (litera) zajmował jedną kolumnę, stąd dane wyjściowe to tablica (4x5 x 20 dużych liter)

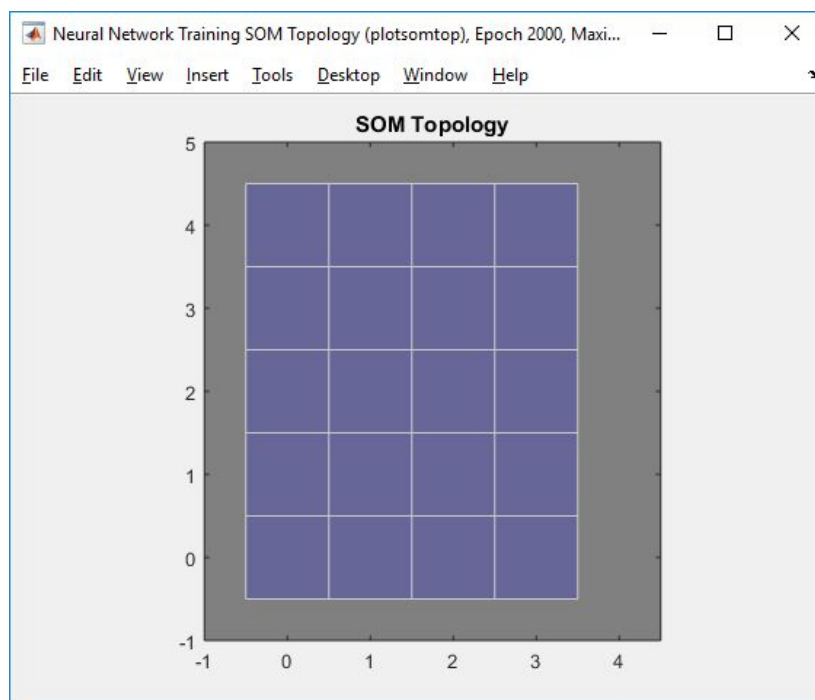
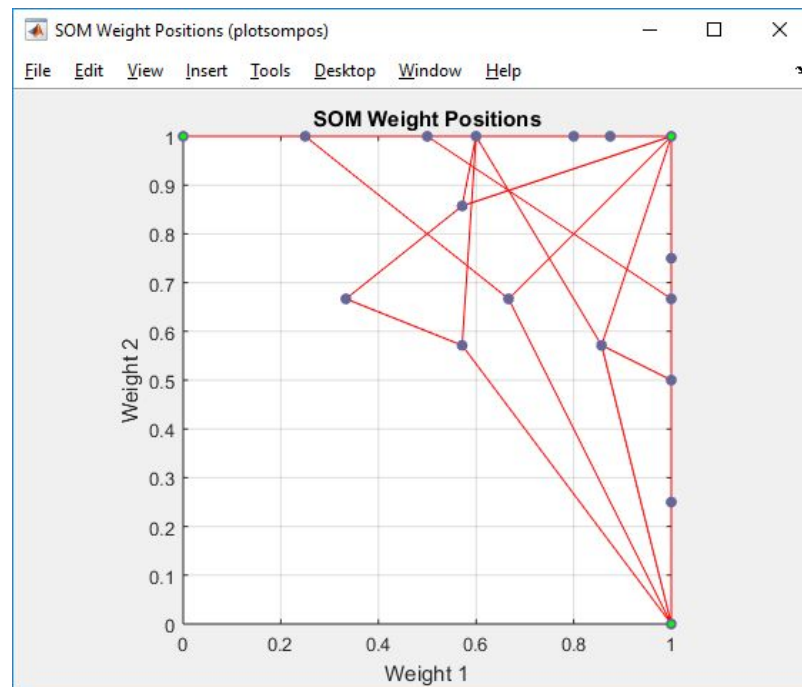
- Do przeprowadzenia ćwiczenia posłużyłem się funkcją Kohonena przy wykorzystaniu reguły WTM opisanej powyżej
- Opis parametrów jak i szczegółowy opis funkcji zostały zawarte w listingu programu w komentarzach
- Sieć została stworzona z domyślnymi wymiarami 4 x 5 (wielkość jednej litery) w kształcie sześciokątów, natomiast początkowy rozmiar sąsiedztwa ustawiony jest na 1, liczba epok = 2000

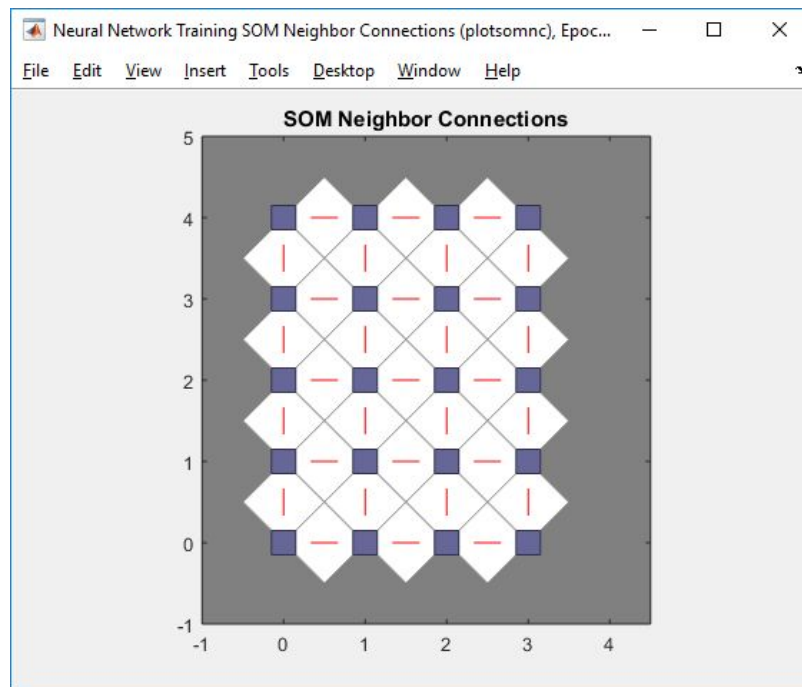
- Listing programu

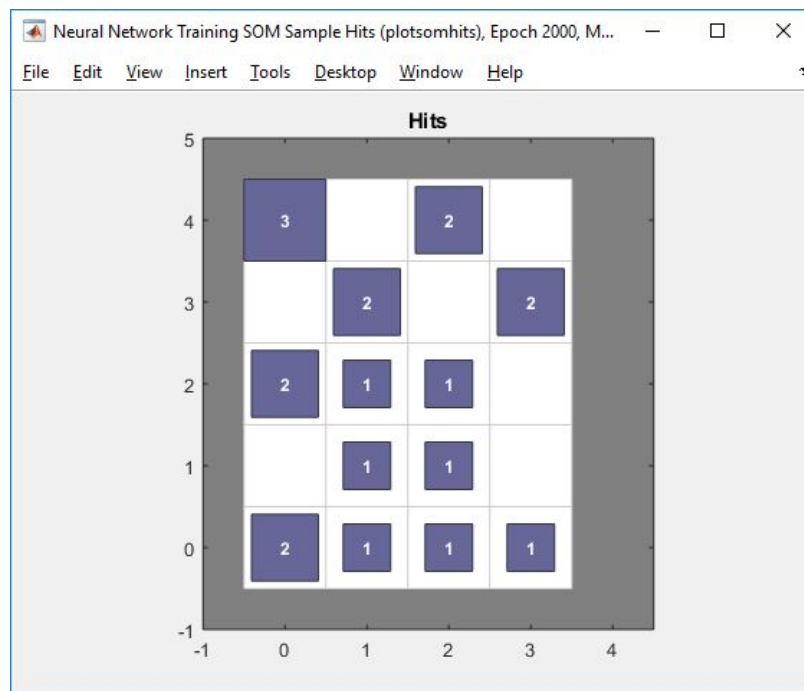
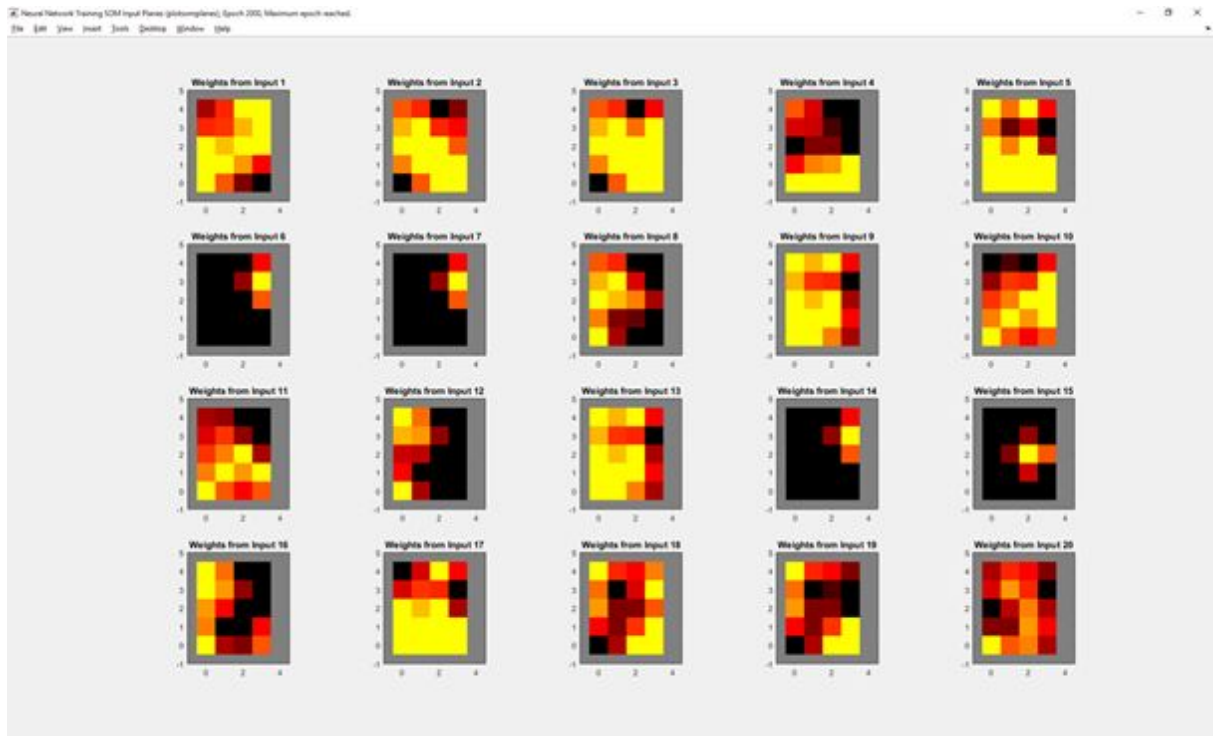
```
close all; clear all; clc;
%A B C D E F G H I J K L N O P R S T U Y
iData= [0 1 0 1 1 1 0 1 1 1 1 1 1 0 1 1 0 1 1 1;
        1 1 1 1 1 1 1 0 0 1 0 0 0 1 1 1 1 1 0 0;
        1 1 1 1 1 1 1 0 0 1 0 0 0 1 1 1 1 1 0 1;
        0 0 1 0 1 1 1 1 0 1 1 0 1 0 0 0 1 0 1 0;
        1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1;
        0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0;
        0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1;
        1 1 0 1 0 0 0 1 0 1 0 0 1 1 1 1 0 0 1 0;
        1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 0 1 0;
        1 1 0 0 1 1 0 1 0 0 1 0 0 0 1 1 1 1 0 1;
        1 1 0 0 1 1 1 1 0 0 0 0 1 0 1 1 1 0 0 0;
        1 0 0 1 0 0 1 1 0 1 0 0 1 1 0 0 0 0 1 0;
        1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 0 1 0;
        0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1;
        0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0;
        1 1 0 1 0 0 1 1 0 1 0 0 1 1 0 0 1 0 1 0;
        1 1 1 1 1 1 0 1 1 0 1 1 1 0 1 1 1 0 0 0;
        0 1 1 1 1 0 1 0 0 0 0 1 0 1 0 0 1 1 1 1;
        0 1 1 1 1 0 1 0 0 0 0 1 0 1 0 0 1 0 1 0;
        1 0 1 0 1 0 1 1 0 1 1 1 1 0 0 1 0 0 0 0;
    ];
dimensions = [4 5];
coverSteps = 100;
initNeighbor = 1;
topologyFcn = 'gridtop';
%topologyFcn = 'hextop';
distanceFcn = 'dist';
net = selforgmap(dimensions,coverSteps,initNeighbor,topologyFcn,distanceFcn);
net.trainFcn = 'trainbu';
net.trainParam.epochs = 2000;
[net,tr] = train(net,iData);
y = net(iData);
plotsompos(net,iData);
grid on
```

- Analiza Wyników

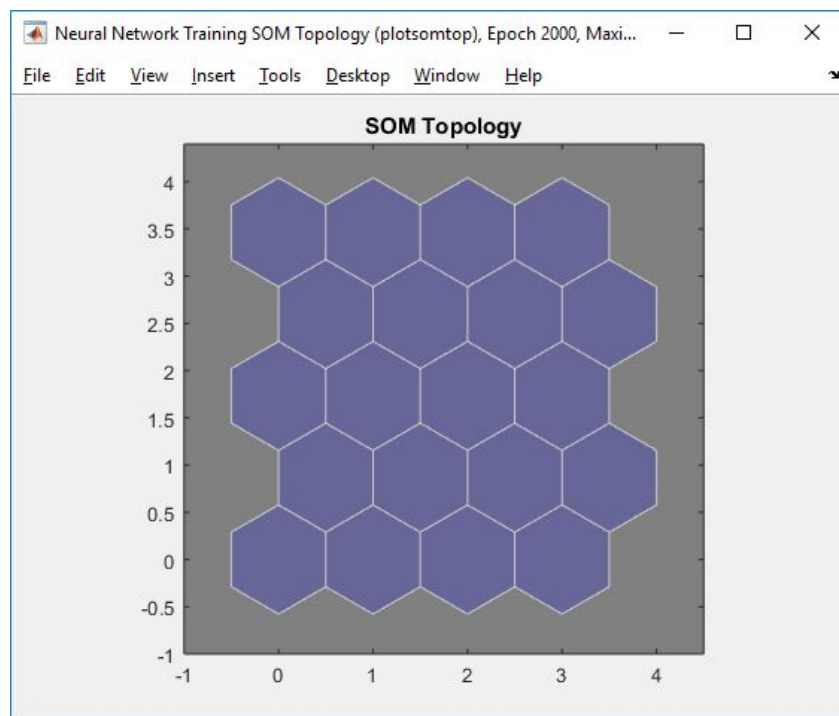
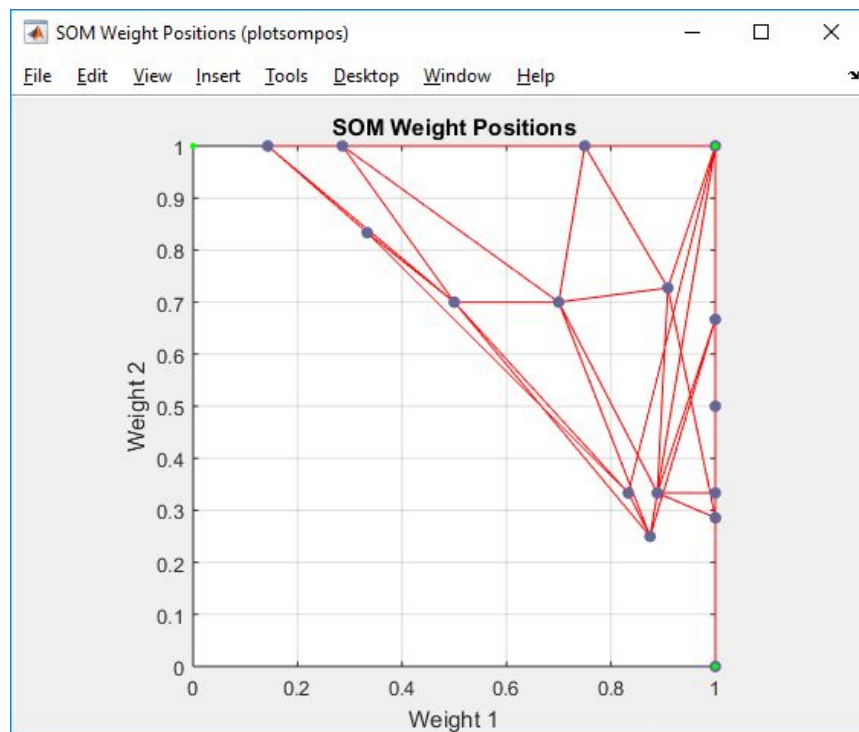
GridTop:

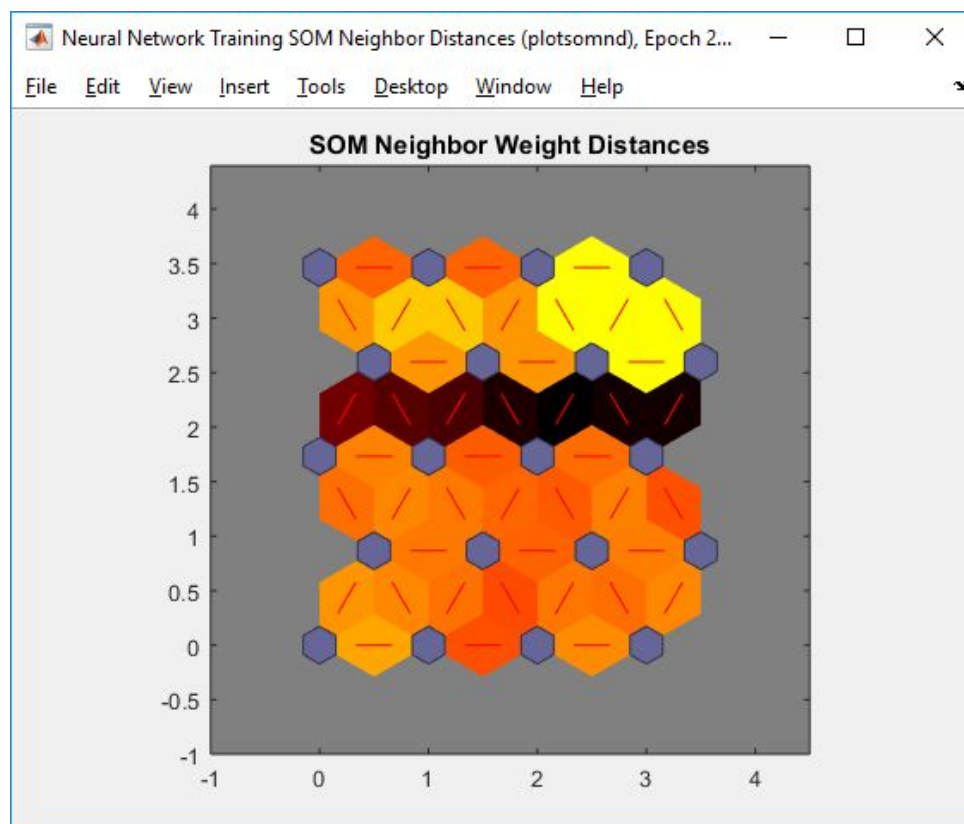
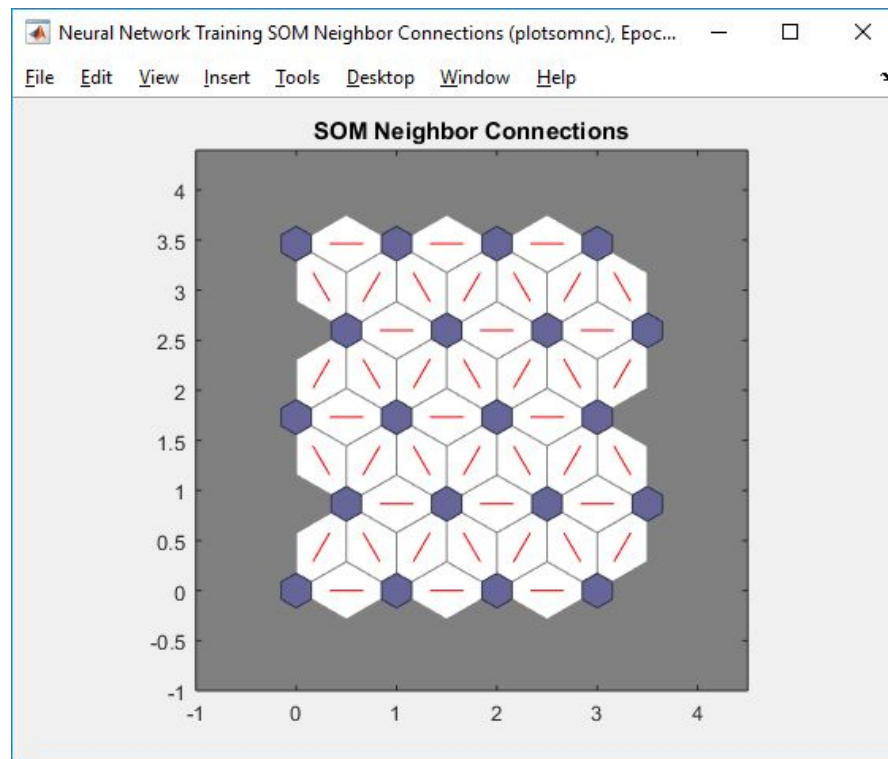


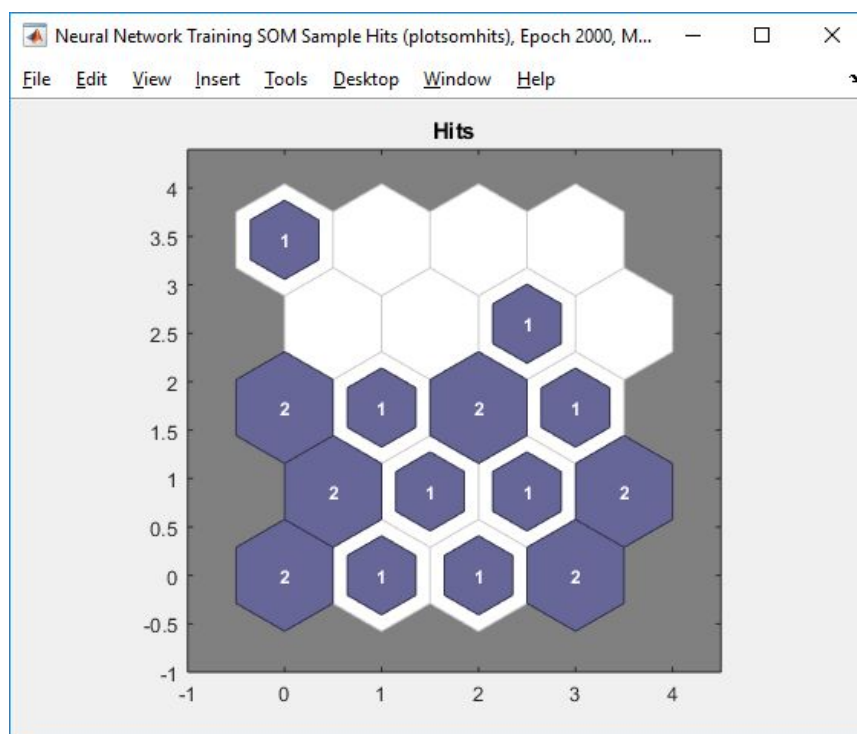
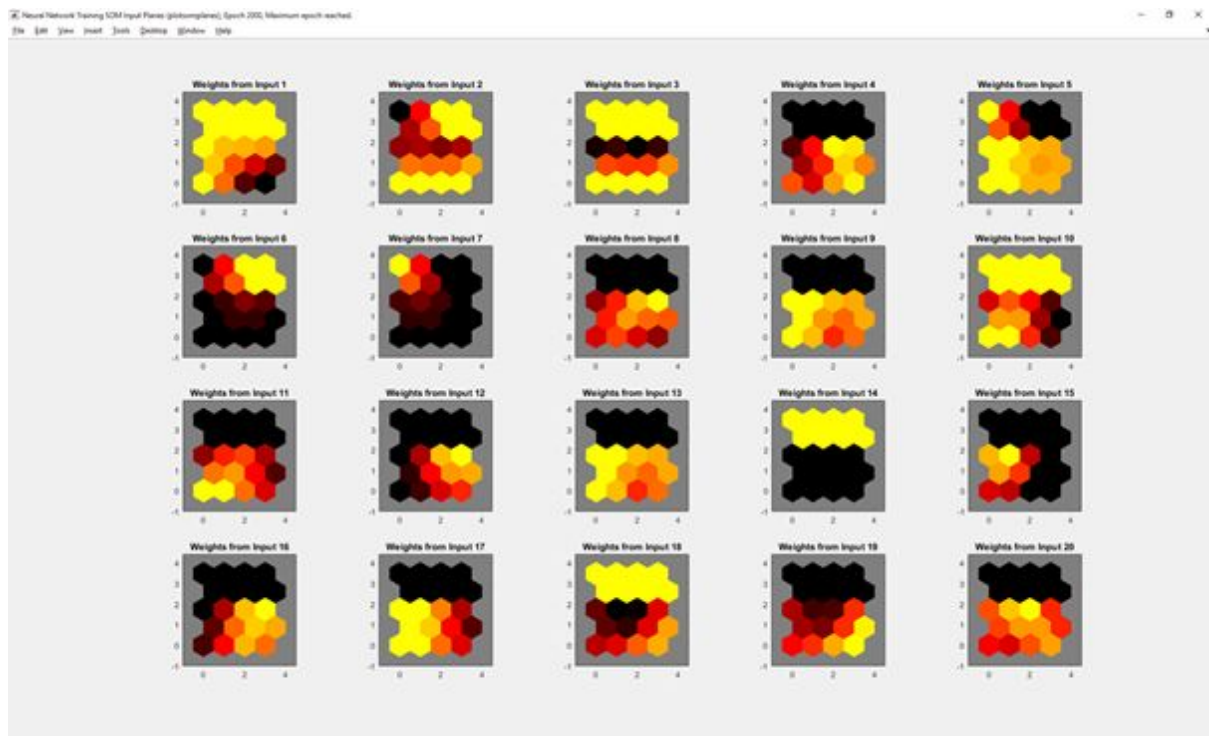




Hextop:







W wyniku analizy otrzymanych wyników (łącznie z tymi zawartymi w załącznikach) możemy zdecydowanie stwierdzić iż algorytm oparty na sieci Kohonena z wykorzystaniem WTM (winner takes most) może być dokładniejszy i bardziej efektywny od stosowanego we wcześniejszym projekcie WTA, gdyż wagi nie są brane dla całości programu poprzez wyselekcjonowanie najmocniejszej, lecz poprzez uśrednienie za pomocą sąsiedztwa. (rozkład zwycięstw jest bardziej równomierny, aniżeli skupiony w jednym miejscu wokół jednego „zwycięzcy” jak to ma miejsce dla WTA).

Mamy również możliwość porównania rozkładu jaki zapewnia nam wersja topologii Hexagonalna względem GridTop, zwłaszcza gdy sieć nie jest równych wymiarów jak w tym przypadku (4x5), należy również pamiętać o tym, iż dla stabilnego działania programu zwiększając wielkość sąsiedztwa powinno się zachowywać odpowiednie proporcje wielkości sieci. Oczywiście wraz ze zwiększeniem rozmiaru sieci należy zwiększyć liczbę epok potrzebnych na rozwiązanie problemu.

● Wnioski

W zależności od współczynnika uczenia pewne podziały na klastry i tendencje stają się silniejsze.

Rozkłady sił koncentrują się wzdłuż brzegów siatki topologii w zależności od współczynnika uczenia – im jest on wyższy, tym siły bardziej skupiają się wzdłuż brzegów siatki i tylko w tych miejscach.

W zależności od współczynnika uczenia zmienia się rozkład wag i kształt ich powiązań, ale co ciekawe jest niska ilość neuronów martwych (niepowiązanych, niepodobnych do siebie), co zdecydowanie odróżnia system WTM od WTA.

W metodyce WTM sąsiedztwo gra niebagatelną rolę i determinuje ono kształt korelacji i zależności – im jest ono wyższe tym podziałów jest mniej, ale bardziej porozrzucane po całej siatce i poszczególne neurony znajdują się w innych kategoriach.

Rozkłady sił stają się równomierne, im sąsiedztwo jest wyższe. Również rozkład sił na siatce koncentruje się w poziomym kierunku, równolegle bądź współliniowo do dolnego brzegu siatki Kohonena.

Im wyższe sąsiedztwo, tym obiekty stają się bardziej do siebie podobne w pewnych cechach i zanika ilość podziałów.

W metodologii WTM obiekty są bardziej ze sobą powiązane, mimo zmian i w sąsiedztwie i we wsp. uczenia. Jedyne co się zmienia to ilość wag i kształt tych powiązań.

Metodologia ta powinna być stosowana wszędzie tam, gdzie istnieje potrzeba znalezienia jakichkolwiek powiązań z daną cechą.