

CS 797O: Neural Nets and Deep Learning

Unit 3: Assignment 1- Neuralworks Iris

Self-Organizing Maps (SOM)

Abu Tyeb Azad
WSU ID: Q688C867

April 1, 2023

1 Basic Network

Grid Size = 10 x 10
Configuration = Square
Training Epochs = 75

1.1 Network Training

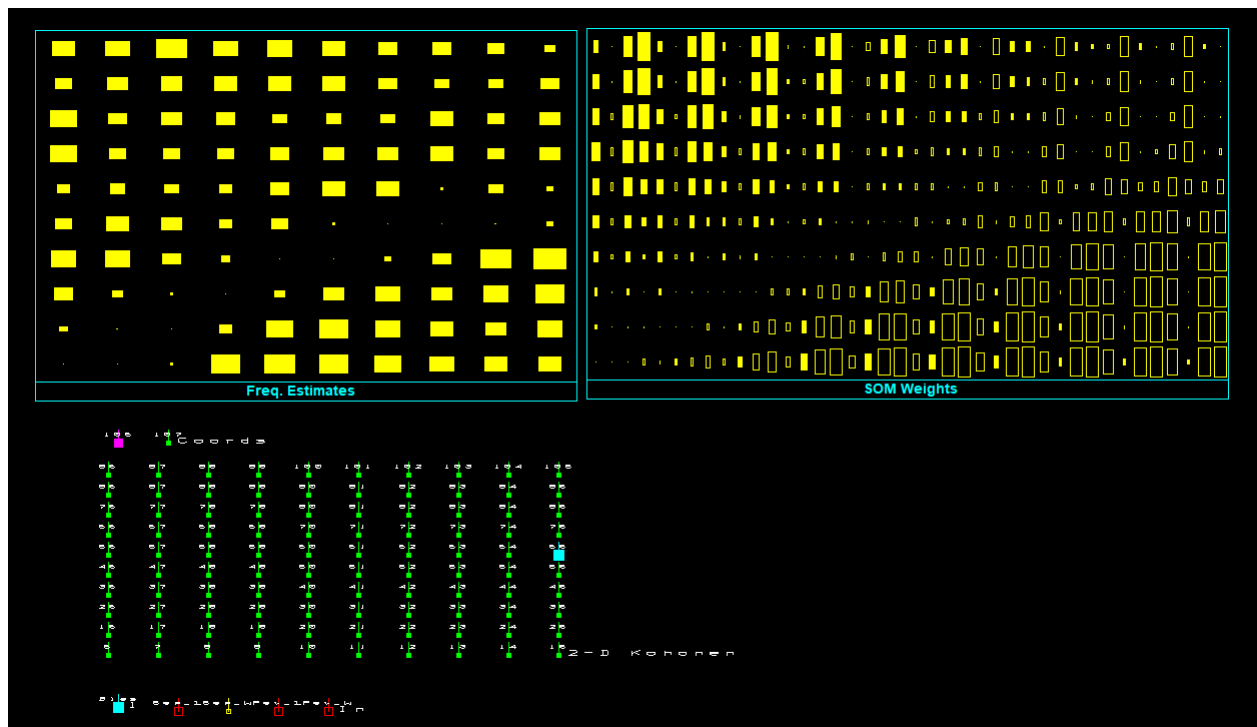


Figure 1: Basic Network Training

1.2 Network Testing

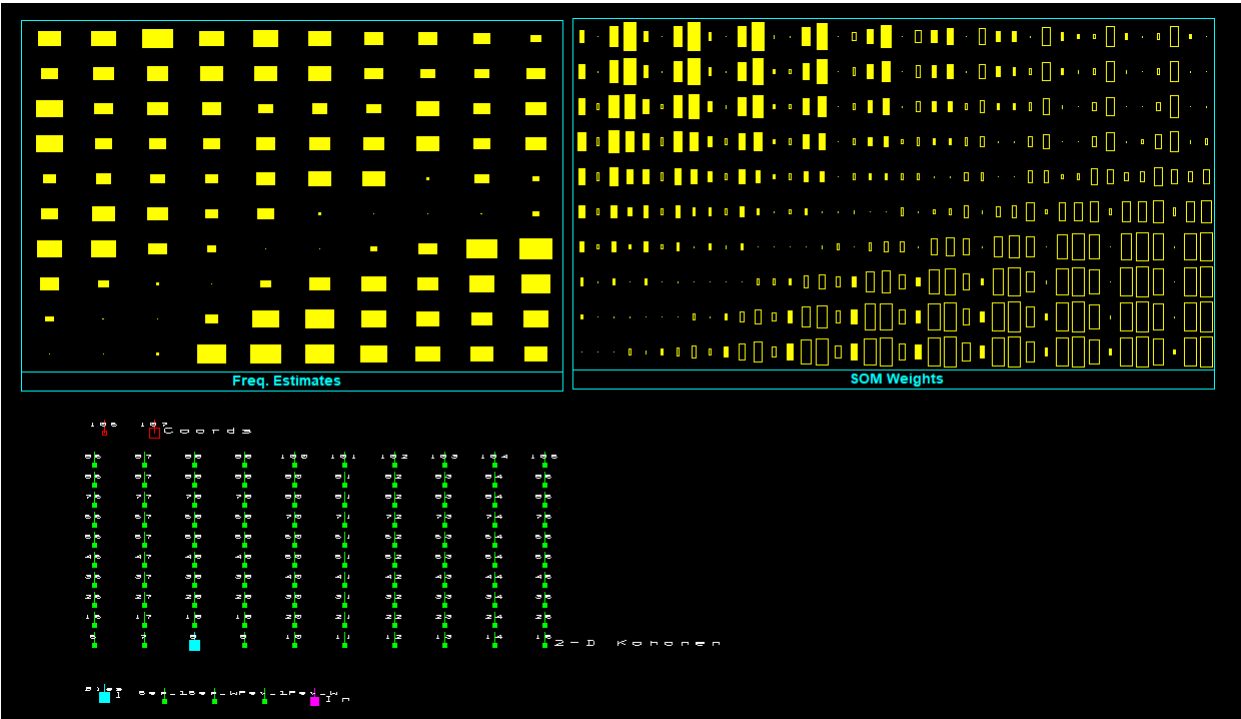


Figure 2: Basic Network Testing

1.3 Network Configuration

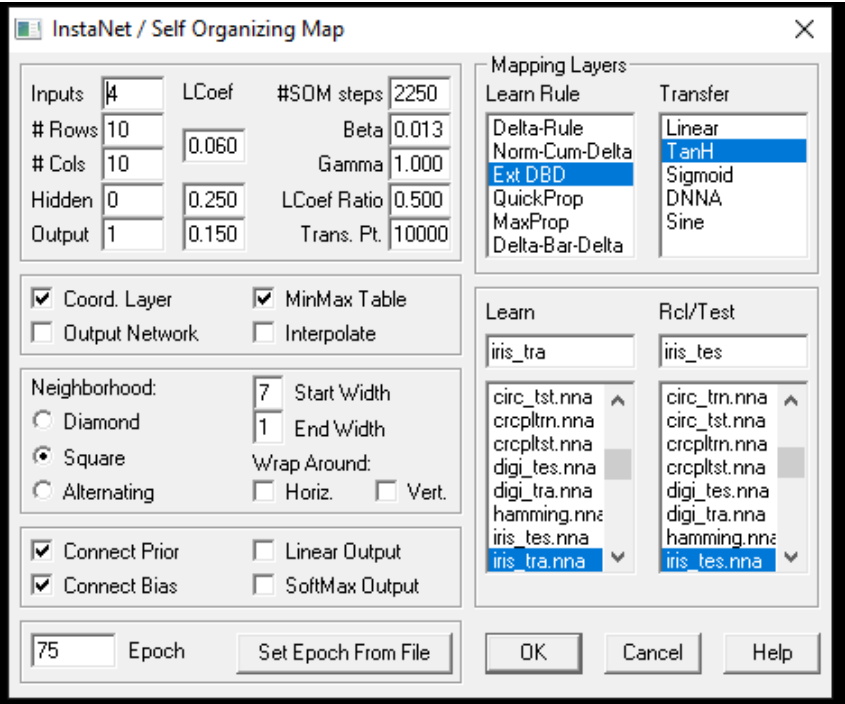


Figure 3: Basic Network Configuration

1.4 Excel Plot

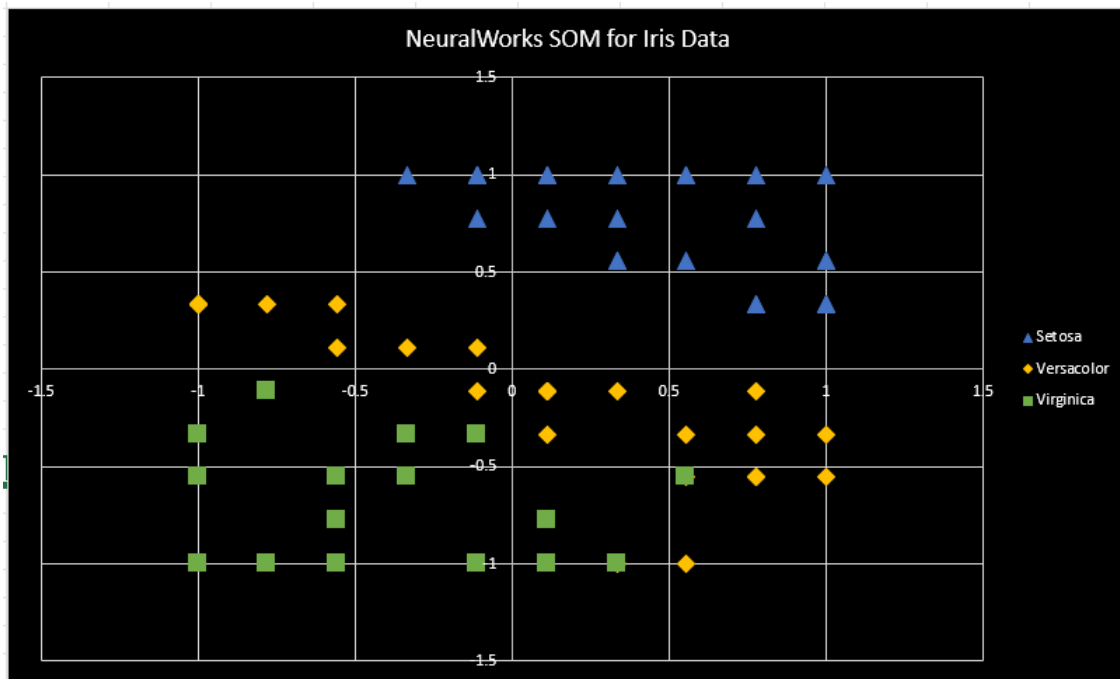


Figure 4: Basic Network Scatter Plot

2 Network 1

Grid Size = 15 x 15

Configuration = Diamond

Training Epochs = 400

2.1 Network Training



Figure 5: Network 1 Training

2.2 Network Testing



Figure 6: Network 1 Testing

2.3 Network Configuration

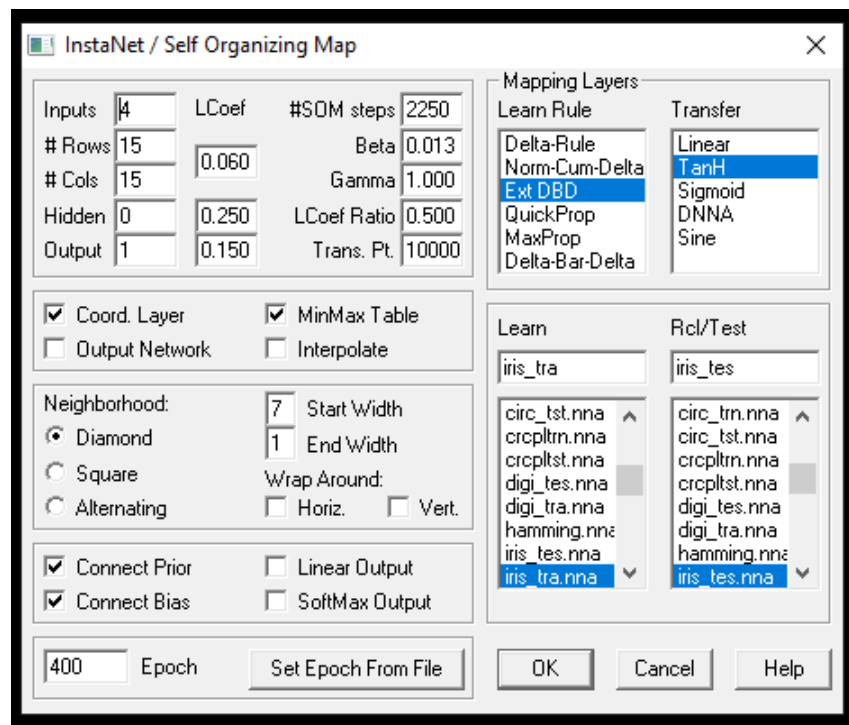


Figure 7: Network 1 Configuration

2.4 Excel Plot

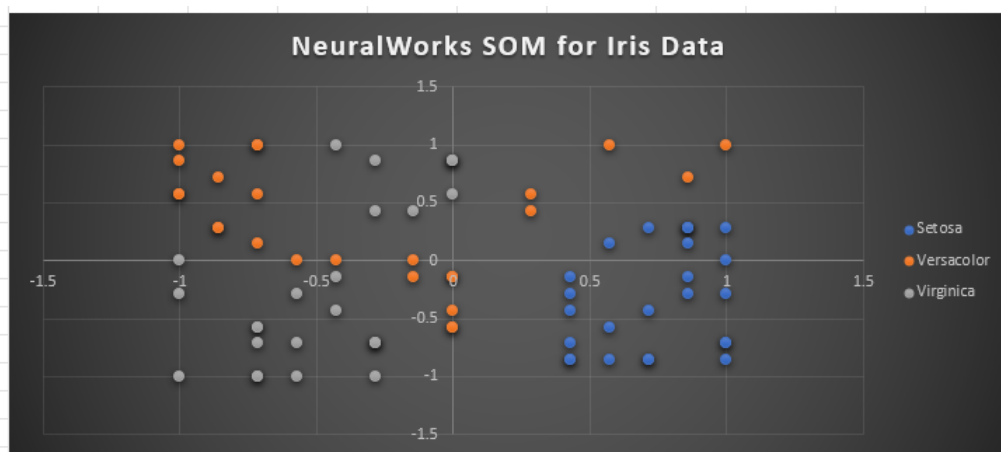


Figure 8: Network 1 Scatter Plot

3 Network 2

Grid Size = 20 x 20
Configuration = Diamond
Training Epochs = 200

3.1 Network Training

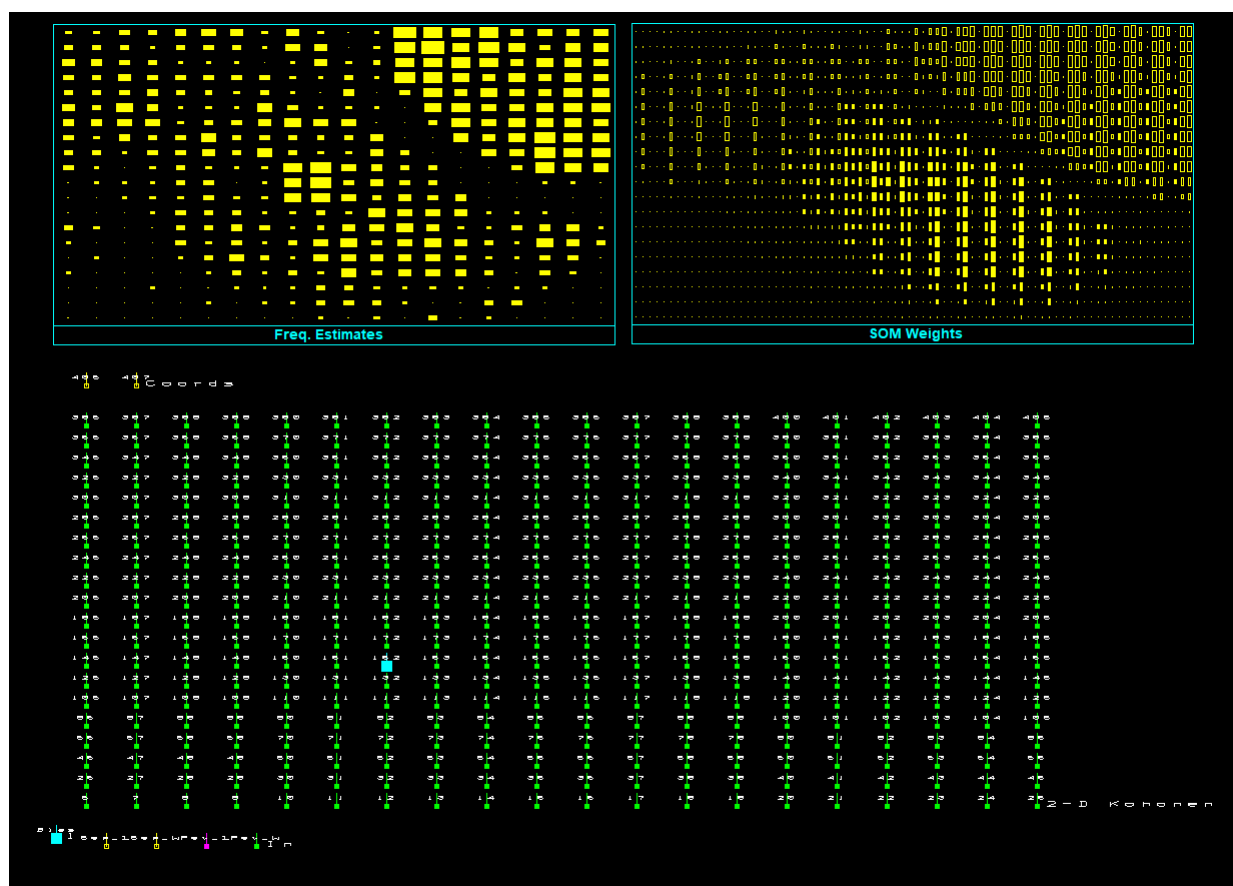


Figure 9: Network 2 Training

3.2 Network Testing

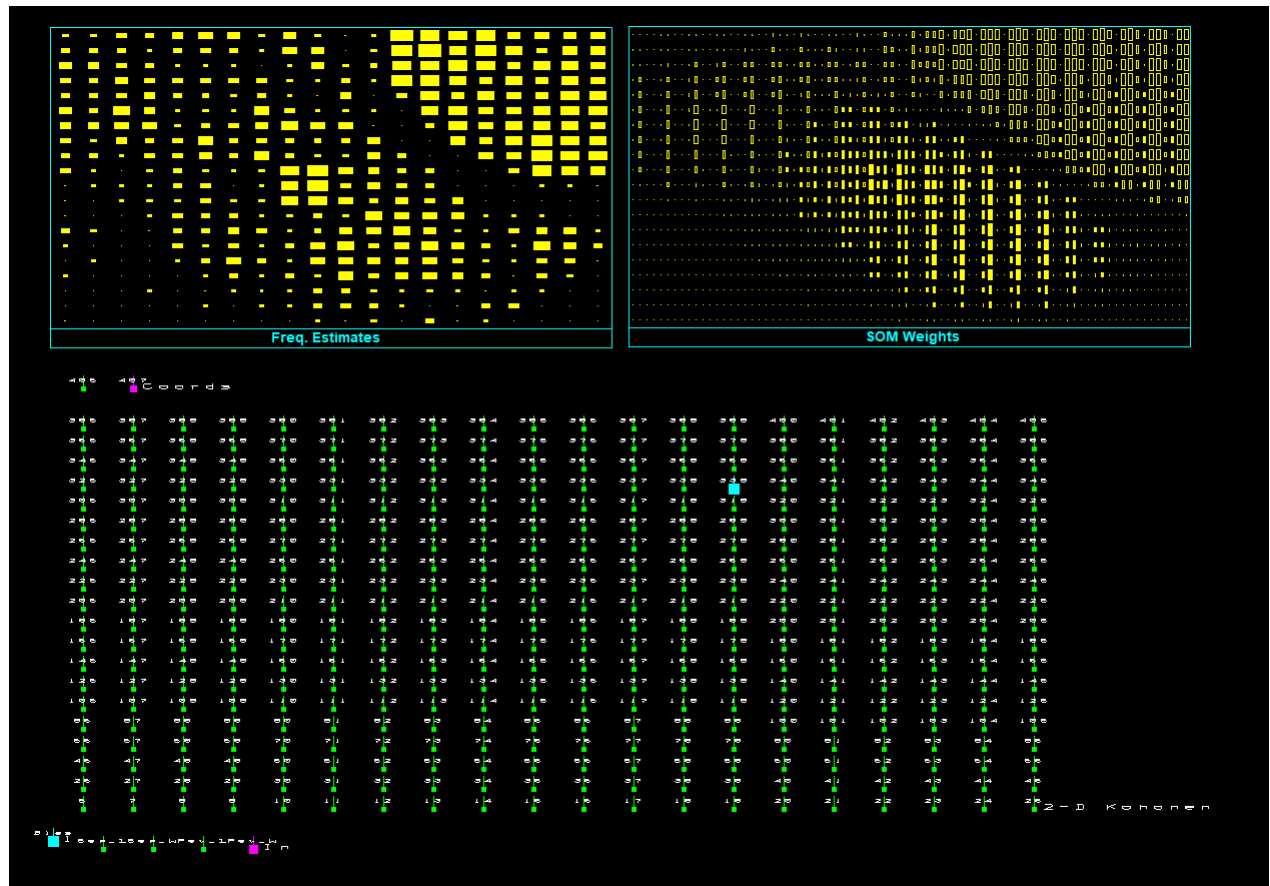


Figure 10: Network 2 Testing

3.3 Network Configuration

The screenshot shows the 'InstaNet / Self Organizing Map' configuration window. It is divided into several sections for setting network parameters.

- Inputs:** Inputs: 4, # Rows: 20, # Cols: 20, Hidden: 0, Output: 1.
- Learning Parameters:** LCoef: 0.060, Beta: 0.013, Gamma: 1.000, LCoef Ratio: 0.500, Trans. Pt: 10000, #SOM steps: 2250.
- Mapping Layers:** Learn Rule: Ext DBD (selected), Transfer: TanH (selected). Other options include Delta-Rule, Norm-Cum-Delta, QuickProp, MaxProp, and Delta-Bar-Delta.
- Options:** ☒ Coord. Layer, ☒ MinMax Table, ☐ Output Network, ☐ Interpolate.
- Neighborhood:** Diamond (selected), Start Width: 7, End Width: 1, Square, Alternating, Wrap Around: ☐ Horiz., ☐ Vert.
- Connectivity:** ☒ Connect Prior, ☐ Linear Output, ☒ Connect Bias, ☐ SoftMax Output.
- Epochs:** 200 Epoch, Set Epoch From File.
- Learn and Test Data:** Learn: iris_tra.nna, Test: iris_tes.nna. Lists of training and testing files are shown in scrollable areas.
- Buttons:** OK, Cancel, Help.

Figure 11: Network 2 Configuration

3.4 Excel Plot

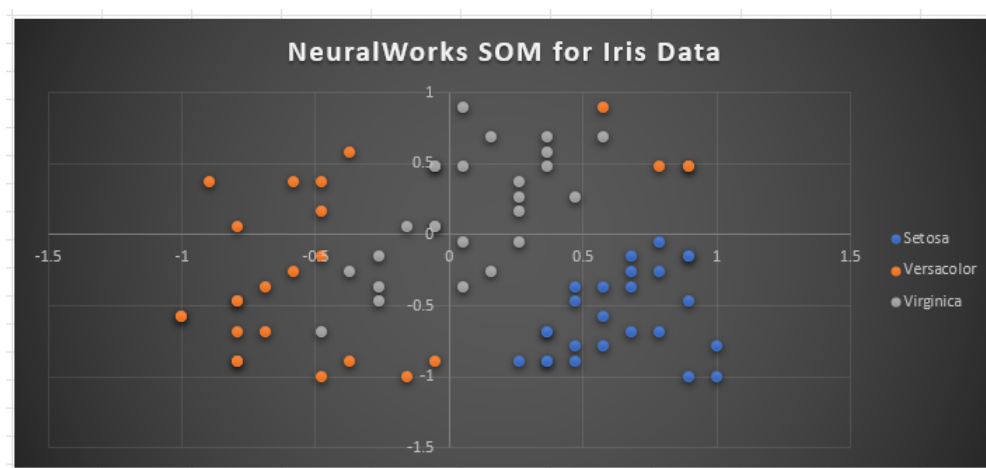


Figure 12: Network 2 Scatter Plot

4 Network 3

Grid Size = 20 x 20
Configuration = Diamond

Training Epochs = 400

4.1 Network Training



Figure 13: Network 3 Training

4.2 Network Testing

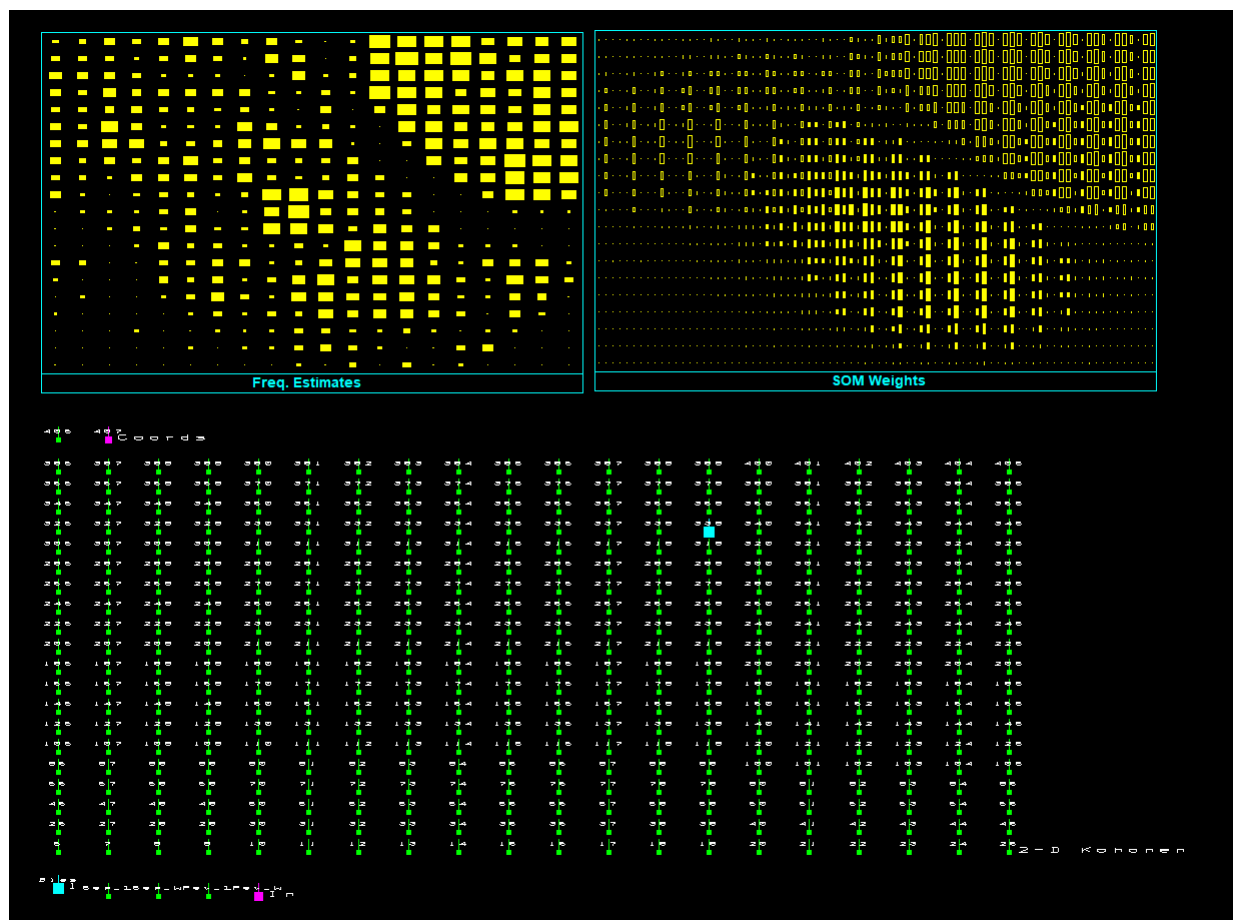
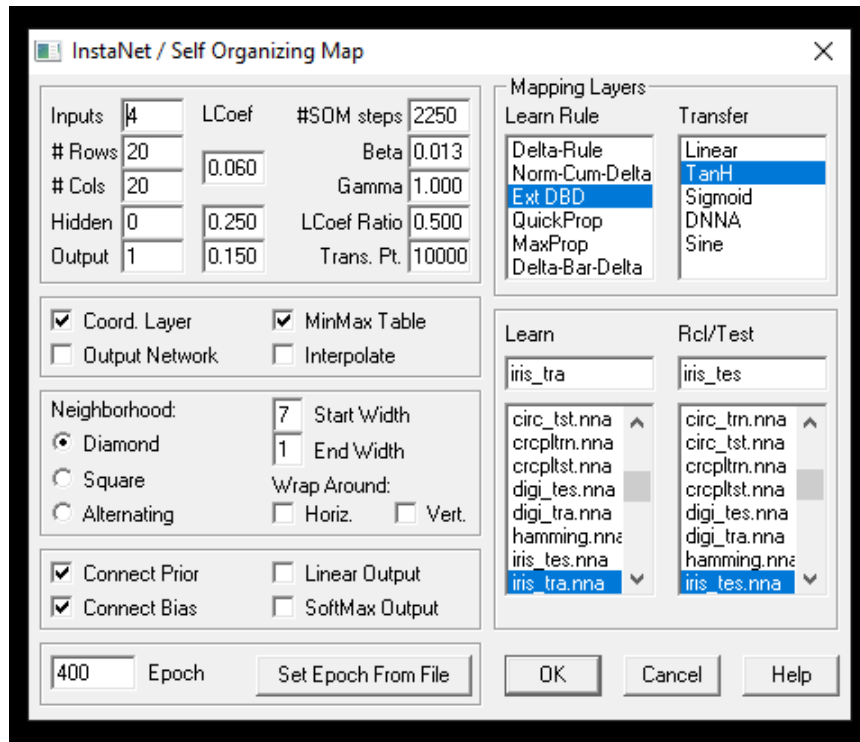


Figure 14: Network 3 Testing

4.3 Network Configuration



InstaNet / Self Organizing Map

Inputs: 4, # Rows: 20, # Cols: 20, Hidden: 0, Output: 1

LCoef: 0.060, Beta: 0.013, Gamma: 1.000, LCoef Ratio: 0.500, Trans. Pt: 10000

Mapping Layers

Learn Rule: Delta-Rule, Norm-Cum-Delta, **Ext DBD**, QuickProp, MaxProp, Delta-Bar-Delta

Transfer: Linear, **TanH**, Sigmoid, DNNA, Sine

Coord. Layer: ☒, Output Network: ☐, MinMax Table: ☒, Interpolate: ☐

Neighborhood: ☒ Diamond, ☐ Square, ☐ Alternating

Start Width: 7, End Width: 1, Wrap Around: ☐ Horiz., ☐ Vert.

Connect Prior: ☒, Connect Bias: ☒, Linear Output: ☐, SoftMax Output: ☐

Epoch: 400, Set Epoch From File

OK, Cancel, Help

Figure 15: Network 3 Configuration

4.4 Excel Plot

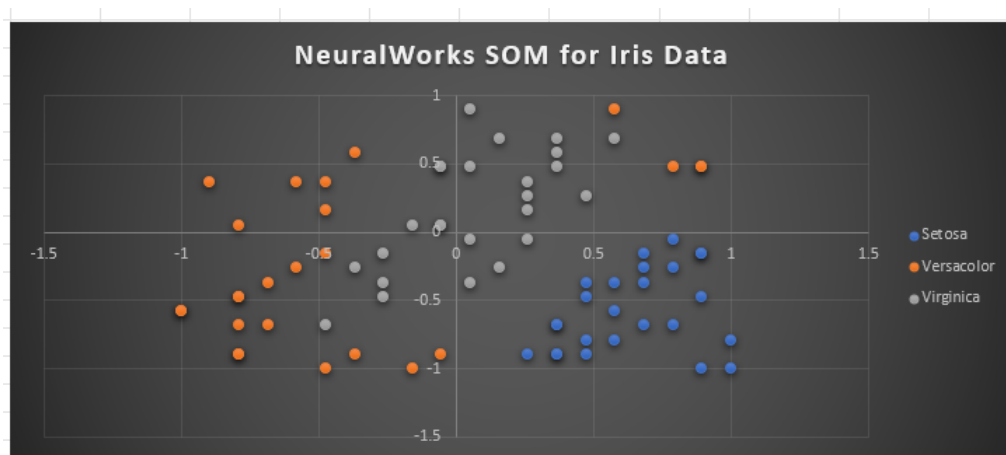


Figure 16: Network 3 Scatter Plot

5 Findings

-In general using diamond configuration helps in better clustering because the winning weights are less connected to distant weights. Since the no. of connections are less, the winner neuron/weight

has to cooperate (update weight) with less neighboring weights compared to square configuration. For example, for neighborhood radius $R=1$, in square configuration including winner weight, 9 weights are to be updated. But for same R , in diamond configuration 5 weights will be updated. And this continues to occur for other values of R . Usage of diamond configuration helps in minimizing long-range weight update and better cluster formation. In network 2 and 3 we can observe this effect.

-Increasing grid size improves performance because the SOM weights gets better distributed over the input space due to decreased competition among them. If grid size is larger (more weights in SOM), the weights tend not to compete for getting representation power over smaller input subspaces. As a result of that, the feature map generation process gets paced up and generates a better representation. For example, network 2 performs better than network 1. Other than a larger grid size (20x20) network 2 is similar to network 1.

-Increasing epochs helps in generating a better and stable feature map because the weights get more time to adjust themselves and their neighbors. Network 1, 2, and 3 perform better than basic network and they have more epochs than the basic net (400, 200, 400 respectively). But if the no. of epochs are set too high, further improvement may not occur. Network 2 and network 3 have the same diamond configuration and grid size. Increasing the no. of epochs for network 3 from 200 to 400, has not improved the feature map. In this case, the weights are getting saturated after around 200 epochs. So, addition of another 200 epochs is creating no effect on the weights.