Controlling the spread of dynamic self-organising maps

Presented by: Hoi Fung Lam

Based on the paper: L. D. Alahakoon, ”Controlling the spread of dynamic self-organising maps,” *Neural Computing & Applications*, Volumn 13, Number 2, 2004, 168-174.

Content

- Introduction
- Background
- Weakness of traditional SOM
- Conclusion

Introduction

Problem:

- Traditional SOM does not provide a measure for identifying the size of a feature map with it's level of spread
- People can only refer to the length and width of the grid to relate to map size
- Predefined, fixed structure

Solution:

- Using spread factor (SF) in growing self-organising map to control the spread of the map

Background — Growing SOM

Initialization phase:

1. Initialize the weight vectors of starting nodes
2. Calculate the growth threshold (GT) for the given data set

Background — Growing SOM

Growing phase

1. Present input to the network
2. Determine the winner weight vector
3. Apply the weight adaptation to the neighborhood of the winner and to the winner itself
4. Increase the error value of the winner
5. When $TE_i \geq GT$, grow nodes if i is a boundary node, otherwise distribute weights to neighbors
6. Initialize the new node weight vectors to match the neighboring node weights
7. Reinitialize the learning rate to its starting value
8. Repeat 2 to 7 until all inputs have been presented, and node growth is reduced to a minimum level

Background — Growing SOM

Smoothing phase:

1. Reduce learning rate and fix a small starting neighborhood
2. Find winner and adapt weights in the same way as in the growing phase (instead of weight adaptation in original SOM, GSOM adapts its weight and architecture)

Background — The spread factor

The total error (TE) for a node i is calculated as:

$$TE_i = \sum_{H_i} \sum_{j=1}^{D} (x_{i,j} - w_j)^2$$

where H is the number of hits to the node i and D is the dimension of the data. $x_{i,j}$ and w_j are the input and weight vectors of the node i respectively.

Instead of having to provide GT, which would take different values for differen data sets, another value SF which will be used to calculate the GT value depending on the dimensionality of the data was used as follows:

$$GT = -D \times \ln(SF)$$

Why using GSOM

Relationship between the shape of the SOM and the input data distribution (Please refer to fig. 1)

- Inter-cluster distances have changed in such a way that the cluster positions have been forced into maintaining the proportions of the SOM grid
- The effect has been described by Kohonen as a limitation of the SOM called the *oblique orientation*

Conclusion

- Difference between GSOM and traditional SOM in terms of obtaining maps which fit into the data distribution
- Advantages of the parameter called spread factor in the GSOM model
- Traditional SOM required defining the *correct* grid size (rows to columns ratio) in order to arrive a *best fitting* grid
- A better alternative to traditional SOM

Benefits to our work

- Load balancing using SOM