10.1098/rsta.2003.1192

THE ROYAL SOCIETY

Self-organized maps of sensory events

By Teuvo Kohonen

Helsinki University of Technology, Neural Networks Research Centre, PO Box 5400, 02015 HUT, Finland

Published online 6 May 2003

Over the years, many divergent meanings have been associated with the term 'self-organization', e.g. automatic creation of structured systems and optimization of parameters in adaptive learning. In this paper, we shall discuss a special type of data-driven self-organization, namely, automatic formation of ordered, compressed representations of sensory events. Such ordered and organized representations of an organism's experiences and environment exist in the nervous systems, where specific feature-sensitive information-processing functions are usually associated with these representations. As a matter of fact, three types of neuronal organization called 'brain maps' can be distinguished: sets of feature-sensitive cells, ordered projections between neuronal layers, and ordered maps of abstract features, respectively. The latter are most intriguing as they may also reflect quite abstract properties of the input data in an orderly fashion. It is proposed that such 'maps' are learned in a process that involves competition between sets of neural cells on common input data, and sensitization or tuning of the most strongly responding cells and their local neighbours to this input. While serving as a model for brain maps, the 'self-organizing map' principle has been used as an analytical tool in exploratory data analysis. In the latter, it has had practical applications ranging from industrial process control to marketing analyses, and from linguistics to bioinformatics.

Keywords: brain model; data mining; data-driven self-organization; representation of sensory events; neural network; self-organizing map

1. Introduction

In this paper we shall discuss data-driven self-organization, in particular the emergence of ordered and structured representations and corresponding detector functions of sensory events.

The brain is a highly structured system of information-processing functions. Many of these functions have been developed in phylogenesis over many generations, and passed on to the next generations by the genes. However, many functions are also learned directly, postnatally, by adaptation to sensory experiences and other occurrences. A significant part of this adaptation involves self-organization.

What we call 'artificial neural networks' are computational principles believed to underlie the operation of the biological neural networks. They were initially introduced as models for what were believed to be the neural circuits in the brain. First

One contribution of 18 to a Theme 'Self-organization: the quest for the origin and evolution of structure'.

of all, the operations of the cells of the network are not fixed as in digital computers, but *adaptable*: when the cells transform signals, a required transformation can be learned, with the help of a teacher or even autonomously from the signals. Second, unlike in the digital circuits, the same signals are propagated through the network along many parallel paths, and thus their transformation depends on a great many cells. On the other hand, many different transformations can be implemented by the same network. This feature is called the *distributedness* of the operations, and in biological information processing it provides for robustness. Third, it is characteristic of the more advanced artificial neural networks that their computing functions and/or structures are created by *self-organization*, in the same way as the information processing ability emerges in the biological brain during its adaptation to the environment.

These three properties, namely adaptability, distributedness and self-organization, are useful and even necessary when one has to deal with natural information (such as speech, pictures, natural text) or process data, that is, information of an unknown nature. The natural signals are usually very noisy and unstable, but at the same time the signals depend on each other in many ways. One may imagine an industrial process, the internal conditions of which may change in an unpredictable way. Therefore, only the most central information should be extracted, and the most robust measuring and analysis techniques be applied. The situations in the industrial processes very often resemble those of the biological organisms, for which the nervous systems have evolved. The more advanced artificial neural networks are similarly able to evolve and be adapted to difficult environments and conditions. What we aim at is a *model of experimental data*, not of the process variables.

It is amazing that although masses of experimental data and observations convincingly demonstrate the existence of a meaningful *spatial order and organization* of the brain functions, and this order seems to be ubiquitous in the nervous systems, the majority of works on artificial neural networks do not take it into account in any way. This order is useful for many different reasons, such as the following examples.

(i) By bringing mutually relevant functions close to each other spatially, the wiring can be minimized.

(ii) If the responses are spatially segregated (although the underlying network may be distributed), there will be minimal 'crosstalk' between the functions, and the brain architecture can be made more logical and robust.

(iii) It seems that for effective representation and processing of knowledge one anyway needs some kind of metric 'conceptual space' (Gärdenfors 2000) to facilitate the emergence of natural concepts.

If a logic concept were defined only in terms of its attributes, as made in the classical philosophy, one would run into the 'property inheritance' problem (Fahlman 1981), because a concept should contain all the attributes of its superordinates; but where could they be stored? It would be more natural that a concept is represented in terms of its relations to the most relevant concepts that are located in the neighbourhood in the ordered 'representation space'.

The first work in which ordered representations, namely ordered orientation-specific neural cells, were produced in simulations was due to von der Malsburg

(1973). Later, topographically ordered anatomical projections between neuronal layers were analysed, for example, by Amari (1980) and many others.

Before proceeding further, it will be necessary to emphasize that one has to distinguish the following three different kinds of 'maps' in the brain.

(i) Feature-specific cells, i.e. neurons or neuron groups that elicit an active response upon presentation of a patterned object to the sensory organs, such as the silhouette of a hand, a human face, or an acoustic tune. The spatial locations of such cells then do not necessarily correlate with the feature values.

(ii) Anatomical projection of some receptive surface onto, say, the cortex. Examples are the areas in the visual cortex that constitute transformed images of the retina, and the somatosensory cortex, which forms an ordered image of the body.

(iii) Ordered maps of some abstract features, for which no receptive surfaces exist. An example is the colour map in the visual area V4.

My own contribution to this theory was launched when I tried to define the self-organizing process in its most fundamental and abstract form around 1981–1982 (Kohonen 1981, 1982*a*, *b*). I also discovered that topographically or topologically ordered 'maps' of very abstract features can be made to emerge automatically. The formation of many abstract feature maps known from neurobiology has been simulated by this principle or algorithm called the 'self-organizing map' (SOM), and the SOM algorithm has also been introduced to a great number of applications in various fields of technology and science, ranging from industrial process analysis to knowledge discovery in databases, from linguistics to bioinformatics, and many others.

2. The basic SOM

The SOM is an unsupervised-learning (adaptive) algorithm in the neural-network category. It means that the representations of information are determined automatically from the metric relationships between the data items; no 'teacher' is needed, i.e. no input–output relations are defined *a priori*.

The SOM forms a *nonlinear projection* from a high-dimensional data manifold onto a regular, usually two-dimensional, grid. Thereby it carries out clustering, visualization and abstraction of the multidimensional input data.

The SOM thus consists of a two-dimensional regular grid of nodes. A *model* of some observation is associated with each node. Figure 1 delineates a two-dimensional hexagonally arranged grid intended to represent short-time spectra of natural speech (Finnish).

The SOM algorithm computes the collection of the models so that it optimally describes the domain of (discrete or continuously distributed) observations. The models are automatically organized in a meaningful two-dimensional order so that similar models become closer to each other in the grid than the more dissimilar ones. In this sense the SOM is a similarity graph and a clustering diagram. Its computation is a non-parametric, recursive regression process.

The self-organizing process may be realized in any set of elements, illustrated schematically in figure 2, where only a few basic operational conditions are assumed.

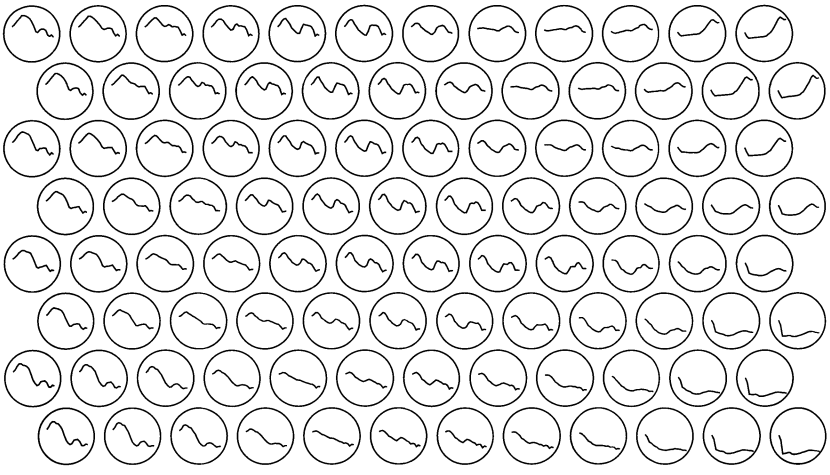

Figure 1. An SOM grid. A model of a short-time spectrum of natural speech (Finnish) is associated with each node.

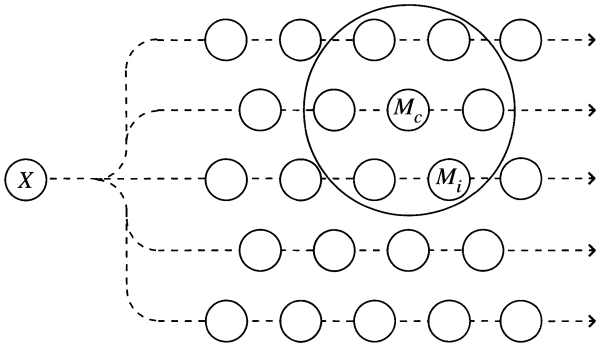

Figure 2. A self-organizing model set. An input message X is broadcast to a set of models M_i, of which M_c best matches X. All models that lie in the vicinity of M_c (larger circle) improve their matching with X. Note that the location of M_c differs from one message to another.

For simplicity, let the elements (e.g. single neurons or groups of closely cooperating neurons) form a regular planar array and let each element represent a set of numerical values M_i, which we call a *model*. These values may correspond to some parameters of the neuronal system, and in contemporary computational neuroscience it is customary to identify them with synaptic efficacies. We further assume that each model is modified by the messages the element receives.

Let there exist some mechanism by which an ingoing message X, a set of parallel signal values, can be compared with all models M_i. In brain theory it is customary to speak of 'competition' between the elements, when they are stimulated by common input, and the element, whose parameters are fittest to this input is activated most. This element is called the 'winner' if it succeeds in suppressing the activity in the neighbouring neurons by, for example, lateral inhibition. The 'winner' model is denoted by M_c. Neural circuits that implement the 'winner take all' function have been suggested, for example, by Kohonen (1993) and Kaski & Kohonen (1994). Another requirement for self-organization is that the models shall be modified only

in the *local vicinity* of the winner(s) and, after that, all the modified models shall resemble the prevailing message better than before.

When the models in the neighbourhood of the winner simultaneously start to resemble the prevailing message X better, they also tend to become more similar mutually, i.e. the differences between all the models in the neighbourhood of M_c are smoothed. Different messages at different times affect separate parts of the set of models, and thus the models M_i, after many learning steps, start to acquire values that relate to each other smoothly over the whole array, in the same way as the original messages X in the 'signal space' do; in other words, maps related topologically to the sensory events start to emerge, as can be proven mathematically (Cottrell *et al.* 1997).

These three subprocesses—broadcasting of the input, selection of the winner and adaptation of the models in the spatial neighbourhood of the winner—seem to be sufficient, in the general case, to define a self-organization process that then results in the emergence of the topographically organized 'maps'. It has to be emphasized, however, that the mathematical theory is extremely difficult and its development is still in progress. Proofs exist only for the simplest cases (Cottrell *et al.* 1997).

Simulations of self-organization with idealized networks can be implemented by many mathematical rules. The particular rules selected for this work constitute a compromise between effectivity of self-organization and biological realizability.

A simple representation of message X is the data vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, i.e. a list of numbers that represent the signal values. The model M_i is similarly represented by the 'model vector' $\boldsymbol{m}_i = (m_{i1}, m_{i2}, \ldots, m_{in})$ of the same dimensionality as $\boldsymbol{x}$. The degree of matching between $\boldsymbol{x}$ and $\boldsymbol{m}_i$, or actually that of mismatch, is expressible, e.g. as the Euclidean distance or difference between $\boldsymbol{x}$ and $\boldsymbol{m}_i$. The winner, or the best-fitting model $\boldsymbol{m}_c$, shall then be defined by equation (2.1), which describes the comparison process

$$c = \arg\min_i \{\|\boldsymbol{x} - \boldsymbol{m}_i\|\}. \tag{2.1}$$

A message $\boldsymbol{x}(t)$ at time t will modify the values of models $\boldsymbol{m}_i(t)$ to new values $\boldsymbol{m}_i(t+1)$ one time-step later. The SOM algorithm applied in this article modifies the models in the neighbourhood of the winner so that the difference between them and the input message increases.

Let the neighbourhood of the 'winner' be described by the *neighbourhood function* h_{ci}, which has its maximum for the winner (when $i = c$). The value of h_{ci} decreases with increasing distance of neuron i from the winner c in the array of neurons; h_{ci} may also change with time. In simulations, $h_{ci}(t)$ can be given different mathematical forms.

Equation (2.2) shows how the modified values of the 'winning model' and its neighbouring models are assumed to depend on the messages they receive: the larger the difference between the message at time t and the model $\boldsymbol{m}_i(t)$, the larger is the change towards values of the message; however, $h_{ci}(t)$ restricts the changes into the vicinity of the winner, namely,

$$\boldsymbol{m}_i(t+1) = \boldsymbol{m}_i(t) + h_{ci}(t)\{\boldsymbol{x}(t) - \boldsymbol{m}_i(t)\}. \tag{2.2}$$

When the SOM was introduced for the first time, it was shown that the initial values of its models in the learning process can be selected as random vectors. However, it is not necessary to prove every time that the SOM algorithm will order the

models. Therefore, we now recommend that the initial models be chosen as a regular array of vectors on the hyperplane spanned by the two largest principal components of input data. The learning process then becomes smoother and faster.

3. The generalized SOM algorithm

In the previous section we exemplified the SOM algorithm with vectorial data and models. The following scheme, however, applies to both vectorial and non-vectorial (e.g. symbolic) input data, as long as some similarity or distance measure between the data items is definable.

In order that data-driven self-organization be most effective, the following two partial processes should always be implementable in as pure a form as possible.

(i) Find that cell in the network that matches best with the present input (in the sense of some criterion).

(ii) Modify this cell and its neighbours in the network to improve their matches with the present input.

These partial processes will now be combined into the following batch computation scheme. Consider figure 3, where a two-dimensionally ordered array of nodes, each one having a general model $\boldsymbol{m}_i$ associated with it, is shown. (It is not necessary to regard the $\boldsymbol{m}_i$ here as real vectors, as earlier.) The initial values of the $\boldsymbol{m}_i$ may be selected as random, preferably from the domain of the input samples. Then consider a list of input samples $\boldsymbol{x}(t)$, where t is an integer-valued index. Compare each $\boldsymbol{x}(t)$ with all the $\boldsymbol{m}_i$ and list each $\boldsymbol{x}(t)$ into a sublist associated with that node, the model of which is closest to $\boldsymbol{x}(t)$ relating to some general distance measure. When all the $\boldsymbol{x}(t)$ have been distributed into the respective sublists in this way, we consider the neighbourhood set N_i around model $\boldsymbol{m}_i$. Here N_i consists of all nodes up to a certain radius in the grid from node i. In the union of all sublists in N_i, the next task is to find the 'middlemost' sample $\bar{\boldsymbol{x}}_i$, defined as the sample that has the smallest sum of distances from all the samples $\boldsymbol{x}(t)$ in N_i. This sample $\bar{\boldsymbol{x}}_i$ is now called the *generalized median* in the union of the sublists. The name 'median' may be justified here, since it is easy to show that if the $\boldsymbol{x}(t)$ are real scalars, and the distance between two samples were the absolute value of their difference, then the generalized median would be identical to the arithmetic median of the samples.

If $\bar{\boldsymbol{x}}_i$ is restricted to being one of the samples $\boldsymbol{x}(t)$, then it may be further proper to call it the *generalized set median*; notwithstanding the fact that it may be possible to find another item $\bar{\boldsymbol{x}}'_i$ that is not one of the $\boldsymbol{x}(t)$ but has an even smaller sum of distances from the $\boldsymbol{x}(t)$ in N_i. For simplicity, however, we shall also then call $\bar{\boldsymbol{x}}'_i$ here the generalized median over the union of the sublists. Notice too that for the Euclidean vectors the generalized median is equal to their arithmetic mean if we look for an arbitrary Euclidean vector that has the smallest sum of squares of the Euclidean distances from all the samples $\boldsymbol{x}(t)$ in the union of the sublists.

In the same way as in the traditional SOM algorithms for vectorial variables, one can also use weights in forming the smallest sum of distances. Consider that i is the index of the node around which N_i is centred, and let k be another node in N_i. Then the weighting can be made by the factor h_{ik} that is similar to the neighbourhood function in the traditional SOM. The next step in the process is to form $\bar{\boldsymbol{x}}_i$ or $\bar{\boldsymbol{x}}'_i$

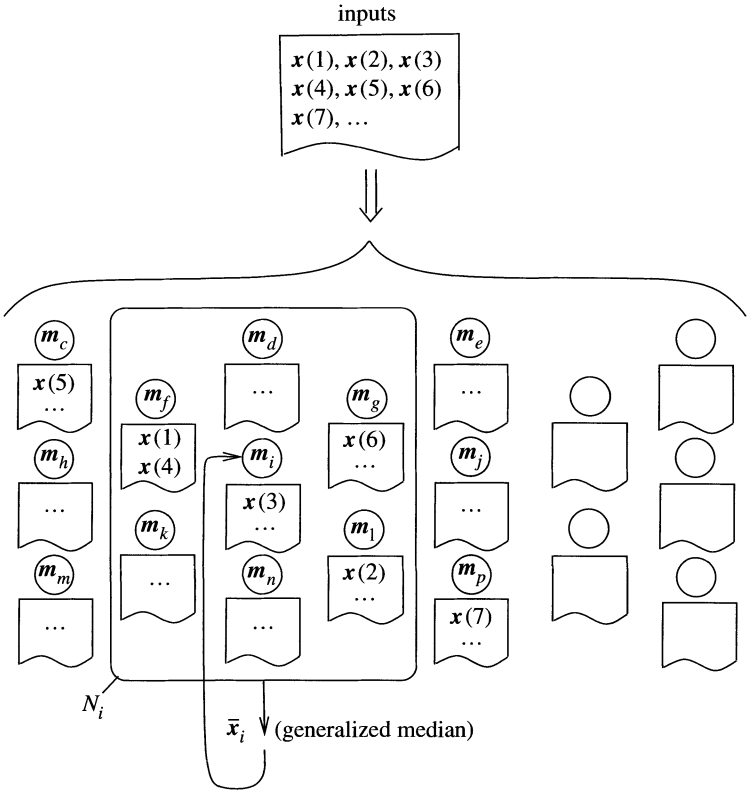

Figure 3. Illustration of the batch process in which the input samples are distributed into sublists under the best-matching models, and then new models are determined as generalized medians of the sublists over the neighbourhoods N_i.

for each node in the above manner, always considering the neighbourhood set N_i around each node i, and to replace each old value of $\boldsymbol{m}_i$ by $\bar{\boldsymbol{x}}_i$ or $\bar{\boldsymbol{x}}'_i$, respectively. This subprocess is similar as in the Batch Map algorithm (Kohonen 1992).

The above procedure shall be iterated until the $\boldsymbol{m}_i$ can be regarded as stationary: in other words, in the next batch step of computation, the $\boldsymbol{x}(t)$ are again distributed into the sublists, and the new $\bar{\boldsymbol{x}}_i$ or $\bar{\boldsymbol{x}}'_i$ are made to replace the $\boldsymbol{m}_i$, and so on. The convergence, however, has not been proved for a general distance measure; as a matter of fact, only one related theoretical treatment for vectorial data exists so far (Cheng 1997).

Notice also that the set of input samples $\boldsymbol{x}(t)$ used in the process need not be fixed: from all the available inputs one can randomly pick up a smaller set of samples, which are then stochastically different at each cycle of iteration. As long as the stochastic process that defines the $\boldsymbol{x}(t)$ is ergodic, the $\boldsymbol{m}_i$ are expected to converge.

The following types of models have already been used in SOM architectures (Kohonen 2001):

(i) vectorial models with various distance measures;

(ii) linear subspaces defined by sets of basis vectors;

(iii) operators and parametrized filters such as the LPC estimator;

(iv) symbol strings;

(v) fuzzy set expressions;

(vi) genetic-algorithm parameters.

4. Implementation of SOMs in the real brain

Maps as clearly organized as those observed in various ideal simulation studies may not exist in the brain. Pre-processing in neural realms is already much more complex than that used in simulations. However, the mammalian brain is known to support several feature maps, either orderly representations of the receptive surfaces on skin, retina or cochlea, or purely abstract maps. The acoustic maps are assumed to be associated with sounds to which the organism is most frequently exposed. Tonotopic maps, i.e. maps that are organized according to frequency and sound intensity, known to exist both in auditory pathways and in several cortical auditory fields.

Essentially every level of the nervous system exhibits plasticity under certain circumstances, and thus feature maps can also be expected at all levels. Furthermore, even the 'hardwired' maps are known to depend on the sensory experience and they thus evidently are, at least to some extent, the result of self-organization.

Materialization of the SOM principle in the nervous system would first require a mechanism that distributes essentially the same, or strongly correlated, information to the neurons or neuron groups of a certain region. Spatial diameters of the maps are then determined by the longest distances reachable by common (or highly correlated) input in the neuronal layer. The SOM principle might work well in the thalamocortical system, where the diameter of the map would be primarily determined by the size of the thalamocortical axonal arbours, reaching maximally 1–2 mm (Jones & Peters 1984), and by the ramifications of the apical dendritic trees of the receiving cortical cells, with diameters less than 3 mm. Thus a typical cortical map that describes a single type of abstract feature would be maximally 5 mm wide.

Second, one needs a mechanism that 'selects' the 'winner' neurons, i.e. the centre around which adaptation shall take place. Lateral interconnectivity with excitatory and inhibitory connections may play a central role in this selection, resulting in enhanced discharge rates at a place where the original excitation was high, and suppression of activity elsewhere.

Third, restriction of learning to the neighbourhood of the winner(s) may simply follow the clustering of triggering activity, but one can also assume that some kind of chemical 'learning factor' that controls modification locally, but does not activate the neurons, is emanating from the active neurons.

5. Applications

Over 5000 scientific publications on the SOM have been written. For a documented list of most of them, see Kaski *et al.* (1998). The main application areas are

(i) statistical analysis at large, in particular data mining and knowledge discovery in databases;

(ii) analysis and control of industrial processes and machines;

(iii) new methods in telecommunications, especially optimization of telephone traffic and demodulation of digital signals;

(iv) medical and biological applications.

The SOM has been used in Finnish forest industries for control purposes since 1985, and for numerous industrial applications over the world since then. Recently, for the continuous casting of steel, an on-line monitor based on the SOM has been developed (Alhoniemi *et al.* 1999).

Practical applications have been introduced to finance, for instance bankruptcy analysis, profiling of customers, and analysis of macroeconomic systems (Deboeck & Kohonen 1998). A promising area is in real estate business. The new Finnish forest taxation legislation in 1992 was based on segmentation results obtained by the SOM.

Of the numerous applications from very different scientific fields one may mention the analysis of the Hubble Space Telescope data, where a new quantitative classification of thousands of galaxies has been developed (Naim *et al.* 1997).

Another novel example is from criminology, where an SOM-based system for computer-aided tracking of homicides and sexual assaults has been developed by the Battelle Pacific Northwest Division, in cooperation with the Attorney General of the State of Washington.

The biggest SOM, with over one million models of dimensionality 500, has been developed for document organization, namely for the similarity diagram of about seven million patent abstracts (Kohonen *et al.* 2000). This text corpus is over 20 times as big as all the 34 parts of the *Encyclopaedia Britannica* together. Each document is described by the collection of the words it contains. Five hundred statistical indicators of the high-dimensional word histograms, with a vocabulary of about 50 000, are used as real 500-dimensional input vectors to the SOM. Standard browsing tools are used to display and search for the documents of interest. The two-dimensional order in the document maps makes it possible to find additional relevant information, after the starting point in the document collection has been defined (Kohonen *et al.* 2000).

About 20 monographs, textbooks and edited volumes that concentrate on the SOM have appeared: in addition to Kohonen (2001) and Ritter *et al.* (1992) one may mention Deboeck & Kohonen (1998), Miikkulainen (1993), Tokutaka *et al.* (1999), van Hulle (2000), Oja & Kaski (1999), Allinson *et al.* (2001), Obermayer & Sejnowski (2001) and Seiffert & Jain (2002). A special issue of *Neurocomputing* has been dedicated to the SOM (Sánchez 1998).

6. Conclusion

While serving as a model for brain maps, the 'self-organizing map' principle has been used as an analytical tool in exploratory data analysis. In the latter it has had practical applications ranging from industrial process control to marketing analyses, and from linguistics to bioinformatics.

References

Alhoniemi, E., Hollmén, J., Simula, O. & Vesanto, J. 1999 Process monitoring and modeling using the self-organizing map. *Integrat. Computer-Aided Engng* **6**, 3–14.

Allinson, N., Yin, H., Allinson, L. & Slack, J. (eds) 2001 *Advances in self-organising maps.* Springer.

Amari, S.-I. 1980 Topographic organization of nerve fields. *Bull. Math. Biol.* **42**, 339–364.

Cheng, Y. 1997 Convergence and ordering of Kohonen's batch map. *Neural Comput.* **9**, 1667–1676.

Cottrell, M., Fort, J. C. & Pagès, G. 1997 Theoretical aspects of the SOM. In *Proc. WSOM'97: Workshop on Self-Organizing Maps, Espoo, Finland, 4–6 June 1997*, pp. 246–267. Helsinki University of Technology.

Deboeck, G. & Kohonen, T. (eds) 1998 *Visual exploration in finance with self-organizing maps.* Springer.

Fahlman, S. 1981 Representing implicit knowledge. In *Parallel models of associative memory* (ed. G. E. Hinton & J. A. Anderson), pp. 145–159. Hillsdale, NJ: Lawrence Erlbaum Association.

Gärdenfors, P. 2000 Three levels of inductive inference. In *Logic, methodology and philosophy of science IX* (ed. D. Prawitz, B. Skyrms & D. Westerståhl), pp. 427–449. Elsevier.

Jones, E. G. & Peters, A. (eds) 1984 *Cerebral Cortex*, vol. 2. New York: Plenum Press.

Kaski, S. & Kohonen, T. 1994 Winner-take-all networks for physiological models of competitive learning. *Neural Netw.* **7**, 973–984.

Kaski, S., Kangas, J. & Kohonen, T. 1998 Bibliography of self-organizing map (SOM) papers 1981–1997. *Neural Comput. Surv.* **1**, 1–176.

Kohonen, T. 1981 Automatic formation of topological maps of patterns in a self-organizing system. In *Proc. 2nd Scandinavian Conf. Image Analysis, Espoo, Finland, 15–17 June 1981*, pp. 214–220. Helsinki: Pattern Recognition Society of Finland.

Kohonen, T. 1982*a* Self-organized formation of topologically correct feature maps. *Biol. Cybern.* **43**, 59–69.

Kohonen, T. 1982*b* Clustering, taxonomy, and topological maps of patterns. In *6th Int. Conf. Pattern Recognition, Munich, Germany, 19–22 October 1982*, pp. 114–128. Washington, DC: IEEE Computer Society Press.

Kohonen, T. 1992 New developments of learning vector quantization and the self-organizing map. In *Proc. SYNAPSE'92, Symp. Neural Networks: Alliances and Perspectives, Senri, Osaka, Japan, 24–26 June 1992.* Osaka: Senri Hankyu.

Kohonen, T. 1993 Physiological interpretation of the self-organizing map algorithm. *Neural Netw.* **6**, 895–905.

Kohonen, T. 2001 *Self-organizing maps*, 3rd edn. Springer.

Kohonen, T., Kaski, S., Lagus, K., Salojärvi, J., Honkela, J., Paatero, V. & Saarela, A. 2000 Self-organization of a massive document collection. *IEEE Trans. Neural Networks* **11**, 574–585.

Miikkulainen, R. 1993 *Subsymbolic natural language processing: an integrated model of scripts, lexicon and memory.* Cambridge, MA: MIT Press.

Naim, A., Ratnatunga, K. U. & Griffiths, R. E. 1997 Galaxy morphology without classification: self-organizing maps. *Astrophys. J. Suppl.* **111**, 357–367.

Obermayer, K. & Sejnowski, T. J. (eds) 2001 *Self-organizing map formation: foundations of neural computation.* Cambridge, MA: MIT Press.

Oja, E. & Kaski, S. (eds) 1999 *Kohonen maps.* Amsterdam: Elsevier.

Ritter, H., Martinetz, T. & Schulten, K. 1992 *Neural computation and self-organizing maps: an introduction.* Reading, MA: Addison-Wesley.

Sánchez, A. V. D. (ed.) 1998 Special issue on self-organizing maps. *Neurocomputing* **21**.

Seiffert, U. & Jain, L. C. (eds) 2002 *Self-organizing neural networks: recent advances and applications.* Heidelberg: Physica.

Tokutaka, H., Kishida, S. & Fujimura, K. 1999 *Applications of self-organizing maps.* Tokyo: Kaibundo. (In Japanese.)

von der Malsburg, C. 1973 Self-organization of orientation sensitive cells in the striate cortex. *Kybernetik* **14**, 85–100.

van Hulle, M. M. 2000 *Faithful representations and topographic maps: from distortion- to information-based self-organization.* Wiley.