Gabriela Ciołek

Nr indeksu: 293083

Inżynieria Obliczeniowa

Wydział Inżynierii Metali i Informatyki Przemysłowej

Akademia Górniczo-Hutnicza w Krakowie

**Sprawozdanie 5**

Przedmiot: Podstawy sztucznej inteligencji

Temat: Budowa i działanie sieci Kohonena dla WTA

**CEL:**

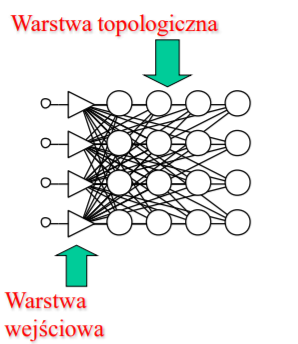
Celem ćwiczenia jest poznanie budowy i działania sieci Kohonena przy wykorzystaniu reguły WTA do odwzorowywania istotnych cech kwiatów.

**TEORIA:**

*SIECI KOHONENA*

Sieci Kohonena są jednym z podstawowych typów sieci samoorganizujących się. Stanowią one synonim całej grupy sieci, w których uczenie odbywa się metodą samoorganizującą typu konkurencyjnego. Polega ona na podawaniu na wejścia sieci sygnałów, a następnie wybraniu w drodze konkurencji zwycięskiego neuronu, który najlepiej odpowiada wektorowi wejściowemu. Dokładny schemat konkurencji i późniejszej modyfikacji wag synaptycznych może mieć różną postać. Wyróżnia się wiele podtypów sieci opartych na konkurencji, które różnią się dokładnym algorytmem samoorganizacji.

STRUKTURA SIECI KOHONENA



Cechy charakterystyczne:

• sieć uczy się bez nauczyciela,

• uporządkowane neurony wyjściowe,

• jest konkurencja i wyłaniany jest neuron „zwycięski”,

• ważną rolę odgrywa „sąsiedztwo”,

• w wyniku uczenia powstaje mapa topologiczna,

• po uczeniu można ustalić, jakie znaczenie mają poszczególne rejony mapy topologicznej - ale wyłącznie na podstawie analizy konkretnych przykładów danych wejściowych.

*ALGORYTM WTA*

Algorytm WTA polega na obliczaniu aktywacji każdego neuronu, a następnie wyborze zwycięzcy o największym sygnale wyjściowym. W konkurencyjnej metodzie uczenia sieci, tylko jeden element wyjściowy może znajdować się w stanie aktywnym. Nazywany jest on zwycięzcą, a schemat takiej reguły aktywacji neuronów określany jest mianem zwycięzca bierze wszystko (ang. Winner Takes All - WTA). Neuron, który wygra, jeszcze bardziej jest upodobniany do przykładu, którym uczymy.

**LISTING KODU WRAZ Z KOMENTARZAMI:**

close all; clear all; clc;

%Tablica input[] zawiera wartości identyfikujące rodzaj kwiatu.

%Kolumna I-długosc działki kielicha; Kolumna II-szerokosc działki kielicha

%Kolumna III-długosc płatka kwiatu; Kolumna IV-szerokosc płatka kwiatu

input=iris\_dataset;

plot(input(1, :) ,input(2, :), 'k.', input(3, :), input(4, :), 'b.');

legend('działki kielicha', 'płatki kwiatu');

hold on;

grid on;

dimensions = [10 10]; % wymiary wektora

coverstep = 50; %etapy szkolenia

initNeighbor = 0; % wejściowy rozmiar sąsiedztwa

topologyFcn = 'hextop'; %dane przyjmuja ksztalt szescianow

distanceFcn = 'dist'; %funkcja dystansu neuronów

%STWORZENIE SIECI SAMOORGANIZUJACEJ

net = selforgmap(dimensions, coverstep, initNeighbor, topologyFcn, distanceFcn);

net.trainParam.epochs = 500;

net.trainParam.lr = 0.1; %współczynnik uczenia

%TRENING SIECI

[net, tr] = train(net, input); %trening sieci

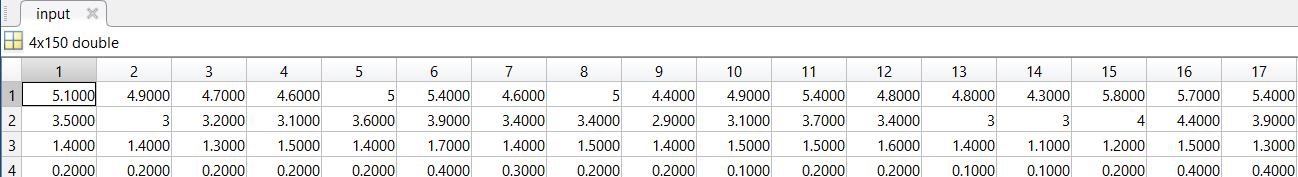
y = net(input); %testowanie i zapis wyników osiągniętych przez sieć

**OPIS WYKONANEGO ZADANIA:**

Do moich zadań należało:

1. Przygotowanie danych uczących zawierających numeryczny opis cech kwiatów.

Dane uczące zostały pozyskane ze zbioru danych już zawartych w pakiecie Matlab. Posłużyła mi do tego linijka input=iris\_dataset. Do poprawnego działania programu tablica musiała zostać transponowana.



Zawartość tablicy zawierającej numeryczny opis kwiatów

1. Przygotowanie (implementacja lub wykorzystanie gotowych narzędzi) sieci Kohonena i algorytmu uczenia opartego o regułę Winner Takes All (WTA).

Do stworzenia sieci Kohonena, która jest siecią samoorganizującą się wykorzystałam zawartą w pakiecie Matlab funkcję *selforgmap().* Przyjmuje ona następujące argumenty:

(dimensions, coverstep, initNeighbor, topologyFcn, distanceFcn)

gdzie:

dimension-wymiary wektora,

coverstep – etapy szkolenia,

initNeighbor – wejściowy rozmiar sąsiedztwa,

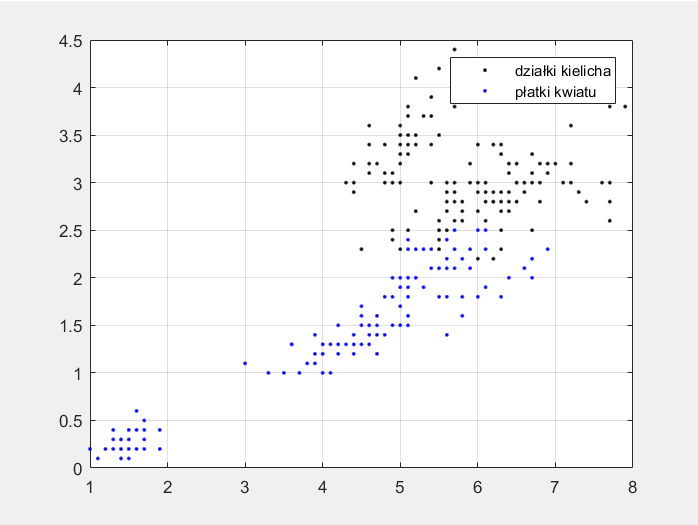
topologyFcn – kształt, który przyjmą dane,

distanceFcn – dystans neuronów

c) Uczenie sieci dla różnych współczynników uczenia.

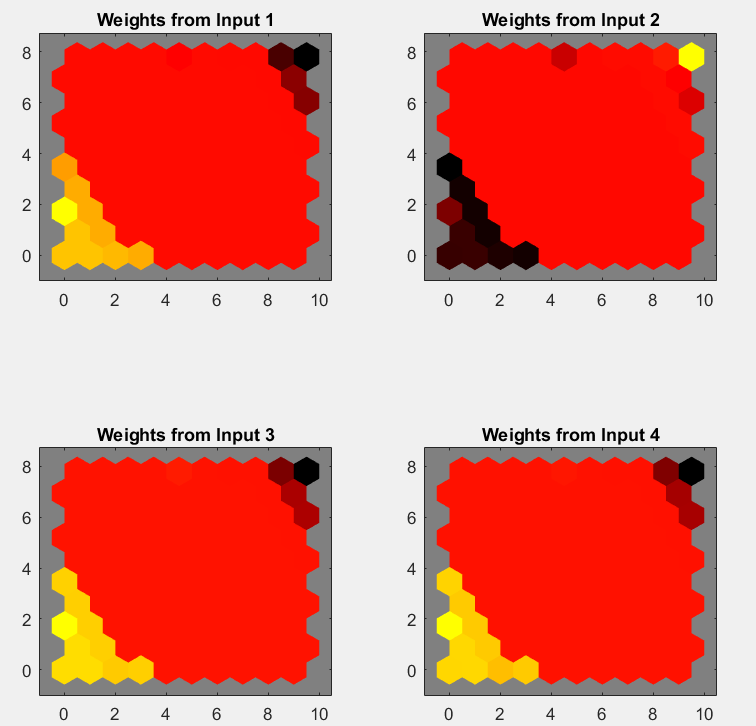
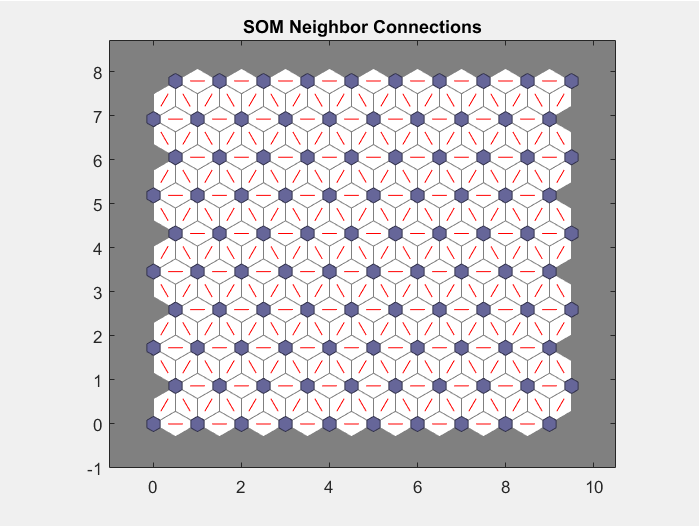
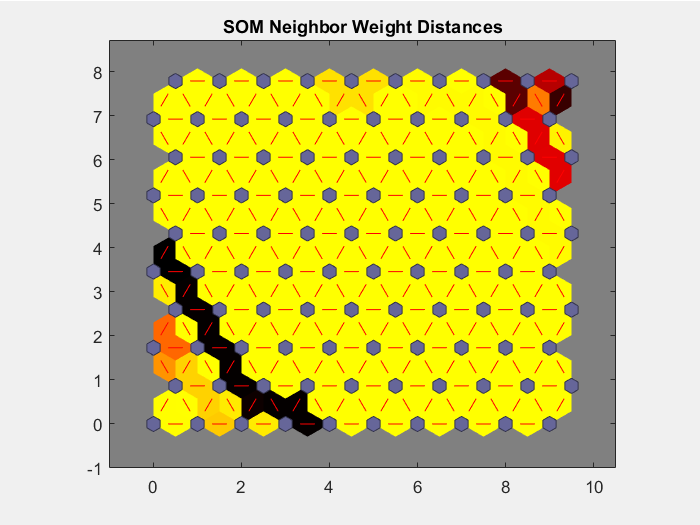
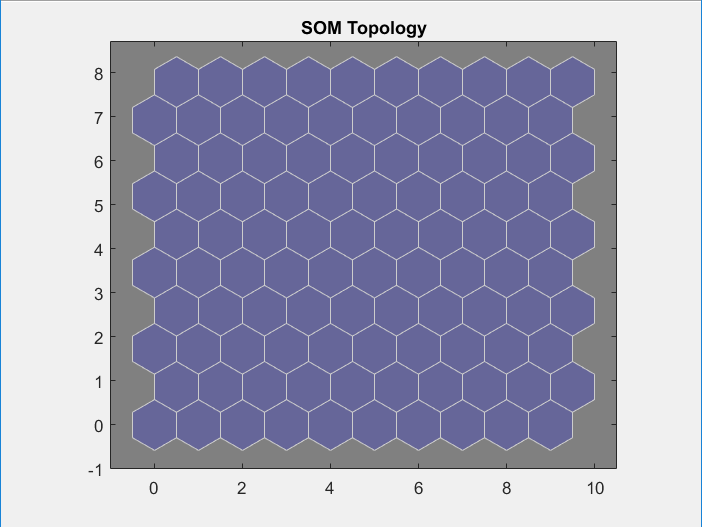
Przyjęłam 3 różne wartości współczynnika uczenia: 0.1, 0.4, 0.7. Otrzymane wykresy przedstawiam w punkcie poniżej.

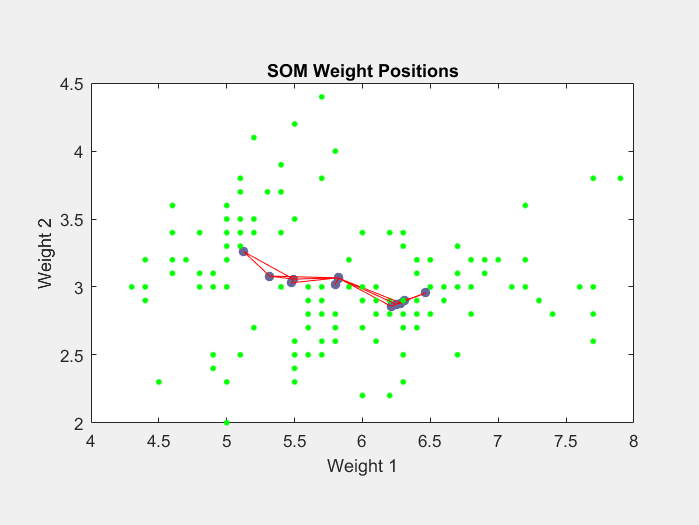
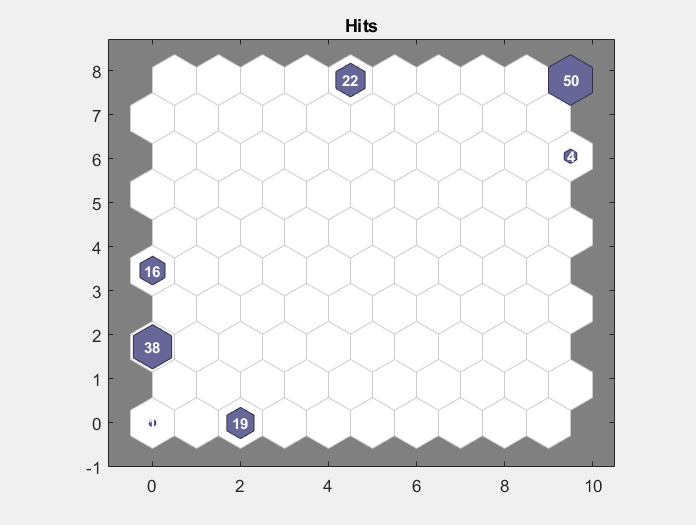
d) Testowanie sieci.



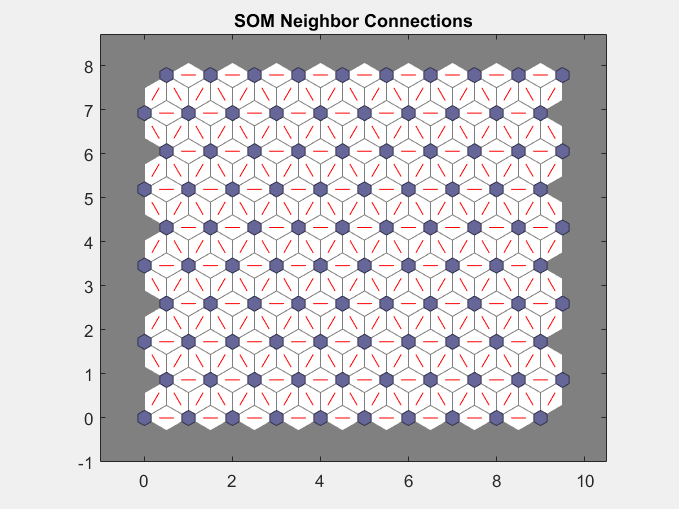
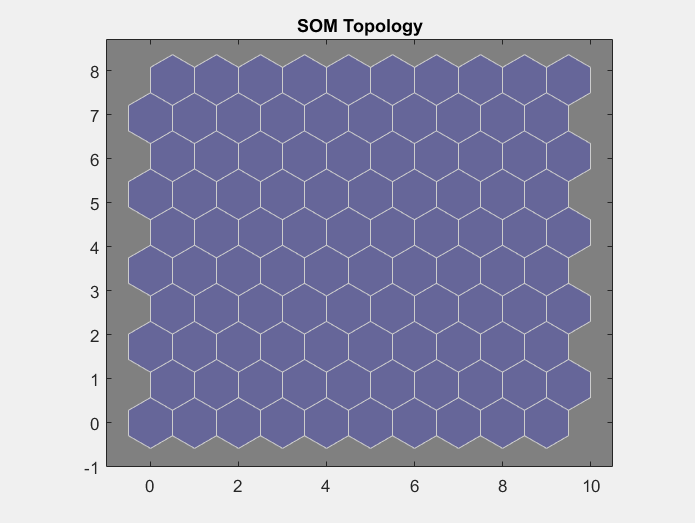
Utworzona sieć.

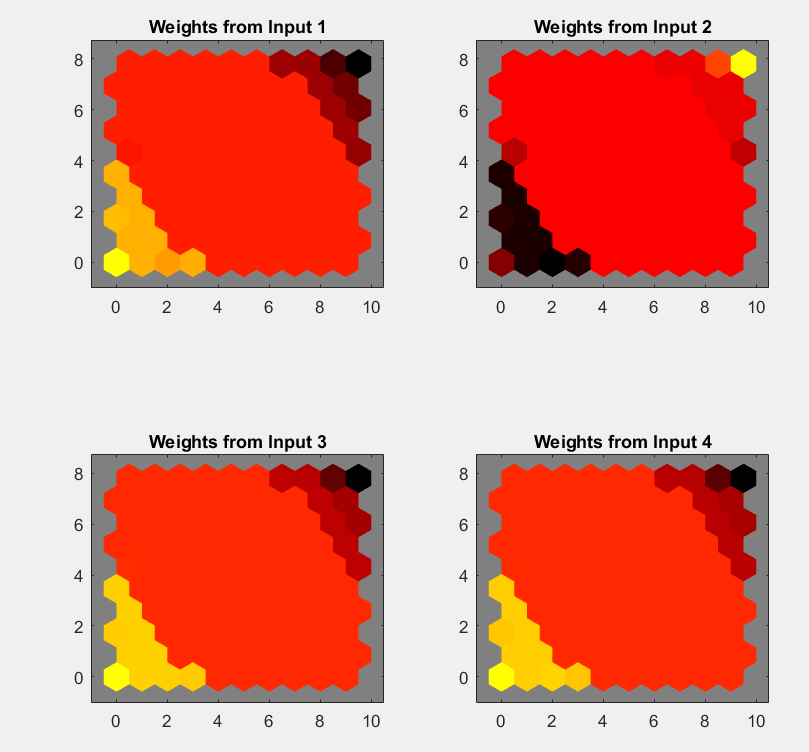
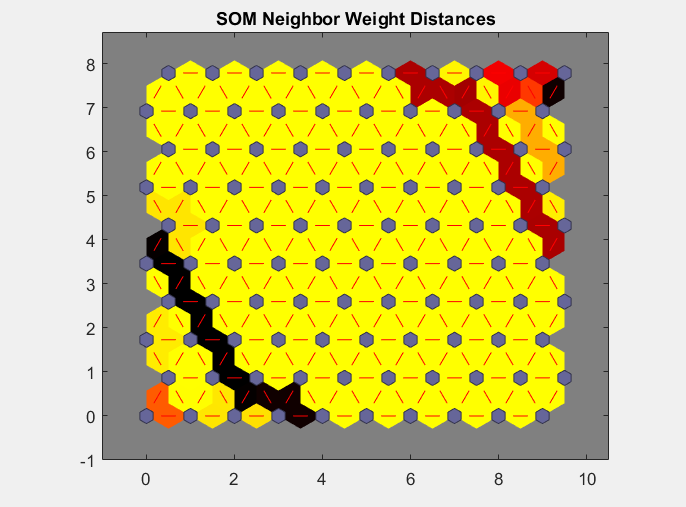
WSPÓŁCZYNNIKI UCZENIA = 0.1

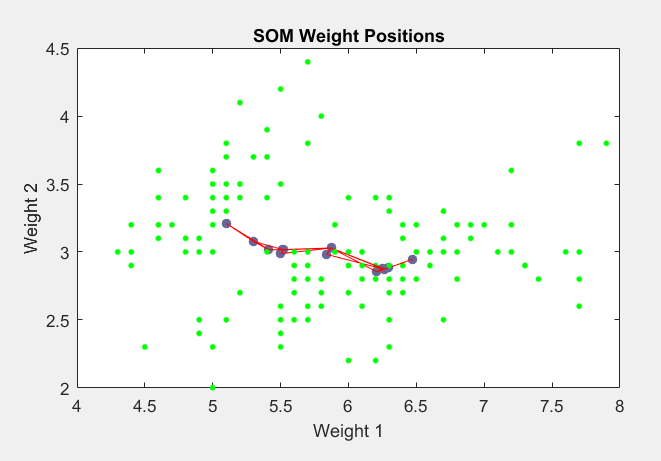
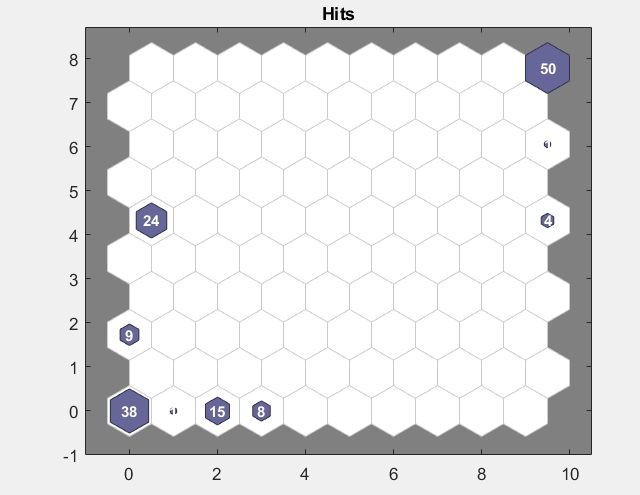




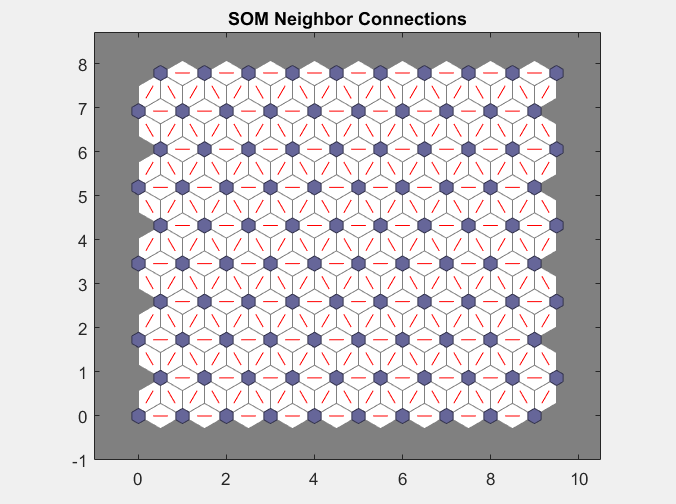
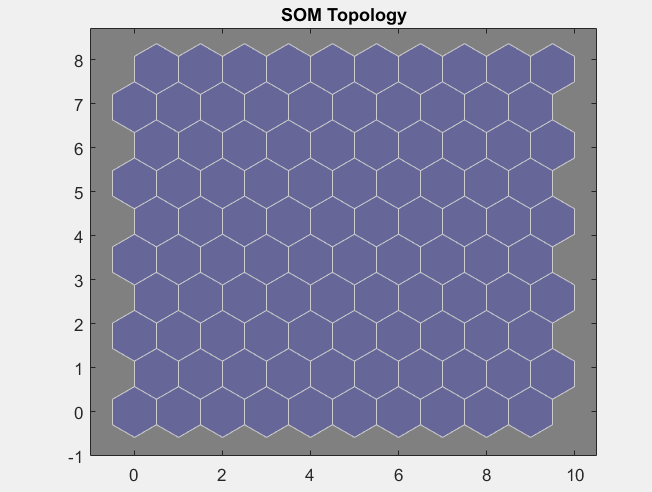
WSPÓŁCZYNNIK UCZENIA: 0.4

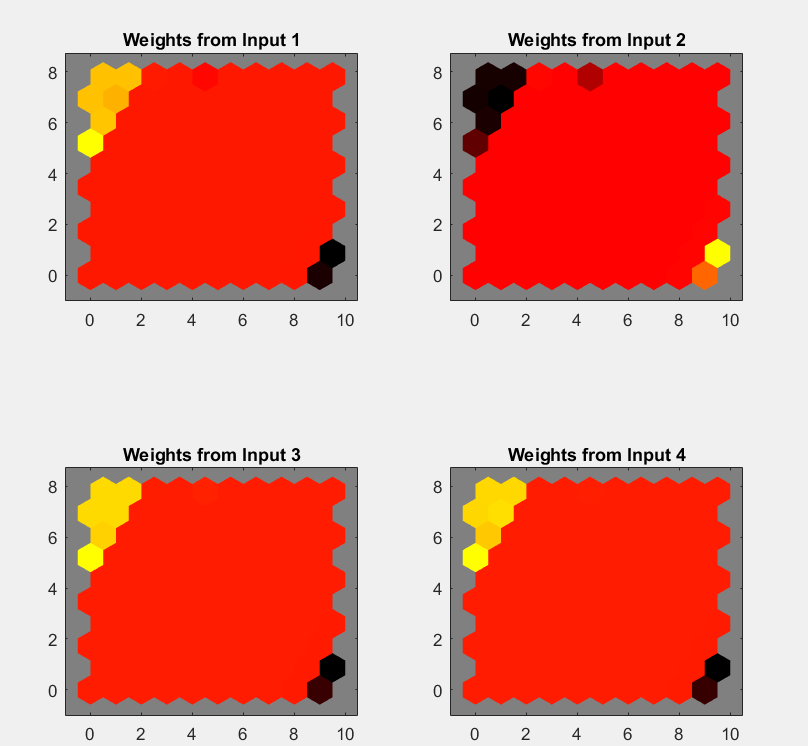
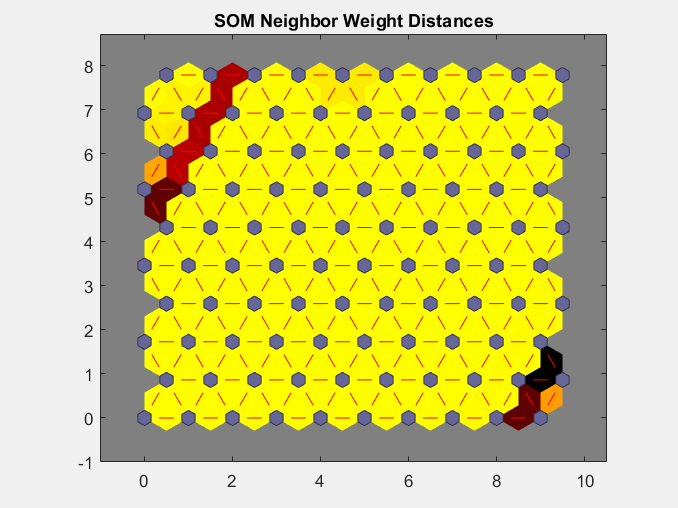


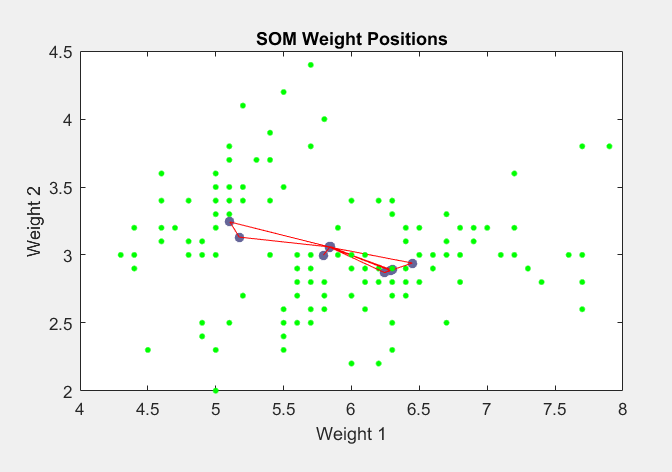
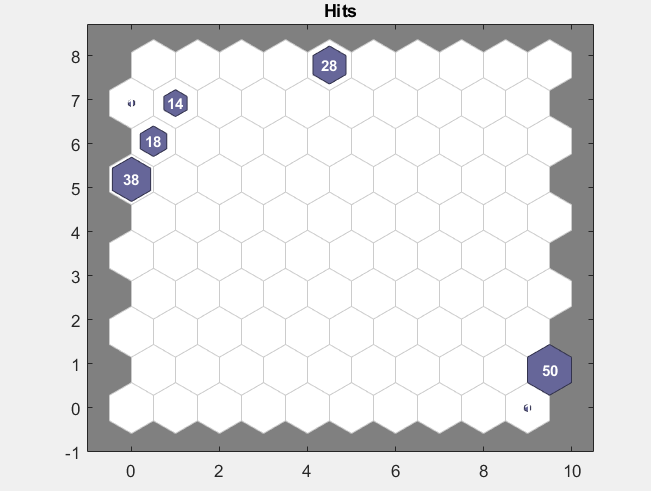




WSPÓŁCZYNNIK UCZENIA: 0.7

****

****

****

SOM Topology – topologia sieci Kohonena. Każdy z sześciokątów reprezentuje neuron. Sąsiedztwo dwóch figur może wskazywać na ich podobieństwo.

SOM Neighbor Connections - reprezentują połączenia pomiędzy poszczególnymi neuronami.

SOM Neighbor Distance – reprezentują jak silne są powiązania między poszczególnymi neuronami. Większe wagi są oznaczone ciemniejszym kolorem.

SOM Weights – wskazuje na rozkład wag poszczególnych neuronów w sieci

SOM Hits – pokazuje ile trafień znalazło się w naszym uruchomieniu programu. Czyli ile podobnych wyników zostało uzyskanych.

SOM Weight Position – zielone kropki to dane wejściowe, połączone ze sobą liniami wykazują korelację.

**WNIOSKI:**

**-** SOM pozwala na klasyfikację bardzo dużej ilości danych wejściowych, które mogą być wielowymiarowe.

- Analiza barw na danych wykresach pozwala zobrazować jak w przybliżeniu może wyglądać dany kwiat.

- Im wyższy współczynnik uczenia, tym dłużej czasu sieć potrzebowała na nauczenie się algorytmu.

- Poprawne dobranie współczynników pozwala na zoptymalizowanie działania programu.