Self-organizing map using matlab

Create a Self-Organizing Map Neural Network: selforgmap

Syntax:

selforgmap (dimensions, coverSteps, initNeighbor, topologyFcn, distanceFcn)

takes these arguments:

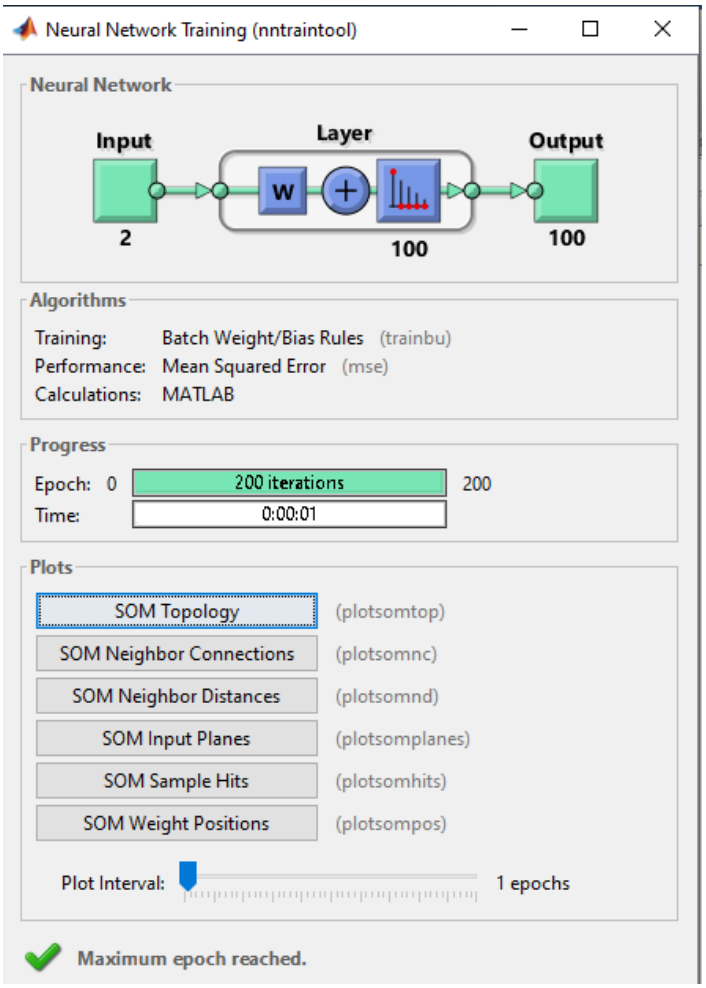
dimensions	Row vector of dimension sizes (**default = [8 8]**)
coverSteps	Number of training steps **(default = 100)**
initNeighbor	Initial neighborhood size **(default = 3)**
topologyFcn	Layer topology function **(default = 'hextop')**
distanceFcn	Neuron distance function **(default = 'linkdist')**

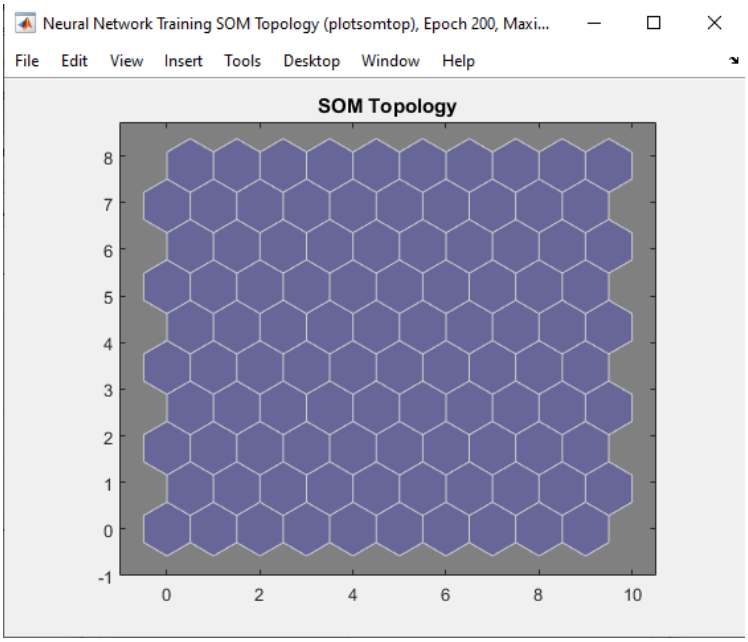
and returns a **self-organizing map**.

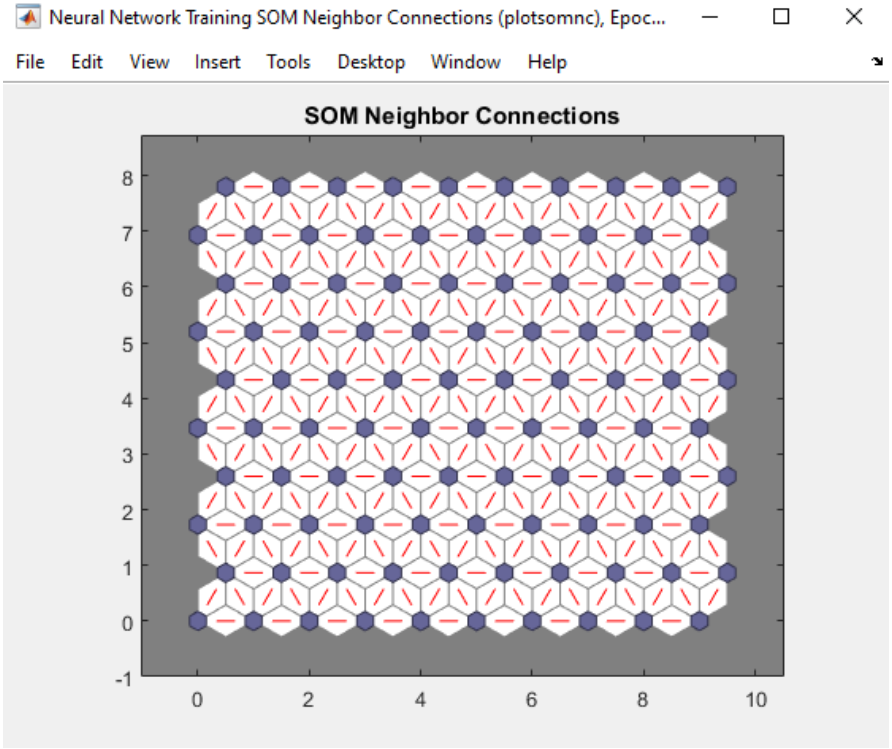
- The neurons in the layer of an **SOFM** are arranged originally in physical positions according to a **topology function**. The function **gridtop**, **hextop**, or **randtop** can arrange the neurons in a grid, hexagonal, or random topology.
- Distances between neurons are calculated from their positions with a **distance function**. There are four distance functions, **dist**, **boxdist**, **linkdist**, and **mandist**. Link distance is the most common.

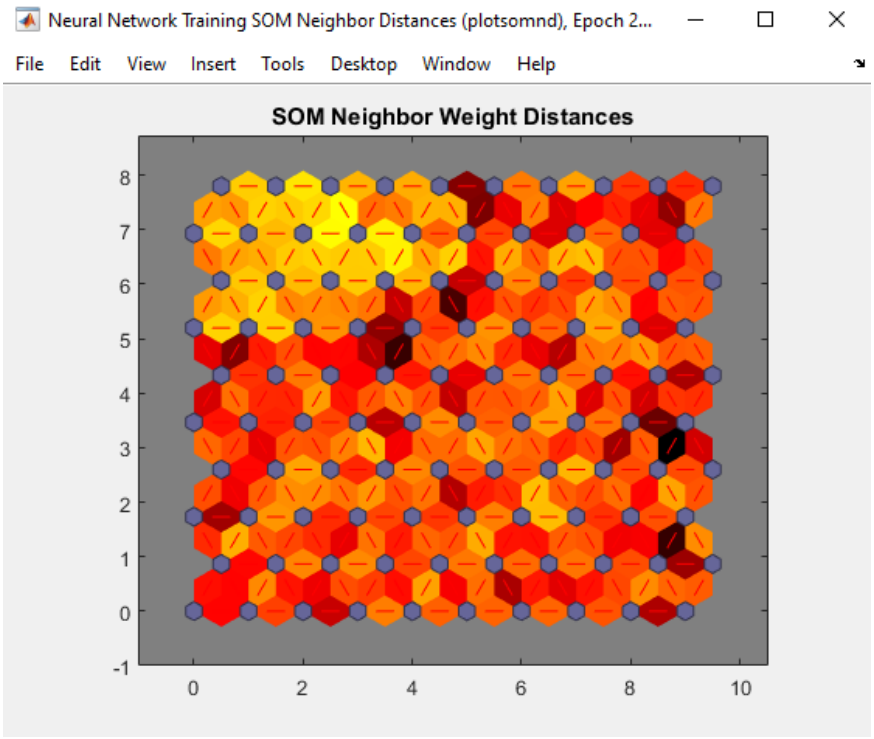
Example 1:

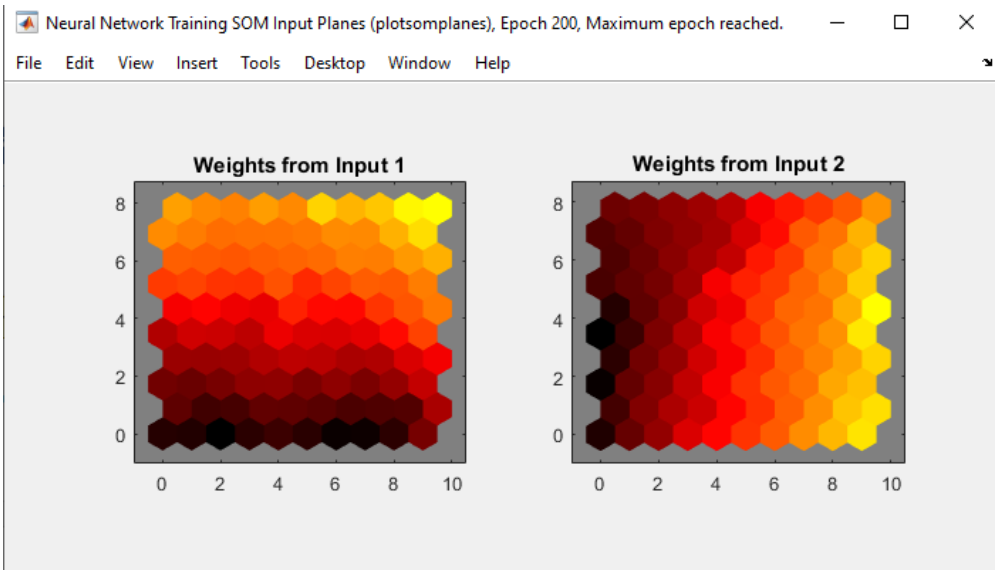
```
% Clustering Problem using a Self-Organizing Map
load simplecluster_dataset;
x = simpleclusterInputs;
% Create a Self-Organizing Map
dim1 = 10;
dim2 = 10;
net = selforgmap([dim1 dim2]);
% Train the Network
[net,tr] = train(net,x);
% Test the Network
y = net(x);
% View the Network

view(net)
```

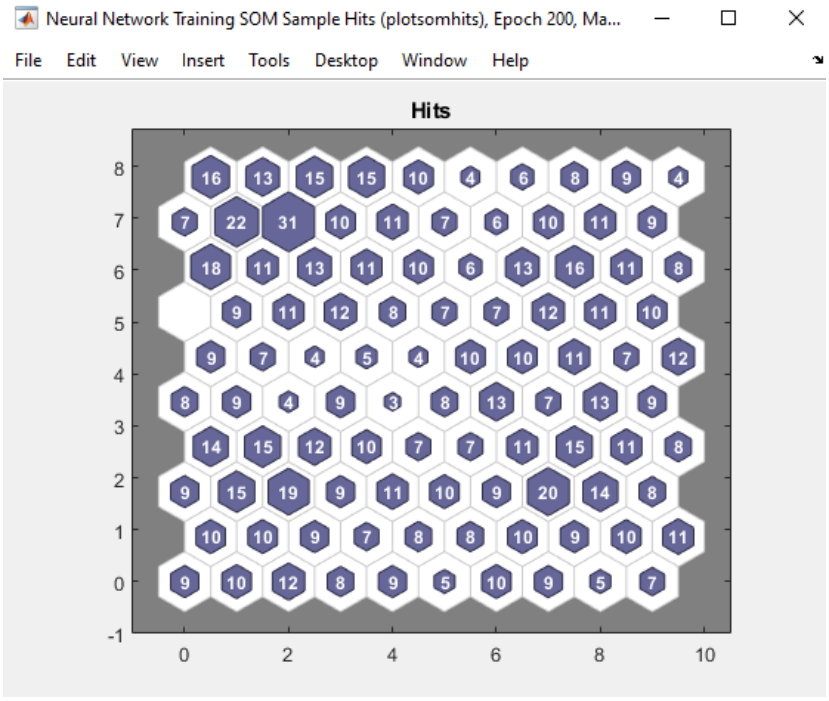

Neural Network Training (nntraintool)
Neural Network
Input
Layer
Output
2
100
100
Algorithms
Training: Batch Weight/Bias Rules (trainbu)
Performance: Mean Squared Error (mse)
Calculations: MATLAB
Progress
Epoch: 0
200 iterations
200
Time:
0:00:01
Plots
SOM Topology
(plotsomtop)
SOM Neighbor Connections
(plotsomnc)
SOM Neighbor Distances
(plotsomnd)
SOM Input Planes
(plotsomplanes)
SOM Sample Hits
(plotsomhits)
SOM Weight Positions
(plotsompos)
Plot Interval:
1 epochs
Maximum epoch reached.

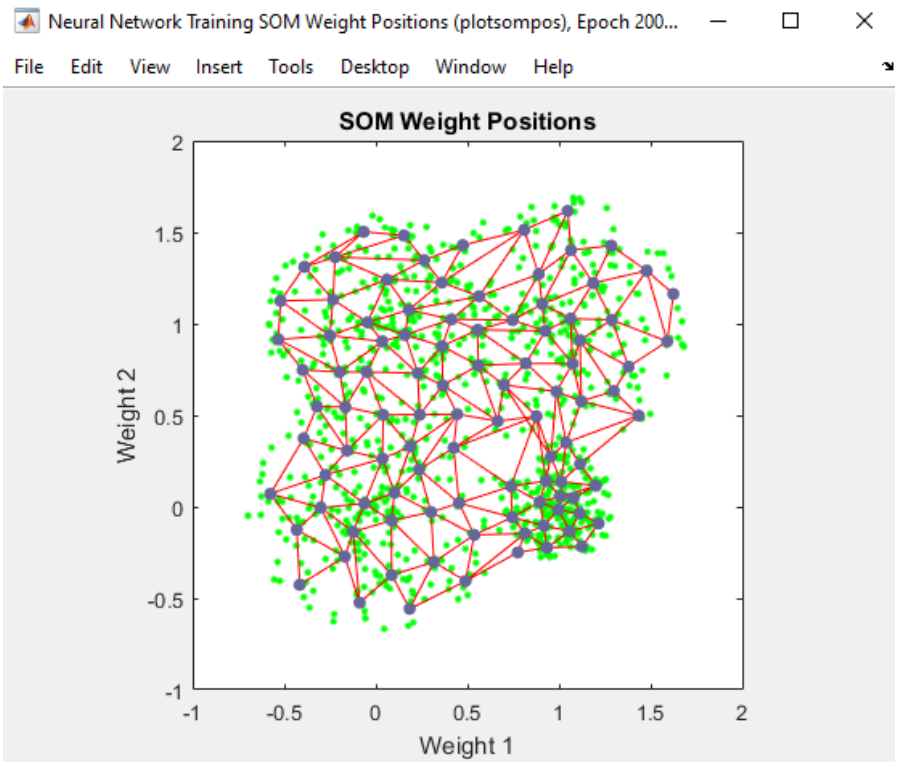

Neural Network Training SOM Topology (plotsomtop), Epoch 200, Maxi...
File Edit View Insert Tools Desktop Window Help
SOM Topology
8
7
6
5
4
3
2
1
0
-1
0
2
4
6
8
10


Neural Network Training SOM Neighbor Connections (plotsomnc), Epoc...
File Edit View Insert Tools Desktop Window Help
SOM Neighbor Connections
8
7
6
5
4
3
2
1
0
-1
0 2 4 6 8 10


Neural Network Training SOM Neighbor Distances (plotsomnd), Epoch 2...
File Edit View Insert Tools Desktop Window Help
SOM Neighbor Weight Distances
8
7
6
5
4
3
2
1
0
-1
0 2 4 6 8 10


Neural Network Training SOM Input Planes (plotsomplanes), Epoch 200, Maximum epoch reached.
File Edit View Insert Tools Desktop Window Help
Weights from Input 1
Weights from Input 2


Neural Network Training SOM Sample Hits (plotsomhits), Epoch 200, Ma...
File Edit View Insert Tools Desktop Window Help
Hits



Example 2: iris_dataset: This dataset can be used to create a neural network that classifies iris flowers into three types.

irisInputs - a 4x150 matrix of four attributes of 1000 flowers.

1. Sepal length in cm
2. Sepal width in cm
3. Petal length in cm
4. Petal width in cm

irisTargets - a 3x150 matrix of 1000 associated class vectors defining which of four classes each input is assigned to. Classes are represented by a 1 in one of four rows, with zeros in the others.

```
% Clustering Problem using a Self-Organizing Map
% iris_dataset.
load iris_dataset;
x = irisInputs;
% Create a Self-Organizing Map
dim1 = 10;
dim2 = 10;
net = selforgmap ([dim1 dim2]);
% Train the Network
[net,tr] = train(net,x);
```

```
% Test the Network
y = net(x);
% View the Network
view(net)
```