UNIVERSITY OF BURGUNDY

Supervisor: Prof.Elizabeth Thomas

VISUAL PERCEPTION

Human Psychophysics [self-Organizing Maps]

SHAH Bhargav B

Email: shahbhargav1992@gmail.com

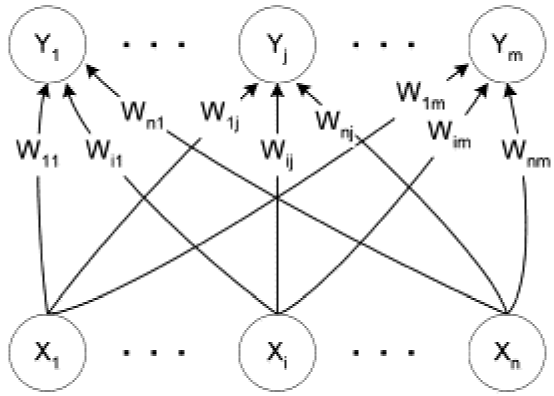
1. Summary

Our goal is using self-organizing map (SOM) to find classes in unlabeled data and then classify the data into the discovered classes. SOM is a two layer network with no feedback. With this structure the conventional repeated learning procedure is modified to learn just once. The once learning manner is more similar to human learning and memorizing activities. During training, weight updating is managed through a sequence of operations among some transformation and operation matrix. Every connection between neurons of input/output layers is considered as a independent processes. In this way all elements of the Euclidean distance matrix and weight matrix are calculated. The big idea is that similar data produce excitations in geometrically nearby output node in a trained network, and hence are mapped so that relations between them can be inferred from geometry of output nodes.

2. Introduction

The self-organizing map is one of the most popular neural network models. It's a data organization and analysis tool that allows us to discover complex relationships between high dimensional signals, and visualize them in a low dimensional grid. We use the SOM for clustering data without knowing the class membership of the input data. The SOM was developed by prof. Kohonen.

The self-organizing neural networks assume a topological structure among the cluster units. There are m cluster unit, arranged in a one or two dimensional array. The weight vector for a cluster unit serves are an examples of input patterns associated with the cluster. During the SOM process, the cluster unit whose weight vector matches the square of minimum Euclidean distance is chosen as the winner. In terms of topology of the cluster units update their weights. The weight vector of neighboring units are not, in general, close to the input pattern. The architecture and algorithm that follow for the net can be used to cluster a set of p continuous valued vector x = (x1,…xi,…..xn) into m clusters. Note that the connection weight do not multiply the signal sent from the input units to the cluster units.



Kohonen self-organized map

3. Algorithm

1. Initial weight wij. (Random values may be assigned for the initial weights as a possible choices)
2. Set topological neighborhood parameters R (spatial parameter)
3. Set the learning rate parameter α
4. For each input vector x, and for each j, compute: (Euclidian distance)

$$\boldsymbol{D(j) = \sum_i (w_{ij} - x_i)^2}$$

5. Find index J such that D (J) is minimum.
6. For all units j within a specified neighborhood of J and for all I update the weights:

$$\boldsymbol{w_{ij}(new = w_{ij}(old) + \alpha[x_i - w_{ij}(old)}$$

7. Update learning rate.
8. Reduce radius of topological neighborhood at specified times.
9. Test stopping condition.

Alternative statures are possible for reducing R and α. The learning rate α is a slowly decreasing function of time or training epochs.

4. Part 1

Construct a Kohonen network in order to carry out the classification of the vectors. (1 1 0 0)(1 0 0 0)(0 0 0 1)(0 0 1 1)

For this I made a function call **SOM_train.m** from training input it's automatically makes one cluster of vector (1 1 0 0) and (1 0 0 0) another cluster of (0 0 0 1) and (0 0 1 1). The initial learning rate is 0.6. In to the kohonen network the cluster is formed with weights as a centroid of the cluster.

Synaptic weight converged 63 iteration.

Final converged weight:

1.0000	0.0000
0.4732	0.0000
0.0000	0.5268
0.0000	1.0000

Once the training is completed carry out a test with the vector (0 0 0 0.9)(0 0 0.8 0.9)(0.7 0 0 0)(0.7 0.9 0 0)

For this I made a function **SOM_test.m** to find class. The vector (0 0 0 0.9) and (0 0 0.8 0.9) fall in class1 while the vector (0.7 0 0 0) and (0.7 0.9 0 0) fall in class2 by calculating Euclidian distance of each test vector with weight vector. And the minimum distance vector wins and assigns its class.

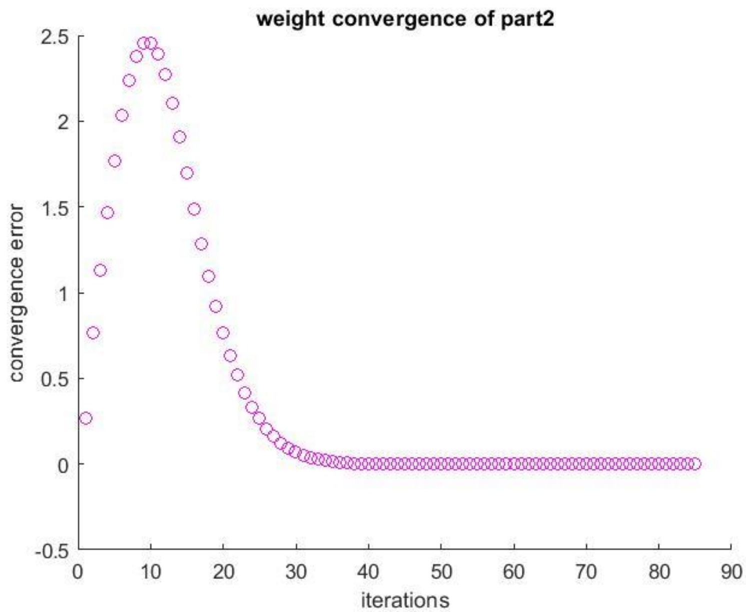
```
test_vector 1 is [0          0          0         0.9]
test_vector 2 is [0          0         0.8        0.9]
test_vector 3 is [0.7          0          0          0]
test_vector 4 is [0.7        0.9          0          0]
test vector 1 belongs to class 2
test vector 2 belongs to class 2
test vector 3 belongs to class 1
test vector 4 belongs to class 1
```

Test data is tested on the trained network

5. Part 2

Train your Kohonen networks using the training data set control.txt and patient.txt

In this create the kohonen network and train the data control.txt and patient.txt for that I made the **SOM_train.m** to fine synaptic weight and convergence error. No of iteration is 100 and learning rate is 0.6. **Synaptic weight converged 86 iterations.**



6. Part 3

Text file contains the data from 4 subjects. You have to identify which ones are patients or controls based on the kohonen network that you created.

The given data file name (test_two.txt) contain 4 subjects. According to previous data control and patient we identify which subject are control and patient in this part.

```
 $$$$$$$$$$$$$$$ PART 3 RESULT $$$$$$$$$$$$$$
test vector 1 belongs to control
test vector 2 belongs to control
test vector 3 belongs to control
test vector 4 belongs to control
```

7. Conclusion

PROS

1) SOM serves both to the dimensionality reduction data visualization and cluster analysis.
2) The method is simple, easy to explain and understand.
3) Numerous visualization possibilities.
4) Ability to handle large bases (linear complexity regarding the number of observation and variables)
5) This is a nonlinear approach for dimensionality reduction.
6) Two step approach for clustering is especially attractive.

7) We can classify data well and then are easily evaluate for their own quality so we can actually calculate how good a map is and how strong the similarities between object.

8) SOM have many practical application in Pattern recognition, speech analysis, industrial and medical diagnostics, data mining.

CONS

1) Processing time may be long on very large bases

2) The visualization and the interpretation become difficult when the number of variables is very high.

3) Every SOM is different and finds different similarities among the sample vector.

8. <u>Bonus Points Work</u>: what are the principle differences if any between a bio- inspired algorithm like the kohonen SOM and two well-known similar clustering algorithms – the kmeans and k nearest neighbor algorithm?

Kohonen SOM	K Means	K nearest
Clustering algorithm	Clustering algorithm	Classification algorithm
Provide robust learning	Provide sensitive to initialization	Function may be discrete valued or real valued
Unsupervised learning	Unsupervised learning or semi supervised	Supervised learning
SOM is check output and remove redundant variables.	It care about global	It care about local
SOM are formed geometrically	K mean are formed centroid and cluster size	K nearest neighbor very efficiently without calculating lots of distances.
More cluster give not better results	K mean give better results than SOM, if K is large	It is easily misled when instance space in high – dimensional.

Less sensitive	More sensitive	Dominated by large number of irrelevant features.
Useful when clustering is some kind of 2D visualization	Can be considered a special case of SOM were no neighbors are taken into account when modifying centroid vectors.	

9. References:

1. Class slides.
2. https://www.youtube.com/watch?v=DsYm4Tlr1rw