Open Systems & Information Dynamics
Vol. x No x (1994) xx-xx

Quantitative measures for the Self-Organizing Topographic Maps

WŁODZISŁAW DUCH[1]

Department of Computer Methods, Nicholas Copernicus University, Grudziądzka 5, 87-100 Toruń, Poland

(Received April 7, 1994)

Self-organizing topographic maps have found many applications as systems capable of unsupervised learning. They are based on the competitive learning algorithm applied to low-dimensional (in practice one, two or three-dimensional) structure of artificial neurons. The iterative algorithm used for competitive learning converges slowly and is computationally very intensive. In this paper direct mapping on the continuos space based on the minimization principle is used to map the high-dimensional input data to the low-dimensional target space. The problem of finding the best low-dimensional representation of the data is reduced to a minimization problem or to the solution of a system of nonlinear algebraic equations.

1 Introduction

In the past decade research on adaptive systems, i.e. systems that can learn, internalize and recognize certain data and patterns, became very active [1]. There are two kinds of learning procedures. First, in supervised learning a "teacher" is correcting the mistakes of the system, requiring perfect repetition of a set of examples (such as multiplication table entries) and meaningful generalization of the learned knowledge. Second kind of learning,

[1] duch@phys.uni.torun.pl

the spontaneous, unsupervised learning, is exemplified perhaps in the best way by the infants recognizing the shapes and the sounds and creating and internal model of reality from the sensory experiences.

Self-organizing topographic map (SOM) [2-5] is one of the most popular models of unsupervised learning. In essence it is a competitive learning algorithm capable of preserving topological relations present in the input data. Great interest in this type of algorithms comes from the fact that the brain uses many feature maps, such as visual field, somatotopic and tonotopic maps. It is not clear how these maps are formed in the process of development of humans and animals (for a review of neurobiological facts and theories see [6]). Although the real excitement in the neural models of self-organization started with the paper of Kohonen in 1982 [2] similar approaches discussing formation of such maps were proposed earlier by Willshaw and von der Malsburg [7] and Amari [8].

Since the publication of the original paper by Kohonen [2] a large number of papers were devoted to the application of the idea of the self-organizing mapping. Some of the applications include [8]:

- statistical pattern recognition, recognition of speech, signal processing, adaptive filters, real-time data processing, analysis of radar, sonar, infrared, medical, seismic and other signals;
- image segmentation, compression and general data compression problems;
- various problems in control theory, control of industrial processes, control of movement in robotics;
- classification of data, for example classification of protein properties or insect courtship songs
- optimization problems
- sentence understanding, artificial intelligence problems

Despite all these applications the mathematical foundations of the method are not clear and most of the results on self-organization are obtained from numerical experiments. The competitive learning approach has other disadvantages: it is slowly convergent, numerically quite expensive, hard to parallelize efficiently and therefore in practical applications the largest networks have only about 1000 cells. Although one can use many nets hierarchically organized into larger system some processes, such as speech recognition, require much larger nets. In this short paper I will give a solid mathematical foundations to the formation of the self-organizing topographical maps, connecting it with the minimization principles. The complexity of the mapping is related to the number and the dimension of the objects and not to the size of the network itself.

2 Self Organizing Topographic Map

For the description of the SOM algorithm the reader is referred to the papers [2-5]; here for the sake of brevity I will restricted myself to a few remarks only. Kohonen algorithm of self-organization is based on a projection from a space of data of high dimension N (input or signal space) to a low n dimensional (usually one, two or three-dimensional) network of artificial neurons (target space), each neuron equipped with a set of N adaptive parameters called weights. Neurons specialize in recognition of patterns in the data by changing these weights in a process of competitive learning. In effect similar data will excite similar neurons. If a new input data vector **x** is presented to the system neuron number c with the weights that are the most similar (in the sense of Euclidean distance) to the vector **x** is selected and the weights of this neurons and adjacent neurons are changed to make them more similar to **x**:

$$\mathbf{W}_i(t+1) = \mathbf{W}_i(t) + h_c(r)[\mathbf{x}(t) - \mathbf{W}_i(t)] \tag{2.1}$$

The function

$$.h_c(r) = h_0(t)\ \exp(-\|r - r_c\|^2/\sigma^2(t)) \tag{2.2}$$

defines a neighborhood around the position of the neuron c, h_0 controls how much is the weight of the neuron c is changed and σ how much are the adjacent neurons affected. In this way two problems are solved at the same time: the weights serve as the prototypes for the data clusters in the input space and a mapping from the input space to the discrete space of the positions of the neurons is performed. The first problem may be solved in an efficient way by forming the prototypes directly in the input space. there are many methods in statistical decision theory that are helpful in this respect, one of the best known is Learning Vector Quantization (LVQ) [5] where these prototypes are called the codebook vectors.

The second process, self-organizing mapping to the target space, is more interesting. Full preservation of metric or even topological relations of the high dimensional space by a low-dimensional map is impossible, as cartographers know very well. The distortions of topology in SOM were analyzed very recently [9]. SOM tries to preserve "topographic" relations, i.e. something more than topological and less than metric relations.

It is convenient to look at the SOM from the point of view of the input space, i.e. space of the parameters in which the data is defined. The weights of the neurons in the Kohonen layer are at first connected with the random points in the parameter space. Iterative procedure (2.1) pulls these weights in the direction of the main data clusters. The weights of all neurons in the neighborhood of the winning one are also pulled towards the same cluster. The neighboring cluster will compete for the weights of the closely laying neurons. The dynamics of this process is very interesting [10]. Although the ability of SOM algorithm to form ordered maps has been demonstrated many times by numerical simulations unfortunately it is hard to say anything general about the self-organization process.

The problem of map formation on a two-dimensional surface is easily solved if the original data is already two-dimensional. The weights of the network may than be directly set and the topographical relations present in the data is directly represented in the two-dimensional layer of neurons. Each cluster in the parameter space is represented by some neuron (i,j) with the weights $(W_1^{ij}, \ldots W_N^{ij})$ close to the coordinates of the cluster in this space.

3 Alternative ways of topographic mappings

There are a few ways one can use to map directly the N-dimensional data to one or two dimensions. First, one may try a direct projection. To preserve the topographic relations of the N-dimensional space one may try to find a point of view (like a photographer trying to find a point of view from which the objects are not blocking each other) which maximizes the distance of the projected objects in the target space. Consider a projection of k objects on a line; designating the positions of the projected points as x_i one may maximize:

$$D_0(\mathbf{x}) = \sum_{i>j}^{k} (x_i - x_j)^2 \tag{3.1}$$

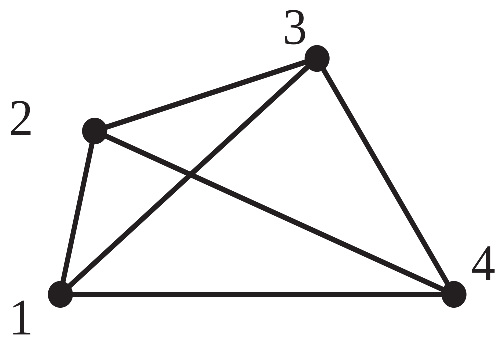
Maximization of this expression corresponds to finding the position from which the object separation is largest. Unfortunately such a procedure will not give interesting mappings, loosing topographical relations present in the input data. It is enough to consider the mapping from 2-dimensional (2-D) space on a line: the only free parameter is then the rotation of the 2-D coordinate system. The expression for the $D_0(\mathbf{x})$ depends then on the angle of rotation. It is easy to compute the maximum of such an expression and to find out that there is a strong tendency to collapse several input points into one; for example if a T-shaped object is projected the maximum will be reached for the longer section of T being parallel to the line on which the projection is made, while all the point of the shorter section are collapsed into one. Intuitively we see that such mapping does not preserve topographical relations of the input space data. A quantitative measure of this preservation is necessary.

This method fails because the function $D_0(\mathbf{x})$ is defined only in the projected space. A much better criterion for the preservation of relative distances is obtained if the function $D(\mathbf{x})$ will compare distances in the input and in the target space. Let us designate the distances in the N-dimensional input space as R_{ij} and in the n-dimensional ($n < N$) target space as r_{ij}. Information is contained in the relative distances of the objects, therefore we should minimize the following expression:

$$D_1(\mathbf{r}) = \sum_{i>j}^{k} (R_{ij} - r_{ij})^2 \tag{3.2}$$

Here R_{ij} are fixed, given numbers and r_{ij} are unknown distances. This function is positively defined and is equal to zero if the same distances between the objects in the target as in the input space are possible. In general we cannot hope that this will be the case. The minimum of this function corresponds to the metric relations of the k objects, mapped to the n-dimensional target space, resembling as closely as possible the original relations in the input space. Should it happen that the data lies on a plane embedded in multidimensional space mapping to two dimensions should recognize this fact and restore perfectly metric relations between all objects.

The main problem in minimizing this function is that in the lower-dimensional space not all distances r_{ij} are independent. It is easily taken into account if the mapping is done to the one-dimensional space, since then one has to deal with the linear dependence. In the simplest nontrivial case if 4 points are mapped from 3-D space to 2-D space the distance r_{34} depends on the 5 other distances, r_{12} r_{13}, r_{14}, r_{23}, r_{24} in a highly nonlinear way.



The explicit conditions of this dependence are rather complicated and involve such factor like:

$$r_{34} = r_{24}\frac{\sin\alpha_{423}}{\sin\left[\arctan\left(\frac{r_{24}-r_{23}}{r_{24}+r_{23}}\cot\frac{\alpha_{423}}{2}\right)+\frac{\pi-\alpha_{423}}{2}\right]} \tag{3.3}$$

with another complicated formula to express the angle α_{423} via distances. The $D_1(\mathbf{r})$ function becomes highly nonlinear and is hard to write even in the two-dimensional target space case.

Fortunately there is an elegant and universal solution to the problem of topographical mapping based on the minimization of the (3.2) function. Let's count first how many free parameters do we have in spaces of various dimensions. For k objects in 1-dimension there are k-1 free parameters; in 2-dimensions there are 2k-3 free parameters (an arbitrary origin and orientation angle of the coordinate system reduces the number of free parameters on 3); for 3-dimensional case 3k-5 (two rotation angles and one point fixed as the origin) and in general for N-dimensional case Nk-N-(N-1) = N(k-2)+1 free parameters. Reducing the number of free parameters while mapping from N to n dimensions is easily done if instead of distances in the target space Cartesian coordinates are used.

Designating by $x^{(l)}$ coordinates in the target space and fixing the origin and the rotation angles of the coordinate system by setting all n components of x_1 and n-1 components of x_2 to 0 we automatically reduce the number of free parameters. Thus the function to be minimized is:

$$D_1(\mathbf{x}) = \sum_{i>j}^{k}\left(R_{ij} - \sqrt{\sum_{l=1}^{n}\left(x_i^{(l)} - x_j^{(l)}\right)^2}\right)^2 \tag{3.4}$$

subject to the simple auxiliary conditions for x_1 and x_2 described above. The nonlinearity is here much simpler. Differentiating this function over $x_i^{(m)}$ for i = 3..k and m=1..n and for i=2 and m=n we obtain n(k-2)+1 equations for free parameters $x_i^{(m)}$

$$\sum_{j(\neq i)}^{k} \Delta x_{ij}^{(m)}\left(\frac{R_{ij}}{r_{ij}}-1\right)=0 \qquad \begin{matrix} i=2, m=n \\ i\geq 3, m=1..n \end{matrix} \tag{3.5}$$

where

$$\Delta x_{ij}^{(m)} = x_i^{(m)} - x_j^{(m)};\; r_{ij}=\sqrt{\sum_{l=1}^{n}\left(x_i^{(l)}-x_j^{(l)}\right)^2} \tag{3.6}$$

To simplify the problem even further another positively defined function may be used:

$$D_2(\mathbf{x})=\sum_{i>j}^{k}\left(R_{ij}^2-\sum_{l=1}^{n}\left(x_i^{(l)}-x_j^{(l)}\right)^2\right)^2 \tag{3.7}$$

with the same auxiliary conditions. Direct minimization gives the following set of nonlinear equations to be solved:

$$\sum_{j\neq i}^{k}\left(x_i^{(m)}-x_j^{(m)}\right)^3+\sum_{j\neq i}^{k}\left(x_i^{(m)}-x_j^{(m)}\right)\sum_{l\neq m}^{n}\left(x_i^{(l)}-x_j^{(l)}\right)^2-\sum_{j\neq i}^{k}R_{ij}^2\left(x_i^{(m)}-x_j^{(m)}\right)=0 \tag{3.8}$$

The solution of this system of equation gives the lowest value of $D_2(\mathbf{x})$ function. Unfortunately (3.5) and (3.8) are large system of nonlinear equations and there are no good methods for solving nonlinear systems of equations. The Newton-Raphson method works well if we have a good initial guess; however, this is precisely what we are looking for since high accuracy of the solution is not required in this case. Other options include globally convergent Newton method and Broydens method [11].

Perhaps more promising method of finding good topographical maps should be based on the direct minimization of the functions $D_1(\mathbf{x})$ or $D_2(\mathbf{x})$ via the gradient descent technique. Such minimization leads to the problem of multiple minima, since the number of independent parameters x_i may be rather large.

4 Summary

It is interesting to consider the relation of the Kohonen's algorithm to the simulated annealing procedure [12] for the minimization of multivariable functions. The annealing procedure uses a parameter, called temperature to stress the analogy with thermodynamical processes, that controls the size of the changes of the parameters while sampling the configuration space of these parameters. The value of the minimized function $D(\mathbf{x})$ changes on ΔE for a new set of parameters. All parameters for which $\Delta E<0$ are accepted while those for which $\Delta E>0$ are accepted with the probability $P(\Delta E)=exp(-\Delta E/T)$. For a given temperature enough configurations are sampled to reach the thermodynamic equilibrium. The parameters giving the lowest value of $D(\mathbf{x})$ are then taken, lower temperature selected and the procedure repeated. In this way local minima are avoided. In the Kohonen's SOM algorithm the cost function is not explicitly introduced. Assuming that the results should be similar to those obtained from minimization of $D_1(\mathbf{x})$ the SOM algorithm performs an indirect minimization, adjusting the parameters locally around the mapping of each new data point. It should be worthwhile to investigate this kind of minimization procedure in details.

We are making numerical experiments to determine how do the maps obtained from minimization of $D_1(\mathbf{x})$ and $D_2(\mathbf{x})$ differ. Kohonen [4] showed that the self-organizing mapping from two to one dimension lead to Peano curves and we would like to see if the same result is obtained from minimization principle. Even though the problem of topographical mapping is difficult to solve at least it has been given in this paper solid mathematical foundations and a measure was introduced, allowing for quantitative comparison of different topographical mappings.

References

[1] J.A. Anderson, and E. Rosenfeld, (Eds). Neurocomputing: Foundations of Research. (The MIT Press: Cambridge, MA.(1988)); J.A. Anderson, A. Pellionisz, and E. Rosenfeld, (Eds). Neurocomputing 2: Directions for Research. (The MIT Press: Cambridge, MA. 1990)

[2] T. Kohonen, Biol. Cybern. 43 (1982) 59-69

[3] T. Kohonen, An Introduction to Neural Computing. Neural Networks 1 (1988) 3-16

[4] T. Kohonen, Proceedings of the IEEE 78 (1990) 1464

[5] T. Kohonen, Self-organization and Associative Memory. (Springer-Verlag, New York, 1984, 2nd Edition: 1988; 3rd edition: 1989).

[6] S.B. Udin, W.J. Fawcet, Annual Review of Neuroscience 11 (1988) 289-327

[7] D.J. Willshaw, C. von der Malsburg, Proc. Roy. Soc. London B 194 (1976) 431-445

[8] S-i. Amari, Bulletin of Math. Biology 42 (1980) 339-364

[9] Bibliography of more than 1200 papers related to the Self-Organizing Map is stored in the Internet anonymous ftp node at the address: colchea.hut.fi, in the catalog /pub/ref

[10] X. Li, J. Gasteiger, J. Zupan, Biol. Cybern. 70 (1993) 189-198; J. Zhang, Neural Computation 3 (1991) 54-66; P. Tavan, H. Grubmüller, H. Kühnel, Biological Cybernetics 64 (1990) 95-105

[11] A.C.C. Coolen, L.G.V.M. Lenders, J. Phys. A 25 (1992) 2593-2606;

[12] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes (Cambridge Univ. Press, 2nd ed, 1992)

[13] S. Kirkpatric, C.D. Gellat Jr, M.P. Vecchi, Science 220 (1983) 671-680