ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 349-354

SOM Algorithms and their Stability Consideration

Youichi Kobuchi[1,2] and Masataka Tanoue[2]

[1]Microelectronics Lab. Université Catholique de Louvain, Belgium
[2]Department of Electronics and Informatics
Faculty of Science and Technology
Ryukoku University, Seta, Japan

A two layered feedforward neural network is considered as Kohonen's dot-product type SOM model which defines a winner function *f* from an input pattern set into an output unit set. It defines pattern classifiers through step by step self-organization. What kinds of classifiers they can ultimately be and how they are attained asymptotically are the basic problems to be solved. Our main result is that the property of being topographic can be preserved by appropriately choosing learning rates and hence the winner function becomes stable in this case.

1. Introduction

Consider a set of vectors which are eventually to be classified according to some similarity measure. We treat an unsupervised learning process to yield such topology preserving classification in its broad sense. Typical networks which realize such tasks have been proposed by Kohonen and others. (See Kohonen 1995, Hulle 2000, and the references therein.) In the case of the set of vectors of binary components, we discussed relations between topology preservation and mapping stability under certain learning rules. (Sakamoto and Kobuchi 2000, 2002; Sakamoto, Seki, and Kobuchi 2003) We continue the works to analyze Kohonen's SOM and relate the stability of a mapping with its topology preservation.

Kohonen's SOM formation algorithms comprise two basic stages.

The first stage is a winner selection or competition: When an input vector is applied to a network, each output unit computes corresponding output value. Then a winner unit is determined depending on the output values. In dot-product type, for example, a unit with the largest dot-product between the input vector and a weight vector associated with each unit is selected. There can be many variations in this selection process and we may even choose more than one winner. (Lee and Verleysen 2002)

The second stage is a learning with cooperative characteristics: The connection strength (called weight) from an input unit to an output unit is updated if the output unit is the winner or if it belongs to the winner's neighbour. A learning rule specifies the amount of weight change as a function of present weight value and input value. There can also be various variations in defining learning rules. (Hulle 1997)

Once a particular SOM formation algorithm is given, one of the most important problems is what kind of classifier will be actually formed eventually and what is its stability. For Kohonen's SOM algorithm, there are many simulation results to answer this question but as far as theoretical consideration is concerned, only a few works seem to exist and only for low dimensional cases. (Benaïm, Fort and Pagès 1997)

In this note, we treat a dot-product type SOM algorithm, and analyze the stability of resulting classifier functions. Our method is of combinatorial nature, and we introduce some new concepts which are suitable for our purposes. Our point is that we do not assume certain topographic mappings a priori but derive them through their stability requirements.

2. Two Layered Network as Pattern Classifier

2.1. Network Scheme

We consider a two layered feedforward network in which the input layer has n units and each of them is designated by an integer from 1 to n. That is, we have the set of input units $V_I = \{1, 2, ..., n\}$. Each input unit i takes a value x_i from a closed interval of real numbers from 0 to 1. Thus an input pattern is an n-dimensional real vector: $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ where $x_i \in [0, 1]$ for $i = 1, 2, \ldots, n$. We consider an input pattern set $\boldsymbol{I}$ which is a subset of $[0, 1]^n$, and assume that the number of the input patterns is finite.

When there are m output units, each is also designated by an integer from 1 to m. The set of output units is then $V_O = \{1, 2, ..., m\}$. There is a connection from $i \in V_I$ to $j \in V_O$ with weight value $w_{ji} \in [0, 1]$. The entire set of weight values are expressed in a matrix form $W = (w_{ji})$ and a j-th row vector is designated by $\mathbf{w}_j = (w_{j1}, w_{j2}, \ldots, w_{jn})$ for $j \in V_O$.

The network scheme is $\boldsymbol{N} = (V_I, (V_O, E_O), \boldsymbol{I}, \boldsymbol{W})$ where $\boldsymbol{W}$ is the set of $m \times n$ matrices over [0, 1]. We here consider a relation $E_O \subseteq V_O \times V_O$ sometimes called a neighbourhood relation, which is reflexive and symmetric. The relation can also be expressed by a function $\sigma_O : V_O \to 2^{V_O}$ where $\sigma_O(u) = \{v \in V_O \mid (u, v) \in E_O\}$. Note that $\sigma_O(u)$ contains u itself and its neighbouring units defined by E_O.

2.2. Winner Function and Unsupervised Learning

Consider a network scheme $\boldsymbol{N} = (V_I, (V_O, E_O), \boldsymbol{I}, \boldsymbol{W})$ with a weight matrix $W \in \boldsymbol{W}$. We assign each output unit $j \in V_O$ a value $y_j = \sum_{i=1}^{n} w_{ji} x_i = \mathbf{w}_j \mathbf{x}$ for an input pattern $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \boldsymbol{I}$. We select a unique output unit which has the maximum output value. That is, let $k = \arg \max_{j \in V_O} \{y_j\}$ and assign k to the input $\mathbf{x}$. (When there are more than one maximum output value unit, select, for example, the one with the smallest index.) We call this a winner function $f : \boldsymbol{I} \to V_O$ where $f(\mathbf{x}) = k$. A weight matrix $W \in \boldsymbol{W}$ thus determines a corresponding winner function f. We write this correspondence as F: $\boldsymbol{W} \to V_O^{\mathbf{I}}$ such that

$F(W) = f$.
Let $\mathbf{x}^{(p)} \in \boldsymbol{I}$ and $f(\mathbf{x}^{(p)}) = k$. Then a standard SOM type learning for an input $\mathbf{x}^{(p)}$ proceeds as

$$\mathbf{w}_j' = \begin{cases} \mathbf{w}_j + \alpha(\mathbf{x}^{(p)} - \mathbf{w}_j) & if\ j \in \sigma_O(k) \\ \mathbf{w}_j & if\ j \notin \sigma_O(k) \end{cases} \quad \text{for } j = 1, 2, \ldots, m$$

where $\alpha \in (0, 1)$ is a learning rate and $\mathbf{w}_j'$ is an updated vector of $\mathbf{w}_j$.
We write the above network as $\boldsymbol{N}(W, \alpha) = (V_I, (V_O, E_O), \boldsymbol{I}, \boldsymbol{W}, W, \alpha)$ or simply (W, α) when the scheme is understood.
Let $\langle p \rangle$ take the value 1 if a proposition p is true and 0 if p is false. Then the above update rule for $\mathbf{w}_j$ can be written as follows.

For $j = 1, 2, \ldots, m$, $\quad \mathbf{w}_j' = \mathbf{w}_j + \langle j \in \sigma_O(f(\mathbf{x}^{(p)})) \rangle \alpha(\mathbf{x}^{(p)} - \mathbf{w}_j)$.

3. Stability Conditions

Let $\boldsymbol{N}(W, \alpha)$ be a network such that $F(W) = f$. Consider y_j' an $\mathbf{x}^{(p)}$-learning such that $f(\mathbf{x}^{(p)}) = k$ which yields $W' = (\mathbf{w}_1', \mathbf{w}_2', \ldots, \mathbf{w}_m')^T$ where T means the transpose of a matrix. Let $F(W') = f'$. For an arbitrarily fixed $\mathbf{x}^{(q)} \in \boldsymbol{I}$, let $f(\mathbf{x}^{(q)}) = l$. We evaluate $f'(\mathbf{x}^{(q)})$ and find conditions which yield $f'(\mathbf{x}^{(q)}) = f(\mathbf{x}^{(q)})$.
Let y_j' be the output value of j unit for the input $\mathbf{x}^{(q)}$ after $\mathbf{x}^{(p)}$-learning. That is,

$$y_j' = \mathbf{w}_j' \mathbf{x}^{(q)} = (\mathbf{w}_j + \langle j \in \sigma_O(f(\mathbf{x}^{(p)})) \rangle \alpha(\mathbf{x}^{(p)} - \mathbf{w}_j))\mathbf{x}^{(q)} \text{ for } j = 1, 2, \ldots, m.$$

In order that $f'(\mathbf{x}^{(q)}) = f(\mathbf{x}^{(q)}) = l$, we should have $y_l' \geq y_j'$ for $j = 1, 2, \ldots, m$.
Thus, after $\mathbf{x}^{(p)}$-learning, $f' = f$ if and only if the following condition holds.

For $\forall\, \mathbf{x}^{(q)} \in \boldsymbol{I}$, let $f(\mathbf{x}^{(q)}) = l$. Then $\mathbf{w}_l' \mathbf{x}^{(q)} \geq \mathbf{w}_j' \mathbf{x}^{(q)}$ for $j = 1, 2, \ldots, m$.

By substituting the updated weight vectors $\mathbf{w}_j'$, we have

$$(\mathbf{w}_l + \langle l \in \sigma_O(k) \rangle \alpha (\mathbf{x}^{(p)} - \mathbf{w}_l))\mathbf{x}^{(q)} \geq (\mathbf{w}_j + \langle j \in \sigma_O(k) \rangle \alpha (\mathbf{x}^{(p)} - \mathbf{w}_j))\mathbf{x}^{(q)}$$

for $j = 1, 2, \ldots, m$ where $l = f(\mathbf{x}^{(q)})$ and $k = f(\mathbf{x}^{(p)})$.
We have to consider the following four cases depending on the values of $\langle l \in \sigma_O(k) \rangle$ and $\langle j \in \sigma_O(k) \rangle$ as follows.

	$j \in \sigma_O(k)$	$j \notin \sigma_O(k)$
$l \in \sigma_O(k)$	$\mathbf{w}_l \mathbf{x}^{(q)} \geq \mathbf{w}_j \mathbf{x}^{(q)}$	$\alpha(\mathbf{x}^{(p)} - \mathbf{w}_l)\mathbf{x}^{(q)} \geq (\mathbf{w}_j - \mathbf{w}_l)\mathbf{x}^{(q)}$
$l \notin \sigma_O(k)$	$(\mathbf{w}_l - \mathbf{w}_j)\mathbf{x}^{(q)} \geq \alpha(\mathbf{x}^{(p)} - \mathbf{w}_j)\mathbf{x}^{(q)}$	$\mathbf{w}_l \mathbf{x}^{(q)} \geq \mathbf{w}_j \mathbf{x}^{(q)}$

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 349-354

Since $\mathbf{w}_l\,\mathbf{x}^{(q)} \geq \mathbf{w}_j\,\mathbf{x}^{(q)}$ ($j = 1, 2, \ldots, m$) by assumption that $f(\mathbf{x}^{(q)}) = l$, the above conditions can be rewritten in a succinct form as below.

Definition 1. A network (W, α) is said to be α - *conservative* if and only if the following condition holds.

For any $\mathbf{x}^{(p)}, \mathbf{x}^{(q)} \in \boldsymbol{I}$, let k and l denote $f(\mathbf{x}^{(p)})$ and $f(\mathbf{x}^{(q)})$, respectively.

If $(f(\mathbf{x}^{(p)}), f(\mathbf{x}^{(q)})) \in E_O$ then $\alpha(\mathbf{x}^{(p)} - \mathbf{w}_l)\mathbf{x}^{(q)} \geq (\mathbf{w}_j - \mathbf{w}_l)\mathbf{x}^{(q)}$ for $j \notin \sigma_O(k)$.

If $(f(\mathbf{x}^{(p)}), f(\mathbf{x}^{(q)})) \notin E_O$ then $(\mathbf{w}_l - \mathbf{w}_j)\mathbf{x}^{(q)} \geq \alpha(\mathbf{x}^{(p)} - \mathbf{w}_j)\mathbf{x}^{(q)}$ for $j \in \sigma_O(k)$.

Then we have the following basic properties.

Lemma 1.

(W, α) is α - conservative if and only if W and W' defines the same winner function where W' is an updated weight matrix after $\mathbf{x}^{(p)}$-learning for any $\mathbf{x}^{(p)} \in \boldsymbol{I}$.

Lemma 2.

For α, $\alpha' \in (0, 1)$ such that $\alpha \geq \alpha'$, if (W, α) is α - conservative, then (W, α') is α' - conservative.

Proof. For α, α' in the Lemma, and arbitrary real numbers A and B, we have the following relations. (i) If $\alpha A \geq B$ and $0 \geq B$ then $\alpha' A \geq B$. (ii) If $\alpha A \leq B$ and $0 \leq B$ then $\alpha' A \leq B$. Using these relations, we can easily show the desired result.

Definition 2. A network W is said to be *conservative* if (W, α) is α - conservative for any $\alpha \in (0, 1)$.

By Lemma 2, we let α tend to 1 to have the following conservative condition.

Lemma 3.

For any $\mathbf{x}^{(p)}, \mathbf{x}^{(q)} \in \boldsymbol{I}$, let k and l denote $f(\mathbf{x}^{(p)})$ and $f(\mathbf{x}^{(q)})$, respectively. W is conservative if and only if the following holds.

If $(f(\mathbf{x}^{(p)}), f(\mathbf{x}^{(q)})) \in E_O$ then $\mathbf{x}^{(p)}\mathbf{x}^{(q)} \geq \mathbf{w}_j\mathbf{x}^{(q)}$ for $j \notin \sigma_O(k)$.

If $(f(\mathbf{x}^{(p)}), f(\mathbf{x}^{(q)})) \notin E_O$ then $\mathbf{x}^{(p)}\mathbf{x}^{(q)} \leq \mathbf{w}_l\mathbf{x}^{(q)}$.

Proof. (Omitted.)

4. Topographic Condition

Hence the conservative condition does not necessarily yield a dichotomy of dot-product input space, we consider a bit more strict case as follows.

Definition 3. W is said to be *topographic* if and only if the following holds.

For any $\mathbf{x}^{(p)}, \mathbf{x}^{(q)} \in \boldsymbol{I}$,

if $(f(\mathbf{x}^{(p)}), f(\mathbf{x}^{(q)})) \in E_O$ then $\mathbf{x}^{(p)}\mathbf{x}^{(q)} > \mathbf{w}_l\mathbf{x}^{(q)}$ and

if $(f(\mathbf{x}^{(p)}), f(\mathbf{x}^{(q)})) \notin E_O$ then $\mathbf{x}^{(p)}\mathbf{x}^{(q)} \leq \mathbf{w}_l\mathbf{x}^{(q)}$

where $f = F(W)$ and $l = f(\mathbf{x}^{(q)})$.

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 349-354

Lemma 4.
W is topographic if and only if the following relation holds.
For any $\mathbf{x}^{(p)}, \mathbf{x}^{(q)} \in \boldsymbol{I}$,

$$\mathbf{x}^{(p)}\mathbf{x}^{(q)} > \mathbf{w}_l\mathbf{x}^{(q)} \text{ implies } (f(\mathbf{x}^{(p)}), f(\mathbf{x}^{(q)})) \in E_O \text{ and}$$

$$\mathbf{x}^{(p)}\mathbf{x}^{(q)} \le \mathbf{w}_l\mathbf{x}^{(q)} \text{ implies } (f(\mathbf{x}^{(p)}), f(\mathbf{x}^{(q)})) \notin E_O$$

where $f = F(W)$ and $l = f(\mathbf{x}^{(q)})$.

It is easy to see that if W is topographic, then it is conservative. When W is topographic, what can we say about the stability of corresponding winner function?
Assume that a given W be topographic. For any $\alpha \in (0, 1)$ and $\mathbf{x} \in \boldsymbol{I}$, consider a step of $\mathbf{x}$-learning in (W, α) to obtain W'. As noted above, W is α-conservative and $F(W') = f' = f$. We analyze the conditions when W' becomes topographic.
For any $\mathbf{x}^{(p)}, \mathbf{x}^{(q)} \in \boldsymbol{I}$,
(I)Let $(f'(\mathbf{x}^{(p)}), f'(\mathbf{x}^{(q)})) \notin E_O$, which is equivalent to $(f(\mathbf{x}^{(p)}), f(\mathbf{x}^{(q)})) \notin E_O$. Then we have $\mathbf{x}^{(p)}\mathbf{x}^{(q)} \le \mathbf{w}_l\mathbf{x}^{(q)}$. Evaluate $\mathbf{w}_l'\mathbf{x}^{(q)} - \mathbf{x}^{(p)}\mathbf{x}^{(q)} = \{\mathbf{w}_l + \langle l \in \sigma_O(f(\mathbf{x}))\rangle\, \alpha\, (\mathbf{x} - \mathbf{w}_l)\}\, \mathbf{x}^{(q)} - \mathbf{x}^{(p)}\mathbf{x}^{(q)}$ depending on the value of $\langle l \in \sigma_O(f(\mathbf{x}))\rangle$ as follows.
(I-a) $l \notin \sigma_O(f(\mathbf{x}))$: $\Delta_a = \mathbf{w}_l\mathbf{x}^{(q)} - \mathbf{x}^{(p)}\mathbf{x}^{(q)} \ge 0$.
(I-b) $l \in \sigma_O(f(\mathbf{x}))$: In this case, we have $\mathbf{x}\,\mathbf{x}^{(q)} > \mathbf{w}_l\,\mathbf{x}^{(q)}$ by assumption. Then,

$$\Delta_b = \{(1 - \alpha)\mathbf{w}_l + \alpha\,\mathbf{x}\}\, \mathbf{x}^{(q)} - \mathbf{x}^{(p)}\mathbf{x}^{(q)} > (1 - \alpha)\mathbf{w}_l\,\mathbf{x}^{(q)} + \alpha\; \mathbf{w}_l\,\mathbf{x}^{(q)} - \mathbf{x}^{(p)}\mathbf{x}^{(q)}$$
$$= \mathbf{w}_l\,\mathbf{x}^{(q)} - \mathbf{x}^{(p)}\mathbf{x}^{(q)} \ge 0.$$

(II) If $(f'(\mathbf{x}^{(p)}), f'(\mathbf{x}^{(q)})) \in E_O$ then $\mathbf{x}^{(p)}\mathbf{x}^{(q)} > \mathbf{w}_l\mathbf{x}^{(q)}$ by assumption. In this case, we evaluate $\mathbf{x}^{(p)}\mathbf{x}^{(q)} - \mathbf{w}_l'\mathbf{x}^{(q)} = \mathbf{x}^{(p)}\mathbf{x}^{(q)} - \{\mathbf{w}_l + \langle l \in \sigma_O(f(\mathbf{x}))\rangle\, \alpha\, (\mathbf{x} - \mathbf{w}_l)\}\, \mathbf{x}^{(q)}$ depending on the value of $\langle l \in \sigma_O(f(\mathbf{x}))\rangle$ as before.
(II-c) $l \notin \sigma_O(f(\mathbf{x}))$: $\Delta_c = \mathbf{x}^{(p)}\mathbf{x}^{(q)} - \mathbf{w}_l\,\mathbf{x}^{(q)} > 0$.
(II-d) $l \in \sigma_O(f(\mathbf{x}))$: In this case, as in (I-b), we have $\mathbf{x}\,\mathbf{x}^{(q)} > \mathbf{w}_l\,\mathbf{x}^{(q)}$ by assumption. Then, $\Delta_d = \mathbf{x}^{(p)}\mathbf{x}^{(q)} - \{(1 - \alpha)\mathbf{w}_l + \alpha\,\mathbf{x}\}\, \mathbf{x}^{(q)} = (\mathbf{x}^{(p)}\mathbf{x}^{(q)} - \mathbf{w}_l\,\mathbf{x}^{(q)}) - \alpha\,(\mathbf{x}\,\mathbf{x}^{(q)} - \mathbf{w}_l\,\mathbf{x}^{(q)})$. If we put $A = \mathbf{x}^{(p)}\mathbf{x}^{(q)} - \mathbf{w}_l\,\mathbf{x}^{(q)}$ and $C = \mathbf{x}\,\mathbf{x}^{(q)} - \mathbf{w}_l\,\mathbf{x}^{(q)}$), then $\Delta_d = A - \alpha C$, $A > 0$, and $C > 0$. If we choose α as less than A/C, then $\Delta_d > 0$.
This concludes that W' is still topographic if α is selected appropriately. Our main result is then summarized as follows.
Theorem 1.
Let (W, α) be a network where W is topographic. If W' denotes an updated matrix after $\mathbf{x}$-learning where $\mathbf{x}$ is an arbitrary input pattern, then W' is also topographic for some appropriate learning rate α.

By the above Theorem and Lemmas, we see that once the topographic condition is attained, we can keep the same winner function by choosing appropriate non-zero learning rate for arbitrary sequence of **x**-learning. This means that the topographic condition implies a stable winner function if we can choose arbitrary learning rate $\alpha \in (0, 1)$.

5. Concluding Remarks

This note tried to clarify what kinds of classifier functions are ultimately obtained in SOM algorithm. If they are to be stable, what are the necessary and sufficient conditions? We first found such conditions that a winner function f becomes stable under an ordinary learning dynamics as α -conservativeness. Note that this is a stability condition of f for one-step learning. We obtained the condition from the requirement that W and W' defines the same winner function where W' is an updated weight matrix after any one-step **x**-learning. Then we deduce the conservativeness concept as α -conservativeness for arbitrary $\alpha \in (0, 1)$. Since the conservative condition does not necessarily yield a dichotomy, we consider a more strict case as topographic. So the topographic property is introduced as a special case of conservativeness. The topographic property thus has the characteristics that the property itself (not only the winner function) can be preserved by appropriately choosing learning rates. Note that there are inclusion relations of the three concepts. That is, W is topographic implies W is conservative, which implies W is α -conservative for arbitrary $\alpha \in (0, 1)$. All the analyses are done using dot-product (i.e., an inner product) as the means to choose a winner unit.

Acknowledgements

This has been written while the first author has been staying at Université Catholique de Louvain as a research fellow of Ryukoku University (2003). Thanks are due to Prof. Verleysen(UCL) for his hospitality and interest. Discussions with Prof. Seki(California State University, Fresno) and Dr. Sakamoto at various stages of this work are gratefully acknowledged. Thanks are also due to the anonymous reviewers whose comments were helpful in revising this note.

References

Benaïm, Fort and Pagès (1997) 5^{th} ESANN Proceedings pp.193-198
Kohonen(1995) Self-Organizing Maps. Springer 362pp.
Lee and Verleysen(2002) Neural Networks Vol. 15, pp.993-1003
Sakamoto and Kobuchi(2000) Neural Networks Vol. 13, pp. 709-718
Sakamoto and Kobuchi(2002) IEICE Trans. on Inf. Sys. Vol. E85-D No.7 pp. 1145-1152
Sakamoto, Seki and Kobuchi(2003) 11^{th} ESANN Proceedings pp.119-124
Van Hulle(1997) Neural Networks Vol. 10, No. 3, pp.431-446
Van Hulle(2000) Faithful Representations and Topographic Maps. Wiley 258pp.