数理解析研究所講究録 1415 巻 2005 年 185-193

Stationary states in some self-organization
(自己組織化における定常状態について)

秋田県立大学 システム科学技術学部　星野満博 (Mitsuhiro Hoshino)
秋田県立大学 システム科学技術学部　木村 寛 (Yutaka Kimura)
Faculty of Systems Science and Technology, Akita Prefectural University

1. 自己組織化マップ

本報告は，多分野において広く応用されている Kohonen タイプ・アルゴリズムまたは自己組織化マップ（self-organizing maps）と呼ばれるアルゴリズムにおける諸問題についての一考察である．主として，このタイプのアルゴリズムにおいて現れる，ある種の定常性及び単調性に関する研究を紹介する．

自己組織化マップのモデルは，Kohonen 等による研究 [5] が有名であるが，ここでは，ノード，ノードの値，入力，学習の 4 要素により特徴づけることとする．以下のように仮定する．

(I) E をノードの空間とし，距離空間であるものとする．ノード（またはユニット，セル）の集合 I は E のある部分集合であるものとする．

(II) 各ノードは，その値をもつ．A をノードの値の空間とし，ノルム空間であるものとする．ノード i をその値 $a(i)$ に対応させる写像 $a: I \to A$ をリファレンス関数と呼ぶ．R をリファレンス関数全体の集合とする．$R = \{a \mid a : I \to A\}$.

(III) 入力集合 X は A のある部分集合であるものとする．$x \in X$ を入力と呼ぶ.

(IV) 入力 x とリファレンス関数 a が与えられとき，リファレンス関数 a は，学習関数 $L_{a,x}: I \to [0,1]$ によって定まる学習率 $L_{a,x}(i)$ に従い入力 x を学習する．この結果，リファレンス関数 a は次のように定義されるリファレンス関数 a' へと更新される．

$$a'(i) = (1 - L_{a,x}(i))a(i) + L_{a,x}(i)x = a(i) + L_{a,x}(i)(x - a(i)). \tag{1.1}$$

今，初期リファレンス関数 a_0 と入力列として $x_0, x_1, x_2, \ldots \in X$ が与えられたとする．このとき，

$$a_{k+1}(i) = (1 - L_{a_k,x_k}(i))a_k(i) + L_{a_k,x_k}(i)x_k = a_k(i) + L_{a_k,x_k}(i)(x_k - a_k(i)) \tag{1.2}$$

によってリファレンス関数 $a_1, a_2, a_3 \ldots$ が逐次に生成される．この生成過程または，X と生成されたリファレンス関数との対応を自己組織化マップと呼ぶことにする．

一般に，適用する問題に応じて，想定するノードの空間，ノード値の空間，および学習方法等，様々なバリエーションが考えられる．通常，応用上はノード集合Iとして，$\boldsymbol{Z}$（整数全体）または$\boldsymbol{Z}^2$の有限部分集合と対等な集合が用いられているが，距離の入れ方は様々である．

本報告においては，学習方法として等式(1.1)および(1.2)を採用する．各リファレンス関数$a \in R$，入力$x \in X$に対して，集合$I(a,x)$を

$$I(a,x) = \left\{ i^* \in I \;\middle|\; \|a_k(i^*) - x_k\| = \inf_{i \in I} \|a_k(i) - x_k\| \right\} \tag{1.3}$$

によって定義する．学習関数$L_{a,x}$として，次のように定義される2つのタイプのものを考える．ここで，関数$L : [0,\infty) \to [0,1]$は単調減少であるものとする．

(i) 学習関数$L^2_{a,x} : I \to [0,1]$を$L^2_{a,x}(i) = L\Big(d_E(J(a,x),i)\Big)$によって定義する．写像$J : R \times X \to I$は$J(a,x) \in I(a,x)$を満たすものとする．

(ii) 学習関数$L^1_{a,x} : I \to [0,1]$を$L^1_{a,x}(i) = L\Big(\inf_{j \in I(a,x)} d_E(j,i)\Big)$によって定義する．ただし，$d_E(\cdot,\cdot)$は$E \times E$上で定義される距離関数を表わす．

2. 1次元ノード配列・$\mathbb{R}$-値ノードの場合

ここでは，自己組織化マップの基本的な例として，加算なノード空間を仮定して，ノードの配列が1次元で，各ノードが実数の値をもつような場合を考える．

(1) ノード集合$I = \{1,2,\ldots,n\} \subset \mathbb{N}$.

(2) 各ノードは，それぞれ1つの値（ここでは実数値）をもつ．ノードの値の空間を$\mathbb{R}$（実数値全体，ユークリッドノルム）とし，初期リファレンス関数を$a_0 : I \to \mathbb{R}$とする.

(3) 入力列$x_0, x_1, x_2, \ldots \in X \subset \mathbb{R}$.

(4) 学習プロセスを次のように定義する. 各$k = 0,1,2,\ldots$に対して

学習範囲: $J(a_k,x_k) = \min\left\{ i^* \in I \;\middle|\; |a_k(i^*) - x_k| = \inf_{i \in I} |a_k(i) - x_k| \right\},$

$N_\delta(J(a_k,x_k)) = \left\{ j \in I \;\middle|\; |j - J(a_k,x_k)| \le \delta \right\}.$

学習率: $0 \le \alpha \le 1.$

学習方法: $a_{k+1}(i) = \begin{cases} (1-\alpha)a_k(i) + \alpha x_k, & \text{if } i \in N_\delta(J(a_k,x_k)), \\ a_k(i), & \text{if } i \notin N_\delta(J(a_k,x_k)), \end{cases}$

今，n個のノード$1, 2, \ldots, n$があり，そのそれぞれに対してノードの値$a_0(1), a_0(2), \ldots, a_0(n)$が与えられている．このとき，入力とこれに伴う学習により各ノードの値が更新される．$x_0 \in X$が入力されたならば，$a_0(1), a_0(2), \ldots, a_0(n)$のなかで$x_0$と最も近いものを選び，その値に対応するノード$i^*$とその周囲のノード$i$に対して学習

$$a_1(i) = (1-\alpha)a_1(i) + \alpha x_1$$

が適用され，それ以外のノードに対しては学習が適用されず，$a_1(i) = a_0(i)$となる．入力$x_1, x_2, x_3, \ldots$に対して，これを繰り返すことにより，逐次にノードの更新がおこなわれ，同時にリファレンス関数$a_1, a_2, a_3, \ldots$が逐次に生成される．

上述のような1次元ノード配列でノードの値が実数値の場合，ノードとノードの値の対応を表すリファレンス関数$a: I \to \mathbb{R}$を

$$a = [a(1), a(2), \ldots, a(n)] \tag{2.1}$$

と表すことにする．

例 1 ノード集合を$I = \{1, 2, 3, 4, 5\}$，ノードの値の空間を$\mathbb{R}$，初期リファレンス関数を$a_0 = [2, 3, 1, 5, 4]$とし，入力列を$4, 5, 2, 1, 3, 4, 3, 5, 4, 1, 5, \ldots$とする．学習方法は次のように定める．

$$J(a_k, x_k) = \min I(a_k, x_k) = \min\left\{ i^* \in I \,\middle|\, |a_k(i^*) - x_k| = \inf_{i \in I} |a_k(i) - x_k| \right\},$$

$$\begin{aligned} N_1(J(a_k, x_k)) &= \left\{ j \in I \,\middle|\, |j - J(a_k, x_k)| \le 1 \right\} \\ &= \{J(a_k, x_k) - 1, J(a_k, x_k), J(a_k, x_k) + 1\} \cap I : \quad \text{学習範囲}, \end{aligned}$$

$$\alpha = \frac{1}{2}: \quad \text{学習率},$$

$$a_{k+1}(i) = \begin{cases} \frac{1}{2}a_k(i) + \frac{1}{2}x_k & i \in N_1(J(a_k, x_k)), \\ a_k(i) & i \notin N_1(J(a_k, x_k)). \end{cases} \qquad k = 0, 1, 2, \ldots.$$

このとき，ノードの値は次のように更新される．

初期リファレンス関数a_0と入力$x_0 = 4$から

$$\begin{aligned} &J(a_0, x_0) = 5, N_1(5) = \{4, 5\}, \\ &a_1(1) = a_0(1) = 2, \quad a_1(2) = a_0(2) = 3, \quad a_1(3) = a_0(3) = 1, \\ &a_1(4) = \tfrac{a_0(4)+x_0}{2} = 4.5, \quad a_1(5) = \tfrac{a_0(5)+x_0}{2} = 4 \\ &a_1 = [2,\ 3,\ 1,\ 4.5,\ 4]. \end{aligned}$$

リファレンス関数a_1と入力$x_1 = 5$から

$$\begin{aligned} &J(a_1, x_1) = 4, \quad N_1(4) = \{3, 4, 5\}, \\ &a_2(1) = a_1(1) = 2, \quad a_2(2) = a_1(2) = 3, \quad a_2(3) = \tfrac{a_1(3)+x_1}{2} = 3, \\ &a_2(4) = \tfrac{a_1(4)+x_1}{2} = 4.75, \quad a_2(5) = \tfrac{a_1(5)+x_1}{2} = 4.5, \\ &a_2 = [2,\ 3,\ 3,\ 4.75,\ 4.5]. \end{aligned}$$

これを繰り返すことにより

$$
\begin{aligned}
a_3 &= [2,\ 2.5,\ 3,\ 4.75,\ 4.5],\\
a_4 &= [1.5,\ 1.75,\ 3.,\ 4.75,\ 4.5],\\
a_5 &= [1.5,\ 2.375,\ 3.,\ 3.875,\ 4.5],\\
a_6 &= [1.5,\ 2.375,\ 3.5,\ 3.9375,\ 4.25],\\
a_7 &= [1.5,\ 2.6875,\ 3.25,\ 3.46875,\ 4.25],\\
a_8 &= [1.5,\ 2.6875,\ 3.25,\ 4.23438,\ 4.625],\\
a_9 &= [1.5,\ 2.6875,\ 3.625,\ 4.11719,\ 4.3125],\\
a_{10} &= [1.25,\ 1.84375,\ 3.625,\ 4.11719,\ 4.3125].\\
&\cdots\cdots
\end{aligned}
$$

が得られる． □

上の例1において，初期リファレンスおよび入力列は特別意味のある数，数列ではないが

$$k \geq 5 \text{ のとき } \ a_k(1) \leq a_k(2) \leq a_k(3) \leq a_k(4) \leq a_k(5)$$

が成り立っている．このように，学習を繰り返すことにより，リファレンス関数が単調性等の性質をもち，各ノードの値の配列にある種の規則性が現れることがあるが，上記の過程において現れるこのような現象は組織化とよばれている．実際，様々なノード集合，ノードの値の空間，学習方法において，このような組織化現象を含む幾つかの興味深い現象が現れる．また，これらの性質を利用することにより，多くの実問題への応用が成されている．しかしながら，組織化等の現象は常に起こるわけではなく，実アプリケーションにおいては，求めるべき結果を得る為に，モデルに応じた学習方法を採用している．ここでの，どのように学習をさせるべきかという問題は非常に難解であると言われている．また，一般に巧く学習させると，組織化された状態，すなわちノードの値が整列した状態に収束する．別の言い方をするならば，要求する良い状態へ収束するように学習させる．例1のように，学習率が固定された値の場合は，いつまでも学習が繰り返され収束しないが，多くの応用において，具体的には，徐々に学習率を小さくして，最終的に0にすることにより強制的に収束させている．

これらの性質を利用した応用として，巡回セースルマン問題がある．巡回セールスマン問題においては，通常，ノードの値の空間として，上記の設定である1次元ユークリッド空間ではないが，2次元のユークリッド空間を用いる．また，入力集合として巡回する都市全体を考え，更に巡回を考慮させる為にノードの空間に1次元配列の両端を結合したループ状の位相構造を導入する．

今，入力条件と学習方法に対して次のような条件を定義する．

入力条件 1 入力変数 x に対して，$P(\{x=c\})>0$ を満たす $c\in\mathbb{R}$ が存在しない．

学習方法 1 学習方法を次のように定義する．各 $k=0,1,2,\dots$ に対して

$$\text{学習範囲:}\quad J(a_k,x_k)=\min\left\{i^*\in I\,\middle|\,|a_k(i^*)-x_k|=\inf_{i\in I}|a_k(i)-x_k|\right\},$$

$$N_1(J(a_k,x_k))=\left\{j\in I\,\middle|\,|j-J(a_k,x_k)|\le 1\right\}$$

$$\text{学習率:}\quad 0\le\alpha\le 1,$$

$$\text{学習方法:}\quad a_{k+1}(i)=\begin{cases}(1-\alpha)a_k(i)+\alpha x_k & i\in N_1(J(a_k,x_k)),\\ a_k(i) & i\notin N_1(J(a_k,x_k)).\end{cases}$$

上記の設定のもとで以下の基本的な性質が成り立つ．

定理 1 学習 1 を仮定する．リファレンス関数 $a_1,a_2,a_3,\dots$ に対して，以下が成り立つ．

(1) a_k が I 上で単調増加ならば, a_{k+1} も I 上で単調増加である;

(2) a_k が I 上で単調減少であるならば a_{k+1} も I 上で単調減少である.

例 1 においては，各ステップでのノードの値の配列は，初期ノードの値と，入力列に依存することがわかる．このようなモデルにおいて学習率が一定の場合，各ステップでのノードの値への影響を考えるとき，初期ノードの値の依存度と初期の入力値の依存度は徐々に薄れてくる．そこで，入力を入力集合 X の値をとる確率変数の実現値として扱うことにする．

以下では，前節で定義した実数のノード値をとる 1 次元ノード配列の場合に限定する．

入力集合を $X=\mathbb{R}$ とする．x を確率空間 $(\Omega,\mathcal{F},P)$ 上の $\mathbb{R}$-値確率変数とする．また，μ を x の分布，すなわち $\mu(A)=P(x\in A)=P\circ x^{-1}(A)$ $(A\in\mathcal{B})$．$\mathcal{B}$ はボレル集合の σ-algebra である．このとき，x を入力変数と呼ぶことにする．

定義 1 x を確率空間 $(\Omega,\mathcal{F},P)$ 上の $\mathbb{R}$-値確率変数とする．次の等式が成り立つとき，リファレンス関数 $w=[w(1),w(2),\dots,w(n)]:I\to\mathbb{R}$ は入力変数 x に関して定常であるという.

$$\begin{aligned}&E\big[a_{k+1}(i)\,\big|\,a_k=w\big]\\&\quad=E\big[a_{k+1}(i)\,\big|\,a_k(1)=w(1),a_k(2)=w(2),\dots,a_k(n)=w(n)\big]=w(i),\\&\qquad k=1,2,\dots,\quad i=1,2,\dots,n.\end{aligned}$$

ただし，左の 2 つの項は，k 番目のステップにおいてノードの値が $a_k(1)=w(1)$, $a_k(2)=w(2)$, …, $a_k(n)=w(n)$ であるという条件の下での，入力変数 x に対して更新されたノード i の値 $a_{k+1}(i)$ の期待値を意味する．

例 2　学習1を仮定する．入力変数 x が区間 $(0,1]$ 上の一様分布に従うときリファレンス関数 $w=[\frac{3}{10},\frac{5}{10},\frac{7}{10}]$ は

$$\begin{aligned}
E\big[a_{k+1}(1)\mid a_k=w\big] &= \int_0^{\frac{6}{10}}\big\{(1-\alpha)\cdot\tfrac{3}{10}+\alpha x\big\}dx+\int_{\frac{6}{10}}^1 \tfrac{3}{10}dx=\tfrac{3}{10}=a_k(1),\\
E\big[a_{k+1}(2)\mid a_k=w\big] &= \int_0^1\big\{(1-\alpha)\cdot\tfrac{5}{10}+\alpha x\big\}dx=\tfrac{5}{10}=a_k(2),\\
E\big[a_{k+1}(3)\mid a_k=w\big] &= \int_0^{\frac{4}{10}}\tfrac{7}{10}dx+\int_{\frac{4}{10}}^1\big\{(1-\alpha)\cdot\tfrac{7}{10}+\alpha x\big\}dx=\tfrac{7}{10}=a_k(3)
\end{aligned}$$

をみたす．したがって，$w=[\frac{3}{10},\frac{5}{10},\frac{7}{10}]$ は入力 x に対して定常である．　□

定理 2　学習1を仮定する．$w: I\to\mathbb{R}$ をリファレンス関数とする．各 $i=1,2,\ldots,n$ に対して

$$\begin{aligned}
\underline{w}(i) &= \max\{w(j)\mid w(j)<w(i),\ j\in I\},\\
\overline{w}(i) &= \min\{w(j)\mid w(j)>w(i),\ j\in I\},\\
S_i &= \begin{cases}\big(-\infty,\frac{w(i)+\overline{w}(i)}{2}\big), & \text{if } w_i=\min_j w(j),\\ \big[\frac{\underline{w}(i)+w(i)}{2},\frac{w(i)+\overline{w}(i)}{2}\big), & \text{otherwise},\end{cases}
\end{aligned}$$

と置く．ただし，$w_i=\max_j w(j)$ のとき，$\overline{w}(i)=\infty$ とする．

このとき，$w=[w(1),w(2),\ldots,w(n)]$ が定常である為の必要十分条件は次のすべての等式が成り立つことである．

$$\begin{cases}\displaystyle\int_{S_1\cup S_2}\alpha(x-w(1))\mu(dx)=0,\\ \displaystyle\int_{S_{i-1}\cup S_i\cup S_{i+1}}\alpha(x-w(i))\mu(dx)=0,\quad i=2,3,\ldots,n-1,\\ \displaystyle\int_{S_{n-1}\cup S_n}\alpha(x-w(n))\mu(dx)=0.\end{cases}\tag{2.2}$$

証明　k ステップ目におけるノードの値が $a_k(1)=w(1),a_k(2)=w(2),\ldots,a_k(n)=w(n)$，入力が $x_k=x$ であったとする．

$x\in S_1$ のとき，$N_1(J(a_k,x))=\{1,2\}$ より

$$\begin{aligned}
a_{k+1}(1) &= (1-\alpha)a_k(1)+\alpha x,\\
a_{k+1}(2) &= (1-\alpha)a_k(2)+\alpha x,\\
a_{k+1}(j) &= a_k(j) \qquad (j\neq 1,2\ \text{のとき}).
\end{aligned}$$

$x\in S_i$ $(i=2,3,\ldots,n-1)$ のとき，$N_1(J(a_k,x))=\{i-1,i,i+1\}$ より

$$\begin{aligned}
a_{k+1}(i-1) &= (1-\alpha)a_k(i-1)+\alpha x,\\
a_{k+1}(i) &= (1-\alpha)a_k(i)+\alpha x,\\
a_{k+1}(i+1) &= (1-\alpha)a_k(i+1)+\alpha x,\\
a_{k+1}(j) &= a_k(j) \qquad (j\neq i-1,i,i+1\ \text{のとき}).
\end{aligned}$$

$x \in S_n$ のとき，$N_1(J(a_k, x)) = \{n-1, n\}$ より

$$\begin{aligned} a_{k+1}(n-1) &= (1-\alpha)a_k(n-1) + \alpha x, \\ a_{k+1}(n) &= (1-\alpha)a_k(n) + \alpha x, \\ a_{k+1}(j) &= a_k(j) \qquad\qquad (j \neq n-1, n \text{ のとき}). \end{aligned}$$

したがって，$a_k(1) = w(1), a_k(2) = w(2), \ldots, a_k(n) = w(n)$ は，それぞれ，次のように更新される．

$$a_{k+1}(1) = \begin{cases} (1-\alpha)a_k(1) + \alpha x & (x \in S_1 \cup S_2), \\ a_k(1) & (x \notin S_1 \cup S_2), \end{cases}$$

$$a_{k+1}(i) = \begin{cases} (1-\alpha)a_k(i) + \alpha x & (x \in S_{i-1} \cup S_i \cup S_{i+1}), \\ a_k(i) & (x \notin S_{i-1} \cup S_i \cup S_{i+1}), \end{cases} \qquad i = 2, 3, \ldots, n-1,$$

$$a_{k+1}(n) = \begin{cases} (1-\alpha)a_k(n) + \alpha x & (x \in S_{n-1} \cup S_n), \\ a_k(n) & (x \notin S_{n-1} \cup S_n). \end{cases}$$

これらから，$i = 2, 3, \ldots, n-1$ のとき

$$\begin{aligned} &E\big[a_{k+1}(i) \bigm| a_k(1) = w(1), a_k(2) = w(2), \ldots, a_k(n) = w(n)\big] \\ &= \int_{S_{i-1} \cup S_i \cup S_{i+1}} \big\{(1-\alpha)w(i) + \alpha x\big\}\mu(dx) + \int_{(S_{i-1} \cup S_i \cup S_{i+1})^c} w(i)\mu(dx) \\ &= \int_{S_{i-1} \cup S_i \cup S_{i+1}} w(i)\mu(dx) + \int_{S_{i-1} \cup S_i \cup S_{i+1}} \alpha(x - w(i))\mu(dx) \\ &\qquad + \int_{(S_{i-1} \cup S_i \cup S_{i+1})^c} w(i)\mu(dx) \\ &= w(i) + \int_{S_{i-1} \cup S_i \cup S_{i+1}} \alpha(x - w(i))\mu(dx). \end{aligned} \tag{2.3}$$

が成り立つ．ここで，$(S_{i-1} \cup S_i \cup S_{i+1})^c$ は集合 $S_{i-1} \cup S_i \cup S_{i+1}$ の補集合を意味する．同様に

$$\begin{aligned} &E\big[a_{k+1}(1) \bigm| a_k(1) = w(1), a_k(2) = w(2), \ldots, a_k(n) = w(n)\big] \\ &\qquad = w(1) + \int_{S_1 \cup S_2} \alpha(x - w(1))\mu(dx), \end{aligned} \tag{2.4}$$

$$\begin{aligned} &E\big[a_{k+1}(n) \bigm| a_k(1) = w(1), a_k(2) = w(2), \ldots, a_k(n) = w(n)\big] \\ &\qquad = w(n) + \int_{S_{n-1} \cup S_n} \alpha(x - w(n))\mu(dx). \end{aligned} \tag{2.5}$$

が成り立つ．

このように，$w = [w(1), w(2), \ldots, w(n)]$ が定常なリファレンス関数であるための必要十分条件は (2.3)，(2.4)，(2.5) の右辺の積分が 0 になることである． □

$I = \{1, 2, \ldots, n\}$ をノード集合とするとき，次のようなリファレンス関数の集合を定義する．

$$M_{\mathrm{in}} = \{w : I \to \mathbb{R} \mid w(1) \leq w(2) \leq \ldots \leq w(n)\},$$
$$M_{\mathrm{de}} = \{w : I \to \mathbb{R} \mid w(1) \geq w(2) \geq \ldots \geq w(n)\}.$$

定理 3 $w \in M_{\mathrm{in}}$ のとき $E[a_{k+1}(\cdot)|a_k = w] \in M_{\mathrm{in}}$. $w \in M_{\mathrm{de}}$ のとき $E[a_{k+1}(\cdot)|a_k = w] \in M_{\mathrm{de}}$.

写像 $T_{\mathrm{in}} : M_{\mathrm{in}} \to M_{\mathrm{in}}$，$T_{\mathrm{de}} : M_{\mathrm{de}} \to M_{\mathrm{de}}$ を次のように定義する．

$$T_{\mathrm{in}} w(i) = E[a_{k+1}(i)|a_k = w], \qquad i = 1, 2, \ldots, n,$$
$$T_{\mathrm{de}} w(i) = E[a_{k+1}(i)|a_k = w], \qquad i = 1, 2, \ldots, n.$$

このとき，以下が成り立つ．

定理 4 $\mathbb{R}$ のノルムとして sup ノルムを用いるとき，入力変数 x が区間 $(a, b]$ 上の一様分布に従うならば，T_{in}，T_{de} は nonexpansive 写像である．

例 3 $T_{\mathrm{in}}, T_{\mathrm{de}}$ が縮小写像とならない例．入力変数 x が $(0, 1]$ 上の一様分布に従い，ノードの個数が 5 個の場合を考える．$w = [0.1, 0.2, 0.3, 0.4, 0.5]$，$y = [0.2, 0.3, 0.4, 0.5, 0.6]$ のとき $\|T_{\mathrm{in}} w - T_{\mathrm{in}} y\|_\infty = \|w - y\|_\infty$. □

3. 2次元ノード配列・$\mathbb{R}$-値ノードの場合について

ここでは，ノードの配列が 2 次元で，ノードの値が実数である場合について考える．1 次元の場合と同様に以下のように定義することができる．

(1) ノード集合 $I = \{1, 2, \ldots, m\} \times \{1, 2, \ldots, n\}$.

(2) ノードの値の空間を $\mathbb{R}$（実数値全体，ユークリッドノルム）とし，初期リファレンス関数を $a_0 : I \to \mathbb{R}$ とする．リファレンス関数 $a : I \to \mathbb{R}$ を

$$a = \begin{bmatrix} a(1,1) & a(1,2) & \ldots & a(1,n) \\ a(2,1) & a(2,2) & \ldots & a(2,n) \\ \vdots & \vdots & \ddots & \vdots \\ a(m,1) & a(m,2) & \ldots & a(m,n) \end{bmatrix}$$

と記すことにする．

(3) 入力変数 x. x を $\mathbb{R}$-値確率変数として x_k $(k = 0, 1, 2, \ldots)$ を k 番目のステップにおける入力とする．

(4) 学習方法として以下を仮定する．各 $k = 0, 1, 2, \ldots$ に対して

学習範囲：

$$J(a_k, x_k) = \min\left\{(i^*, j^*) \in I \;\middle|\; |a_k(i^*, j^*) - x_k| = \min_{(i,j)\in I} |a_k(i, j) - x_k|\right\},$$

$$N_\varepsilon(J(a_k, x_k)) = \left\{(i, j) \in I \;\middle|\; d((i, j), J(a_k, x_k)) \le \varepsilon\right\},$$

ここでの I における順序として例えば辞書式順序を用いる．

学習率： $0 \le \alpha \le 1$.

学習方法：

$$a_{k+1}(i, j) = \begin{cases} (1-\alpha)a_k(i, j) + \alpha x_k, & \text{if } (i, j) \in N_\varepsilon(J(a_k, x_k)), \\ a_k(i, j), & \text{if } (i, j) \notin N_\varepsilon(J(a_k, x_k)). \end{cases}$$

ノード配列が 2 次元の場合，次の意味において 1 次元ノード配列の場合に成立した単調性の保存が成り立たなくなる．例えば，d をユークリッド距離として，学習範囲を

$$N_{\sqrt{2}}(J(a_k, x_k)) = \left\{(i, j) \in I \;\middle|\; d((i, j), J(a_k, x_k)) \le \sqrt{2}\right\}$$

とするとき，各成分を固定した場合，すべての $a_k(i, \cdot), a_k(\cdot, j)$ が（単調な数列という意味において）単調であっても，$a_{k+1}(i, \cdot)$ および $a_{k+1}(\cdot, j)$ は単調であるとは限らない．

参考文献

[1] P. Billingsley, *Probability and Measure*, John Wiley & Sons, 1979.

[2] R. M. Dudley, *Real Analysis and Probability*, Wadsworth & Brooks, 1989.

[3] E. Erwin, K. Obermayer and K. Schulten, *Self-organization maps: stationary states, metastability and convergence rate*, Bio. Cybern., 67 (1992), pp. 35–45

[4] E. Erwin, K. Obermayer and K. Schulten, *Self-organization maps: ordering, convergence properties and energy functions*, Bio. Cybern., 67 (1992), pp. 47–55

[5] T. Kohonen, *Self-Organizing Maps, Third Edition*, Springer, 2001.

[6] P. L. Zador, *Asymptotic quantization error of continuous signals and the quantization dimension*, IEEE Trans. Inform. Theory, vol. IT-28, No. 2, March (1982), pp. 139–149.