

Daniel Wysowski  
Nr. albumu: 286136  
Inżynieria Obliczeniowa  
Wydział Inżynierii Metali i Informatyki Przemysłowej  
Akademia Górniczo-Hutnicza w Krakowie



# **Sprawozdanie z przedmiotu: Podstawy sztucznej inteligencji**

## **Scenariusz 6 - Budowa i działanie sieci Kohonena dla WTA. cz.2**

# **Spis treści:**

- I. Wstęp teoretyczny
- II. Wykonanie zadania + opis działania
- III. Podsumowanie

# I. WSTĘP TEORETYCZNY

Zasada WTM: Najprościej mówiąc zasada WTM realizuje sieć, w której zwycięzca bierze większość. Neuron zwycięski oraz neurony sąsiadujące z neuronem zwycięskim, czyli należące do sąsiedztwa  $N_c(k)$  aktualizują swoje wagi według zasady:

$$w_i(k+1) = w_i(k) + \eta(k)G(i,c)[x(k) - w_i(k)], \quad i \in N_c(k)$$

$G(i,c)$  - wpływ sąsiedztwa.

$$G(i,x) = \begin{cases} 1 & \text{dla } d(i,w) \leq \lambda \\ 0 & \text{dla } d(i,w) > \lambda \end{cases}$$

**Sieci Kohonena:** nazywane samoorganizującymi się mapami (ang. SOM - SelfOrganized Maps), składają się z jednej warstwy neuronów, które tworzą mapę z liczbą wyjść równą liczbie neuronów. Konkurencyjne uczenie w sieciach tego typu polega na prezentowaniu sieci wzorców uczących  $X$ . Wagi poszczególnych neuronów adaptują się w taki sposób, że neurony te stają się reprezentantami poszczególnych klas sygnałów wejściowych.

**SOM:** Sieć samoorganizująca się. Poprzez samouczenie się sieci samoorganizujące się pozwalają na redukcję wymiaru analizowanych sieci.

## II. WYKONANIE ZADANIA I OPIS DZIAŁANIA

### CEL ĆWICZENIA:

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

### WEJŚCIE PROGRAMU:

Litery wybrane przeze mnie: **A C D E F G H K M N O P R S T U W X Y Z**, litery są utworzone na matrycy 5 x 7 czyli na 35 polach. Dane uczące, przedstawiane są za pomocą takich table jak ta z poniższego przykładu, przygotowana dla litery A:

```
letterA= [0 0 1 0 0 ...  
          0 1 0 1 0 ...  
          0 1 0 1 0 ...  
          1 0 0 0 1 ...  
          1 1 1 1 1 ...  
          1 0 0 0 1 ...  
          1 0 0 0 1 ]';
```

Identyczne tabele dla każdej z liter są dla nas wejściem programu.

dimensions- wektor rzędów wymiarów (domyślnie = [8 8])

coverSteps- liczba kroków szkoleniowych dla początkowego pokrycia przestrzeni wejściowej (domyślnie = 100)

initNeighbor - początkowy rozmiar sąsiedztwa (domyślnie = 3) Sąsiedztwo może być sztywne (0 - nie należy, 1 - należy) lub też wyznaczane funkcją przynależności przyjmującej wartości rzeczywiste z przedziału [0,1].

topology

FCN - funkcja topologii warstw (domyślnie= 'hextop') distanceFCN - funkcja odległości neuronowej (domyślnie = 'linkdist')

### Listing kodu:

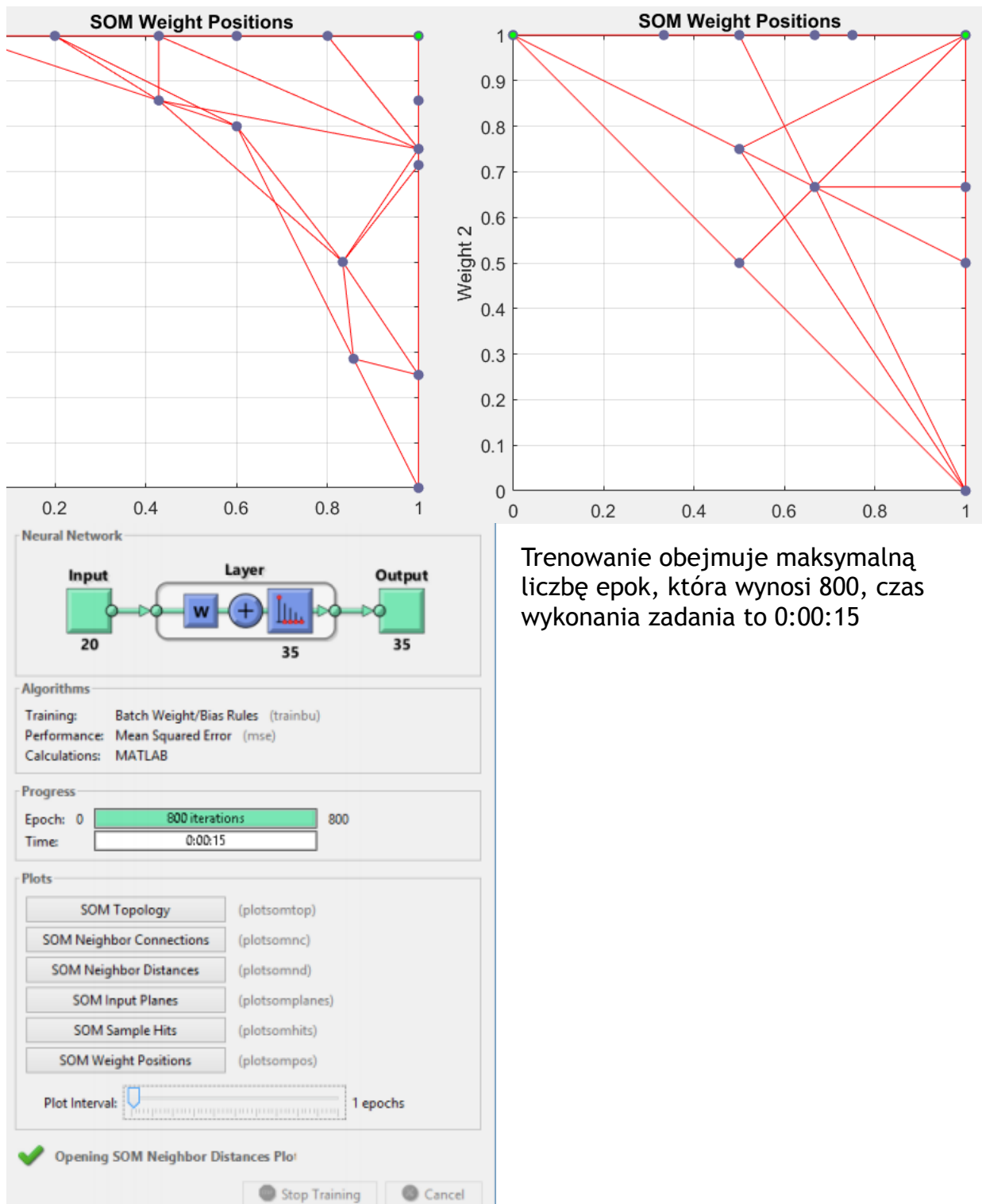
```
A_='A'; B_='B'; C_='C'; D_='D'; ... ; S_='S'; T_='T';  
input=[ A_(:) B_(:) C_(:) D_(:) E_(:) F_(:) G_(:) H_(:) I_(:) J_(:) K_(:) L_(:) M_(:) N_(:) O_(:) P_(:) Q_(:) R_(:) S_(:) T_(:)];  
dimensions = [10 10];  
coverSteps = 100;  
initNeighbor = 1;  
topologyFcn = 'hextop';
```

```

distanceFcn = 'linkdist';
net =
selforgmap(dimensions,coverSteps,initNeighbor,topologyFcn,distanceFcn);
net.trainParam.epochs = 200;
net.trainFcn = 'trainbu';
net = train(net,input);
y = net(input);
classes = vec2ind(y);
plotsompos(net,input);
grid on

```

## WYJŚCIE PROGRAMU:



## Podsumowanie:

Sieci Kohonena wykazują pod względem budowy i zasad funkcjonowania wysokie podobieństwo do ludzkiego mózgu. Podobieństwo to umożliwia efektywne zastosowanie sieci SOM do opracowania i klasyfikacji danych pochodzących z analiz praktycznie dowolnego obiektu badań. Umiejętność rozpoznawania liter alfabetu występujących w danych i możliwości modyfikowania wewnętrznej kompozycji wag neuronów czyni z samoorganizujących się map Kohonena potężne narzędzie współczesnej analizy danych.

W kolejnych iteracjach po znalezieniu neuronu zwycięzcy, wyznaczani są jego sąsiedzi. Co krok uaktualniane są wagi węzłów promienia otaczającego neuron zwycięzcy, który jest zdefiniowany przez nas.