Visualization with Voronoi Tessellation and Moving Output Units in Self-Organizing Map of the Real-Number System

Yuji Matsumoto, Motohide Umano and Masahiro Inuiguchi
Osaka University
Japan

1. Introduction

The Self-Organizing map (SOM) proposed by T. Kohonen (Kohonen, 1982; 1995), is a type of artificial neural network whose training is unsupervised. It produces a low-dimensional representation of high-dimensional input data, preserving the neighborhood relations as far as possible.

The SOM is difficult to interpret even with such coloring methods as the U-Matrix (Ultsch & Semon, 1990), the P-Matrix (Ultsch, 2003) or cluster connections (Merkl & Rauber, 1997). For interpretable maps, some methods are proposed to place and move the output units on the real-number coordinates plane. For example, Adaptive Coordinates (Merkl & Rauber, 1997) move output units towards the best matching unit but its map does not preserve the topology of the input data, where topology preservation means close vectors in the input space are mapped to nearby locations in output space (Kiviluoto, 1996).

For more interpretable map with preserving the topology, we propose a real-number SOM (RSOM), where we can not only place the output units as point freely on the real-number coordinates plane but also add, remove and moreover move them. Voronoi tessellation visualizes the map of the output units. RSOM is a natural extension of the conventional SOM because Voronoi tessellation for the output units on the square grid generates square regions on the output plane, the same as the conventional SOM. We illustrate several visualization methods such as minimum spanning tree and variable boundary, which help us to understand clusters or relations of input data. We also propose spherical RSOM to illustrate the power of RSOM that can place output units freely on an arbitrary surface.

2. Real-number SOM

The output unit of the conventional SOM is geometrically restricted to be square or hexagon with grid structure in two-dimensional visualization and with geodesic dome in three-dimensional visualization, which result less visual presentation. Fig. 1 shows a map of the conventional SOM with the input data in Table 1 (Kohonen, 1995), which has 16 animal names and 13 attributes.

		Dove	Hen	Duck	Goose	Owl	Hawk	Eagle	Fox	Dog	Wolf	Cat	Tiger	Lion	Horse	Zebra	Cow
is	small	1	1	1	1	1	1	0	0	0	0	1	0	0	0	0	0
	medium	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
	big	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
has	2 legs	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
	4 legs	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
	hair	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
	hooves	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
	mane	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	0
	feathers	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
likes to	hunt	0	0	0	0	1	1	1	1	0	1	1	1	1	0	0	0
	run	0	0	0	0	0	0	0	0	1	1	0	1	1	1	1	0
	swim	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0
	fly	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0

Table 1. Animal Names and Their Attributes (Kohonen, 1995)

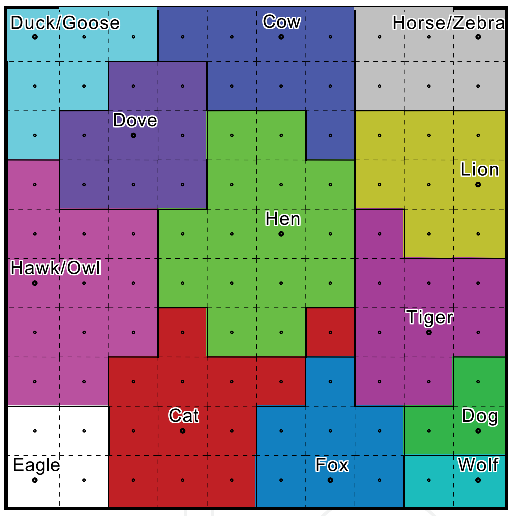


Fig. 1. The conventional SOM trained with input data in Table 1

We propose a real-number SOM (RSOM) whose output units can be freely placed on the real-number coordinates plane. The output unit can be considered as a point rather than a square or a hexagon in the conventional SOM. The visual shapes of these output units on the map of RSOM are determined by the positions of nearby units with Voronoi tessellation. RSOM is a natural extension of the conventional SOM because Voronoi tessellation for the output units on the square grid generates square regions on the output plane, the same as the conventional SOM.

In this section, we describe initialization, training and drawing for RSOM on the two-dimensional plane of $[0,1] \times [0,1]$. Please note that these processes are almost equivalent to the conventional SOM.

2.1 Initial Placement of the Output Units

The output units are placed on the real-number coordinates plane. Random placement is mainly used although many other methods can be applicable.

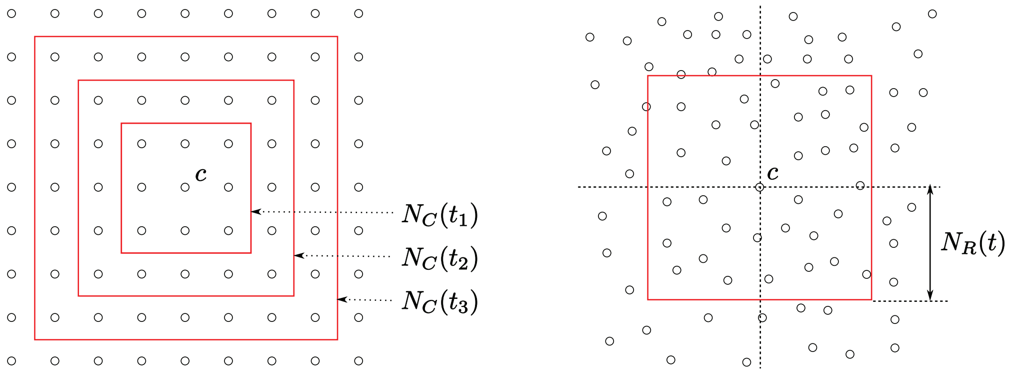
2.2 Training

2.2.1 Selection of the Best Matching Units

We select the best matching unit (BMU) c_i for an input vector $\boldsymbol{x}_i \in \mathbb{R}^n$, which is the smallest of the Euclidean distances of $\boldsymbol{x}_i$ to the reference vector $\boldsymbol{m}_j \in \mathbb{R}^n$ of output unit j:

$$c_i = \arg\min_j ||\boldsymbol{x}_i - \boldsymbol{m}_j||, \tag{1}$$

where $||\cdot||$ is the Euclidean norm and $\arg\min$ is the value of the argument j for which the value of the given expression $||\boldsymbol{x}_i - \boldsymbol{m}_j||$ attains its minimum value. This step is the same as the conventional SOM.



(a) Conventional SOM

(b) RSOM

Fig. 2. Neighborhood Function

2.2.2 Neighborhood Function

The conventional SOM uses discrete coordinates system and defines neighborhood units around the BMU c_i, whose neighborhood function can be defined with Chebyshev distance (i.e., chessboard distance):

$$h_j = \max(N_C(t) - |\boldsymbol{r}_{c_i} - \boldsymbol{r}_j|, 0), \tag{2}$$

where $\boldsymbol{r}_{c_i} \in \mathbb{R}^2$ and $\boldsymbol{r}_j \in \mathbb{R}^2$ are unit locations of the BMU c_i and an unit j on the output layer, respectively, $|\cdot|$ is a chessboard norm ($|\boldsymbol{r}| = \max(x, y)$, where $\boldsymbol{r} = (x, y)$) and $N_C(t) \geq 0$ is some monotonically decreasing function of discrete time t.

Fig. 2-(a) shows the neighborhood function of the conventional SOM, whose rectangle becomes small discretely by time (Kohonen, 1995).

In RSOM, we define a neighborhood function in the same manner as the conventional SOM in the following:

$$h_j = \max(N_R(t) - |\boldsymbol{r}_{c_i} - \boldsymbol{r}_j|, 0), \tag{3}$$

where $N_R(t)$ is a monotonically decreasing function of the time t, e.g., defined as follows:

$$N_R(t) = \max(\delta - \gamma t, 0), \tag{4}$$

where $\delta \geq 0$ and $\gamma \geq 0$ are parameters of neighborhood. Fig. 2-(b) shows the neighborhood function, whose rectangle becomes small smoothly by time.

We can use the neighborhood function of Euclidean norm instead of chessboard norm in Eq. (2): $h_j^E = \max(N_R(t) - ||\boldsymbol{r}_{c_i} - \boldsymbol{r}_j||, 0)$. Gaussian neighborhood function, widely used in the conventional SOM, is also available in RSOM:

$$h_j = \exp\left(-\frac{||\boldsymbol{r}_{c_i} - \boldsymbol{r}_j||^2}{2\sigma^2(t)}\right), \tag{5}$$

where $\sigma(t)$ is the width of the neighborhood at the time t and $|| \cdot ||$ is the Euclidean norm.

2.2.3 Update

In the same manner as the conventional SOM, reference vectors $\boldsymbol{m}_j$ of output units j in the neighborhood of the BMU c_i are adjusted to the input vector $\boldsymbol{x}_i$ with the following rule:

$$\boldsymbol{m}_j := \boldsymbol{m}_j + \alpha(t) h_j \left(\boldsymbol{x}_i - \boldsymbol{m}_j\right), \tag{6}$$

where $\alpha(t)$ is a learning rate at the time t, e.g., defined as follows:

$$\alpha(t) = \alpha_0(1 - t/T), \tag{7}$$

where α_0 is the initial learning rate and T is the number of iteration steps.

2.3 Drawing the Output Layer

The regions of the output units are not explicitly specified since the units are points in RSOM. For visualizing the output layer, we use Voronoi tessellation to divide the output plane into the regions of units, where each point on the output plane belongs to the closest unit. The boundaries of these regions are drawn with dotted line when the dominant classes of two adjacent regions are the same and solid line when these are different. We also put the labels of the dominant class on the BMUs. Fig. 3 shows a map for Table 1 with RSOM, where we place the output units randomly at initial time and use the following parameters: $\alpha_0 = 0.1$, $T = 1000$ and $N_R(t) = 1 - t/T$.

When we place the output units on the square grid, Voronoi tessellation generates square regions on the output plane, like as the conventional SOM, shown in Fig. 1.

Choosing some parameters and functions in RSOM properly, we can have same results as the conventional SOM. Fig. 4 shows other placement patterns of hexagonal grid, spiral, double concentric circle and Sierpinski triangle.

3. Visualization Methods for RSOM

For more interpretable map, we propose two visualization methods such as minimum spanning tree and variable boundary width.

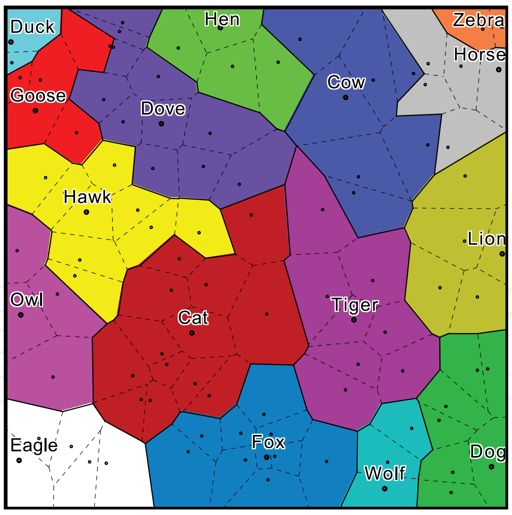


Fig. 3. Voronoi diagram for random placement of RSOM

3.1 Drawing Variable Boundary Width

We can not easily interpret the clusters of “animal family” in Fig. 3. We emphasize the boundaries of two regions whose reference vectors have big difference. When two units j and k are adjacent each other, we draw their boundary with width w_{jk}:

$$w_{jk} = ||\boldsymbol{m}_j - \boldsymbol{m}_k|| \cdot W, \tag{8}$$

where W is multiply factor of boundaries. Fig. 5 shows a Voronoi diagram with variable boundary width for Fig. 3, where boundaries of two similar units are thiner than the ones of different. Fig. 5 is easier to interpret clusters, e.g., *bird*, than Fig. 3.

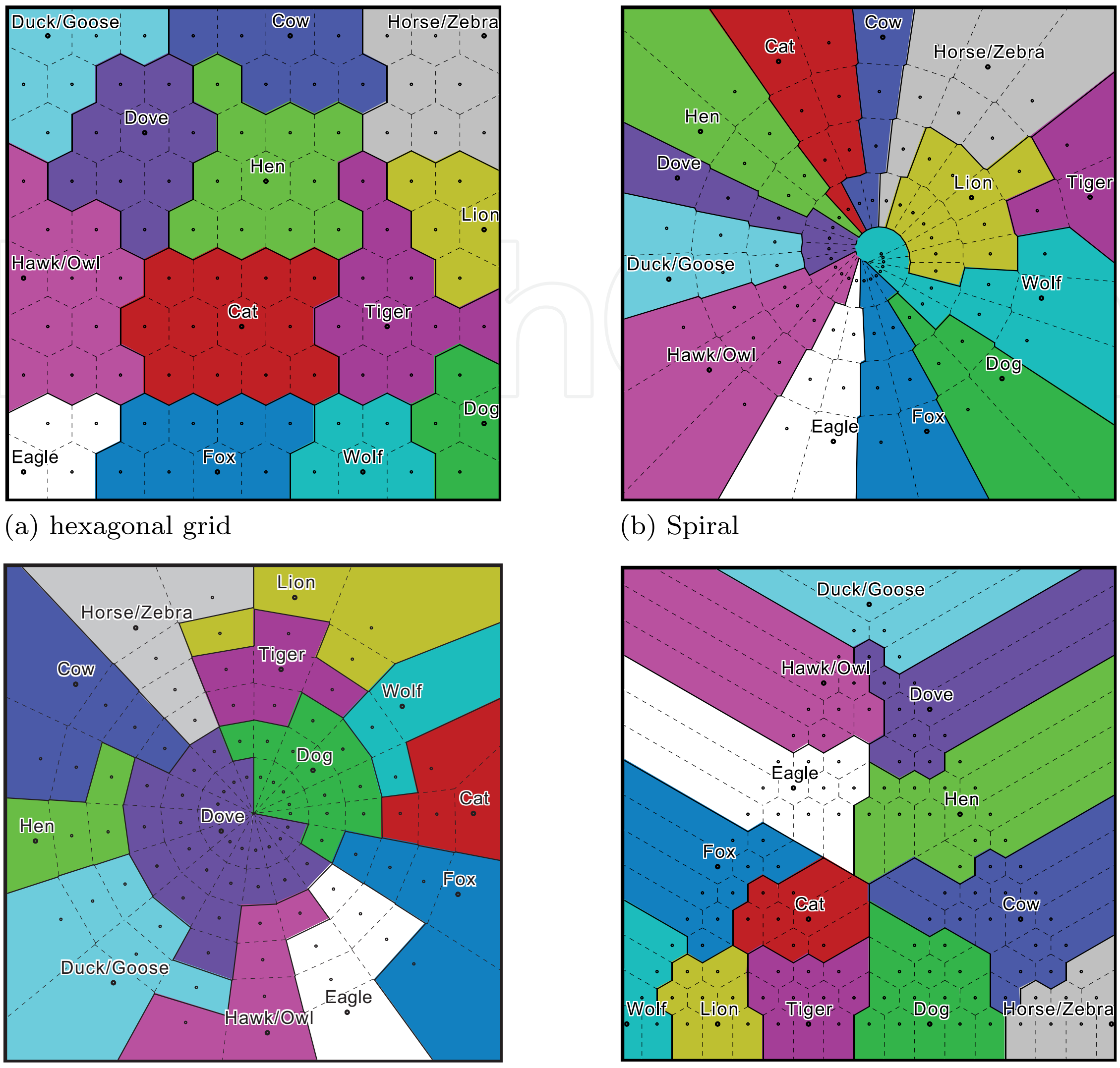
3.2 Drawing Minimum Spanning Tree

Variable boundary width helps us to understand some groups of input. However, this visualization method does not help us to understand relationships between two units very well. For more intuitive understanding, we visualize a tree connecting close units. We have the dual graph for a Voronoi diagram, called the Delauney triangulation whose edges connect adjacent regions in Voronoi diagram. Like Voronoi diagram of RSOM, Delauney triangulation of RSOM shows topological relationship of input vectors in Fig. 6-(a), but this is not easily understandable. Minimum spanning tree in which the weight of each edge is defined in Eq. (8) shows relationships among input vectors like a dendrogram with hierarchical clustering. Fig. 6-(b) shows the minimum spanning tree of Delauney triangulation for the map in Fig. 3, where we can interpret relation of inputs. We can get more understandable maps to use the combination of two visualization methods, variable boundary width and minimum spanning tree in Fig. 7

4. Spherical RSOM

Spherical SOM is useful since it removes the “border effect.” The conventional spherical SOM uses a geodesic dome restricted the number of output units. We, therefore, propose spherical RSOM, whose output units can be freely placed on a unit sphere.

The algorithm for spherical RSOM is the same as that of RSOM except the coordinates systems, the geometrical norm and visualization method of the output layer. For the output units, we



(a) hexagonal grid

(b) Spiral

(c) double concentric circle

(d) Sierpinski triangle

Fig. 4. Voronoi tessellation of hexagonal grid, spiral, double concentric circle and Sierpinski triangle

use polar coordinates $\boldsymbol{s} = (\theta, \varphi)$ $(0 \leq \theta \leq 2\pi$ and $0 \leq \varphi \leq \pi)$ and the sphere norm derived from the spherical law of cosines:

$$||\boldsymbol{s}|| = \arccos(\cos\theta, \cos\varphi). \tag{9}$$

Fig. 8 and Fig. 9 show a result of spherical RSOM and its flat map, respectively, where we place output units randomly and use the following parameters: $\alpha_0 = 0.2$, $T = 1000$ and $N_R(t) = 0.5\pi(1 - t/T)$.
Nishio et al. proposed a method for placement on a surface of a sphere with helix (Nishio et al., 2006), in which the number of the units is not restricted, but it does not add, remove and move the units. We can use helix for unit placement of spherical RSOM, and moreover place output units on an arbitrary surface.

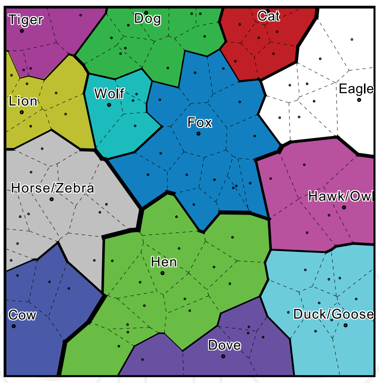


Fig. 5. Voronoi diagram with variable boundary width for Fig. 3

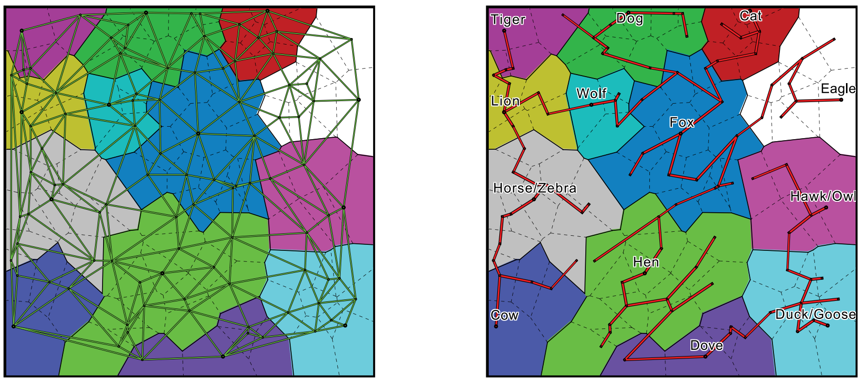


Fig. 6. (a) Delauney triangulation (b) Minimum spanning tree

5. Related Works

RSOM can not only freely place the output units on an arbitrary surface for visualization, but also add, remove and move the output units for more understandable visualization.
Grawing Grid(Fritzke, 1995a) is a variant of SOM. Grawing Grid initially has 2 × 2 units and adds a new column (or row) of units when quantization error of a map is high. Grawing Grid considers a some grid network and does not consider move and remove units.
Growing Neural Gas (Fritzke, 1994), (Fritzke, 1995b) uses similar growth technique of Growing Method. Growing Neural Gas can add and remove units but has no feature map. If we have a 2-D map of Growing Neural Gas, we have to use some method, e.g., simulated annealing to create minimum connections(Ogura et al., 2003).

5.1 Visualization Methods

Two visualization methods mentioned above, variable boundary width and drawing minimum spanning tree can be applicable for the conventional SOM.
In the conventional SOM, some visualization methods are developed.
In U-Matrix (Ultsch & Semon, 1990), the relative distances between neighboring reference vectors are drawn in a gray scaled color. The P-Matrix (Ultsch, 2003) visualizes the density relationships in the input space with the Pareto Density Estimation.
In cluster connections (Merkl & Rauber, 1997), connections of the units are represented by a gray scaled color with distances $||\boldsymbol{m}_j - \boldsymbol{m}_k||$.

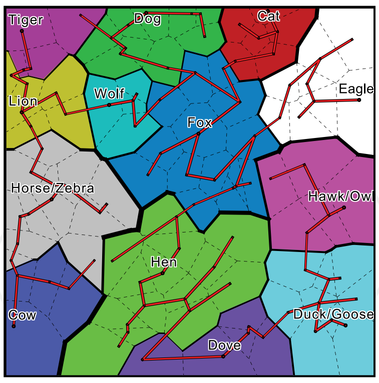


Fig. 7. Variable boundary width overlaying minimum spanning tree for Fig. 5

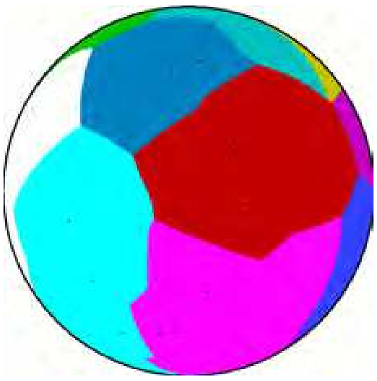


Fig. 8. RSOM on a surface of sphere

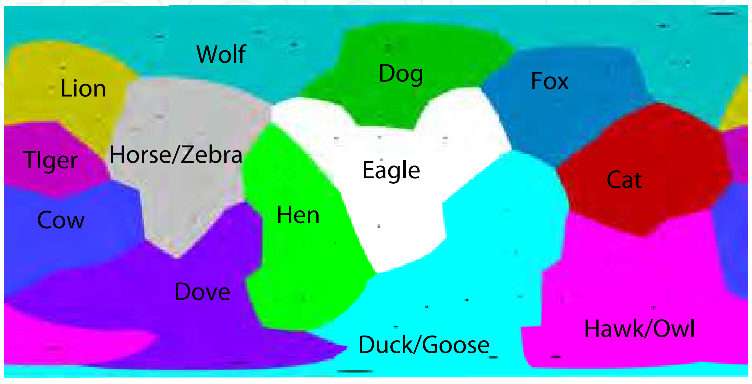


Fig. 9. A flat map of the spherical RSOM in Fig. 8

6. Conclusions

We proposed a real-number SOM (RSOM), which can place output units freely on an arbitrary surface for visualization, with removing and moving the units. We used Voronoi tessellation for visualizing RSOM more expressively. We proposed several visualization methods such as the minimum spanning tree, the variable boundary width, which help us to understand clusters or relations of input data, and the spherical RSOM.

7. References

Fritzke, B. (1994). Fast learning with incremental rbf networks, *Neural Processing Letters* 1(1): 2–5.

Fritzke, B. (1995a). Growing grid—a self-organizing network with constant neighborhood range and adaptation strength, *Neural Processing Letters* 2(5): 9–13.

Fritzke, B. (1995b). *A growing neural gas network learns topologies*, MIT Press, pp. 625–632.

Kiviluoto, K. (1996). Topology preservation in self-organizing maps, *IEEE International Conference on Neural Networks*, pp. 294–299.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps, *Biological Cybernetics* 43: 59–69.

Kohonen, T. (1995). *Self-Organizing Maps Third Edition*, Springer-Verlag.

Merkl, D. & Rauber, A. (1997). Alternative ways for cluster visualization in self-organizing maps, *Workshop on Self-Organizing Maps, Espoo, Finland*, pp. 106–111.

Nishio, H., Altaf-Ul-Amin, M., Kurokawa, K. & Kanaya, S. (2006). Spherical som and arrangement of neurons using helix on sphere, *Information Processing Society of Japan Digital Courier* 2: 33–137.

Ogura, T., Iwasaki, K. & Sato, C. (2003). Topology representing network enables highly accurate classification of protein images taken by cryo electron-microscope without masking, *Journal of structural biology* 143(3): 185–200.

Ultsch, A. (2003). Pareto density estimation: Probablity density estimation for knowledge discovery, *27th Annual Conference of the German Classification Society*, pp. 91–100.

Ultsch, A. & Semon, H. P. (1990). Kohonen's self-organiring feature maps for exploratory data analysis, *International Neural Network Conference, Dordrecht, Netherlands*, pp. 305–308.

Self-Organizing Maps
Edited by George K Matsopoulos

ISBN 978-953-307-074-2
Hard cover, 430 pages
Publisher InTech
Published online 01, April, 2010
Published in print edition April, 2010

The Self-Organizing Map (SOM) is a neural network algorithm, which uses a competitive learning technique to train itself in an unsupervised manner. SOMs are different from other artificial neural networks in the sense that they use a neighborhood function to preserve the topological properties of the input space and they have been used to create an ordered representation of multi-dimensional data which simplifies complexity and reveals meaningful relationships. Prof. T. Kohonen in the early 1980s first established the relevant theory and explored possible applications of SOMs. Since then, a number of theoretical and practical applications of SOMs have been reported including clustering, prediction, data representation, classification, visualization, etc. This book was prompted by the desire to bring together some of the more recent theoretical and practical developments on SOMs and to provide the background for future developments in promising directions. The book comprises of 25 Chapters which can be categorized into three broad areas: methodology, visualization and practical applications.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821