

Ordered states and absorbing states in basic self-organizing maps (自己組織化マップモデルにおける整列状態と吸収状態について)

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

1. 基本的な自己組織化マップモデル

本報告は Kohonen 型アルゴリズム [7] として知られている自己組織化マップモデルにおける整列化とモデル関数の性質に関する一つの理論的考察である。自己組織化マップモデルにおけるノードの配列とノードの値との間に現れるある種の規則性について、モデルの吸収状態と整列化の形成過程における確率的挙動に注目して考察する。

自己組織化マップは非常に実用的であり広範囲に応用例を有し、アルゴリズムも非常にシンプルであるが、その数学的構造はあまり明らかではない。

本報告では、自己組織化マップモデルをノード、ノードの値、インプット、学習プロセスの 4 つの要素によって、以下の様に定義する。

- (i) I をすべてのノードの集合とする。 I は、距離 d をもつある距離空間の加算部分集合とする。
- (ii) 各ノードは、それぞれ 1 つの値をもつ。 V をノードの値の空間とする。 V はノルム空間であると仮定する。 V におけるノルムを $\|\cdot\|$ とする。 $m(i)$ をノード i の値として、その対応 $m: I \rightarrow V$ をモデル関数 (model function, reference function) と呼ぶことにする。 また、 M をモデル関数の全体、 $m_0: I \rightarrow V$ を初期モデル関数とする。
- (iii) $X \subset V$ を入力集合とする。 $x_0, x_1, x_2, \dots \in X$ を入力列とする。
- (iv) 学習プロセスとして以下の 2 つを仮定する。

学習プロセス L_m

- (a) 学習範囲:

$$I(m_k, x_k) = \left\{ i^* \in I \mid \|m_k(i^*) - x_k\| = \inf_{i \in I} \|m_k(i) - x_k\| \right\} \\ (m_k \in M, x_k \in X), \\ N_1(i) = \{j \in I \mid d(j, i) \leq \varepsilon\} \quad (i \in I).$$

- (b) 学習率: $0 \leq \alpha \leq 1$.

- (c) 更新後の値:

$$m_{k+1}(i) = \begin{cases} (1 - \alpha)m_k(i) + \alpha x_k & \text{if } i \in \bigcup_{i^* \in I(m_k, x_k)} N_\varepsilon(i^*), \\ m_k(i) & \text{if } i \notin \bigcup_{i^* \in I(m_k, x_k)} N_\varepsilon(i^*), \end{cases} \quad k = 0, 1, 2, \dots$$

学習プロセス L_m

(a) 学習範囲:

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

$$(m_k \in M, x_k \in X),$$

$$N_\varepsilon(i) = \{j \in I \mid d(j, i) \leq \varepsilon\} \quad (i \in I).$$

(b) 学習率: $0 \leq \alpha \leq 1$.

(c) 更新後の値:

$$m_{k+1}(i) = \begin{cases} (1 - \alpha)m_k(i) + \alpha x_k & \text{if } i \in N_\varepsilon(J(m_k, x_k)), \\ m_k(i) & \text{if } i \notin N_\varepsilon(J(m_k, x_k)), \end{cases} \quad k = 0, 1, 2, \dots$$

上記の2つの学習プロセスは、初期ノードの値によっては、学習反復の初期段階において学習結果に違いが生ずるが、多くの場合、それ以降の反復においては、あまり差が現れない。

2. $\mathbb{R}$ 値ノード, 1次元ノード配列モデル

ここでは、最も単純な自己組織化マップモデルである、 $\mathbb{R}$ 値ノード, 1次元ノード配列の場合について述べる。

- (i) 有限個のノードを仮定する. $I = \{1, 2, \dots, n\} \subset \mathbb{N}$.
 - (ii) ノード値の空間を $\mathbb{R}$ (ユークリッドノルム) とする. $m_0 = [m_0(1), m_0(2), \dots, m_0(n)]$ などと記すことにする.
 - (iii) $x_0, x_1, x_2, \dots \in X \subset \mathbb{R}$ を入力列とする.
 - (iv) 以下の学習プロセスの一方を仮定する.
- 学習プロセス L_A (1次元配列, $\mathbb{R}$ -値ノード, $\varepsilon = 1$)

(a) 学習範囲:

$$I(m_k, x_k) = \left\{ i^* \in I \mid |m_k(i^*) - x_k| = \inf_{i \in I} |m_k(i) - x_k| \right\}$$

$$(m_k \in M, x_k \in X),$$

$$N_1(i) = \{j \in I \mid |j - i| \leq 1\} \quad (i \in I).$$

(b) 学習率: $0 \leq \alpha \leq 1$.

(c) 更新後の値:

$$m_{k+1}(i) = \begin{cases} (1 - \alpha)m_k(i) + \alpha x_k & \text{if } i \in \bigcup_{i^* \in I(m_k, x_k)} N_1(i^*), \\ m_k(i) & \text{if } i \notin \bigcup_{i^* \in I(m_k, x_k)} N_1(i^*), \end{cases} \quad k = 0, 1, 2, \dots$$

学習プロセス L_m (1次元配列, $\mathbb{R}$ -値ノード, $\varepsilon = 1$)

(a) 学習範囲:

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

$$(m_k \in M, x_k \in X),$$

$$N_1(i) = \{j \in I \mid |j - i| \leq 1\} \quad (i \in I).$$

(b) 学習率: $0 \leq \alpha \leq 1$.

(c) 更新後の値:

$$m_{k+1}(i) = \begin{cases} (1 - \alpha)m_k(i) + \alpha x_k & \text{if } i \in N_1(J(m_k, x_k)), \\ m_k(i) & \text{if } i \notin N_1(J(m_k, x_k)), \end{cases} \quad k = 0, 1, 2, \dots$$

今, n 個のノード $1, 2, \dots, n$ があり, そのそれぞれに対してノードの値 $m_0(1), m_0(2), \dots, m_0(n)$ 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, $m_0(1), m_0(2), \dots, m_0(n)$ 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

$$m_1(i) = (1 - \alpha)m_0(i) + \alpha x_1$$

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, $m_1(i) = m_0(i)$ となる. インプット $x_1, x_2, x_3, \dots$ 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, 同時にモデル関数 $m_1, m_2, m_3, \dots$ が逐次に生成される.

このような学習を十分な回数, 繰り返したとき, モデル関数において, 単調性等, 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. 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, 単調化等の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, 多くの実問題へ応用されている.

3. モデルの吸収状態について

次の定理は, 自己組織化マップモデルにおけるモデル関数の単調性保存に関する基本的な結果である.

Theorem 1 学習プロセス L_A を仮定する. モデル関数 $m_1, m_2, m_3, \dots$ に対して以下が成り立つ.

- (i) モデル関数 m_k が I 上で単調増加であるならば, モデル関数 m_{k+1} も I 上で単調増加である.
- (ii) モデル関数 m_k が I 上で単調減少であるならば, モデル関数 m_{k+1} も I 上で単調減少である.
- (iii) モデル関数 m_k が I 上で狭義単調増加であるならば, モデル関数 m_{k+1} も I 上で狭義単調増加である.
- (iv) モデル関数 m_k が I 上で狭義単調減少であるならば, モデル関数 m_{k+1} も I 上で狭義単調減少である.

Theorem 2 学習プロセス L_m を仮定する. モデル関数 $m_1, m_2, m_3, \dots$ に関して, 次が成り立つ.

- (i) モデル関数 m_k が I 上で狭義単調増加であるならば, モデル関数 m_{k+1} も I 上で狭義単調増加である.
- (ii) モデル関数 m_k が I 上で狭義単調減少であるならば, モデル関数 m_{k+1} も I 上で狭義単調減少である.

ここでの単調増加性, 単調減少性のように, モデル関数が一度その状態になると, その状態が保存されるという意味において, このような状態を自己組織化マップモデルの吸収状態と呼ぶことにする. モデルの吸収状態として後で述べる意味での準凸性, 準凹性などがある [6].

4. 2次元ノード配列の場合

ここでは, 以下の自己組織化マップモデルを考える.

- (i) ノード集合 $I = I_1 \times I_2$. ただし

$$I_1 = \{1, 2, \dots, N_1\}, I_2 = \{1, 2, \dots, N_2\}, d_I((i, j), (k, l)) = |i - k| + |j - l|$$

- (ii) ノード値空間 V は内積 $\langle \cdot, \cdot \rangle$ をもつ内積空間とする.
- (iii) $x_0, x_1, x_2, \dots \in X \subset V$ は入力列とする.
- (iv) 学習プロセス

- (a) 学習範囲:

$$I(m, x) = \left\{ i^* \in I \mid \|m(i^*) - x\| = \inf_{i \in I} \|m(i) - x\| \right\} \quad (m \in M, x \in X),$$

$$N_1(i) = \{j \in I \mid d_I(j, i) \leq 1\} \quad (i \in I).$$

- (b) 学習率: $0 \leq \alpha \leq 1$.

- (c) 更新後の値:

$$m'(i) = \begin{cases} (1 - \alpha)m(i) + \alpha x & \text{if } i \in \bigcup_{i^* \in I(m, x)} N_1(i^*), \\ m(i) & \text{if } i \notin \bigcup_{i^* \in I(m, x)} N_1(i^*), \end{cases} \quad k = 0, 1, 2, \dots$$

Condition 5 すべての $(i, j) \in I$ に対して

$$\|m(i+1, j) - m(i, j)\| \leq \|m(i+2, j) - m(i, j)\|$$

$$\|m(i-1, j) - m(i, j)\| \leq \|m(i-2, j) - m(i, j)\|$$

$$\|m(i, j+1) - m(i, j)\| \leq \|m(i, j+2) - m(i, j)\|$$

$$\|m(i, j-1) - m(i, j)\| \leq \|m(i, j-2) - m(i, j)\|$$

が成り立つ.

ここで, Condition 5 はモデルの吸収状態にならないが, 以下の性質をもつことがわかる.

Theorem 3 上の 2次元ノード配列モデルを仮定する. 等式

$$\|m(i^*, j^*) - x\| = \min_{(i,j) \in I} \|m(i, j) - x\|.$$

を満たす (i^*, j^*) が一意に存在すると仮定する. このとき, 以下が成り立つ.

- (1) (a) $(i, j), (i+1, j), (i+2, j) \notin N_1(i^*, j^*)$ に対して $\|m(i+1, j) - m(i, j)\| \leq \|m(i+2, j) - m(i, j)\|$ ならば, すべての α, x に対して $\|m'(i+1, j) - m'(i, j)\| \leq \|m'(i+2, j) - m'(i, j)\|$ が成り立つ.
- (b) $(i, j), (i-1, j), (i-2, j) \notin N_1(i^*, j^*)$ に対して $\|m(i-1, j) - m(i, j)\| \leq \|m(i-2, j) - m(i, j)\|$ ならば, すべての α, x に対して $\|m'(i-1, j) - m'(i, j)\| \leq \|m'(i-2, j) - m'(i, j)\|$ が成り立つ.
- (c) $(i, j), (i, j+1), (i, j+2) \notin N_1(i^*, j^*)$ に対して $\|m(i, j+1) - m(i, j)\| \leq \|m(i, j+2) - m(i, j)\|$ ならば, すべての α, x に対して $\|m'(i, j+1) - m'(i, j)\| \leq \|m'(i, j+2) - m'(i, j)\|$ が成り立つ.
- (d) $(i, j), (i, j-1), (i, j-2) \notin N_1(i^*, j^*)$ に対して $\|m(i, j-1) - m(i, j)\| \leq \|m(i, j-2) - m(i, j)\|$ ならば, すべての α, x に対して $\|m'(i, j-1) - m'(i, j)\| \leq \|m'(i, j-2) - m'(i, j)\|$ が成り立つ.
- (2) (a) $\|m(i^*, j^*) - m(i^* - 1, j^*)\| \leq \|m(i^* + 1, j^*) - m(i^* - 1, j^*)\|$ ならば, すべての α, x に対して $\|m'(i^*, j^*) - m'(i^* - 1, j^*)\| \leq \|m'(i^* + 1, j^*) - m'(i^* - 1, j^*)\|$ が成り立つ.
- (b) $\|m(i^*, j^*) - m(i^* + 1, j^*)\| \leq \|m(i^* - 1, j^*) - m(i^* + 1, j^*)\|$ ならば, すべての α, x に対して $\|m'(i^*, j^*) - m'(i^* + 1, j^*)\| \leq \|m'(i^* - 1, j^*) - m'(i^* + 1, j^*)\|$ が成り立つ.
- (c) $\|m(i^*, j^*) - m(i^*, j^* - 1)\| \leq \|m(i^*, j^* + 1) - m(i^*, j^* - 1)\|$ ならば, すべての α, x に対して $\|m'(i^*, j^*) - m'(i^*, j^* - 1)\| \leq \|m'(i^*, j^* + 1) - m'(i^*, j^* - 1)\|$ が成り立つ.
- (d) $\|m(i^*, j^*) - m(i^*, j^* + 1)\| \leq \|m(i^*, j^* - 1) - m(i^*, j^* + 1)\|$ ならば, すべての α, x に対して $\|m'(i^*, j^*) - m'(i^*, j^* + 1)\| \leq \|m'(i^*, j^* - 1) - m'(i^*, j^* + 1)\|$ が成り立つ.
- (3) (a) $\|m(i^* + 1, j^*) - m(i^*, j^*)\| \leq \|m(i^* + 2, j^*) - m(i^*, j^*)\|$ ならば, すべての α, x に対して $\|m'(i^* + 1, j^*) - m'(i^*, j^*)\| \leq \|m'(i^* + 2, j^*) - m'(i^*, j^*)\|$ が成り立つ.
- (b) $\|m(i^* - 1, j^*) - m(i^*, j^*)\| \leq \|m(i^* - 2, j^*) - m(i^*, j^*)\|$ ならば, すべての α, x に対して $\|m'(i^* - 1, j^*) - m'(i^*, j^*)\| \leq \|m'(i^* - 2, j^*) - m'(i^*, j^*)\|$ が成り立つ.
- (c) $\|m(i^*, j^* + 1) - m(i^*, j^*)\| \leq \|m(i^*, j^* + 2) - m(i^*, j^*)\|$ ならば, すべての α, x に対して $\|m'(i^*, j^* + 1) - m'(i^*, j^*)\| \leq \|m'(i^*, j^* + 2) - m'(i^*, j^*)\|$ が成り立つ.
- (d) $\|m(i^*, j^* - 1) - m(i^*, j^*)\| \leq \|m(i^*, j^* - 2) - m(i^*, j^*)\|$ ならば, すべての α, x に対して $\|m'(i^*, j^* - 1) - m'(i^*, j^*)\| \leq \|m'(i^*, j^* - 2) - m'(i^*, j^*)\|$ が成り立つ.

PROOF. (1),(2) は明らかである. (3) を示す. $m_0 = m(i^*, j^*), m_1 = m(i^* + 1, j^*), m_2 = m(i^* + 2, j^*)$ とおくと, $\|m_1 - m_0\| \leq \|m_2 - m_0\|$ が成り立つならば, 次の不等式が成り

立つ.

$$\begin{aligned}
& \|m'_2 - m'_0\|^2 - \|m'_1 - m'_0\|^2 \\
&= \|m_2 - (1-\alpha)m_0 - \alpha x\|^2 - \|(1-\alpha)m_1 + \alpha x - (1-\alpha)m_0 - \alpha x\|^2 \\
&= \|(1-\alpha)(m_2 - m_0) + \alpha(m_2 - x)\|^2 - \|(1-\alpha)(m_1 - m_0)\|^2 \\
&= (1-\alpha)^2 \|m_2 - m_0\|^2 + 2(1-\alpha)\alpha \langle m_2 - m_0, m_2 - x \rangle + \alpha^2 \|m_2 - x\|^2 \\
&\quad - (1-\alpha)^2 \|m_1 - m_0\|^2 \\
&= (1-\alpha)^2 (\|m_2 - m_0\|^2 - \|m_1 - m_0\|^2) \\
&\quad + 2(1-\alpha)\alpha (\|m_2 - x\|^2 - \langle m_0 - x, m_2 - x \rangle) + \alpha^2 \|m_2 - x\|^2 \\
&\geq (1-\alpha)^2 (\|m_2 - m_0\|^2 - \|m_1 - m_0\|^2) \\
&\quad + 2(1-\alpha)\alpha (\|m_2 - x\|^2 - \|m_0 - x\| \|m_2 - x\|) + \alpha^2 \|m_2 - x\|^2 \\
&\geq (1-\alpha)^2 (\|m_2 - m_0\|^2 - \|m_1 - m_0\|^2) \\
&\quad + 2(1-\alpha)\alpha (\|m_2 - x\|^2 - \|m_2 - x\| \|m_2 - x\|) + \alpha^2 \|m_2 - x\|^2 \geq 0
\end{aligned}$$

同様に $m_0 = m(i^*, j^*)$, $m_1 = m(i^* - 1, j^*)$, $m_2 = m(i^* - 2, j^*)$, または $m_0 = m(i^*, j^*)$, $m_1 = m(i^*, j^* + 1)$, $m_2 = m(i^*, j^* + 2)$, または $m_0 = m(i^*, j^*)$, $m_1 = m(i^*, j^* - 1)$, $m_2 = m(i^*, j^* - 2)$ とおくことにより, 残りの命題が成立することを示すことができる. $\square$

定理3における仮定は, 十分な学習を繰り返した後においては強い条件ではない. 多くの数値モデルにおいて殆ど成り立つ.

5. 1次元ノード配列, $\mathbb{R}^2$ -値ノードの場合について

ここでは, 以下のモデルを仮定する.

- (i) ノード集合 $I = \{1, 2, \dots, n\}$. ただし, 距離を $d_I(i, j) = |i - j|$ によって定義する.
- (ii) モデル関数 $m : I \rightarrow \mathbb{R}^2$. ここで, 各 $x = (x_1, x_2)$, $y = (y_1, y_2)$ に対して, $\|x - y\| = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$ とする.
- (iii) 入力 $x_0, x_1, x_2, \dots \in X \subset \mathbb{R}^2$.
- (iv) 学習プロセス L_A (1次元配列, $\varepsilon = 1$)

(a) 学習範囲:

$$\begin{aligned}
I(m, x) &= \left\{ i^* \in I \mid \|m(i^*) - x\| = \inf_{i \in I} \|m(i) - x\| \right\} \\
&\quad (m \in M, x \in X),
\end{aligned}$$

$$N_1(i) = \{j \in I \mid d_I(j, i) \leq 1\} \quad (i \in I).$$

(b) 学習率: $0 \leq \alpha \leq 1$.

(c) 更新後の値:

$$m'(i) = \begin{cases} (1-\alpha)m(i) + \alpha x & \text{if } i \in \bigcup_{i^* \in I(m, x)} N_1(i^*), \\ m(i) & \text{if } i \notin \bigcup_{i^* \in I(m, x)} N_1(i^*), \end{cases} \quad k = 0, 1, 2, \dots$$

以下の1次元ノード配列, $\mathbb{R}^2$ -値ノードをもつ自己組織化マップモデルの数値例について考察する.

Example 1 以下の15個のノードをもつ自己組織化マップモデルを考える.

- (1) ノード集合 $I = \{1, 2, 3, \dots, 15\}$.
- (2) 初期モデル関数

$$m_0 = [(2, 5), (9, 0), (1, 9), (10, 7), (9, 8), (5, 8), (8, 3), (8, 5), (5, 5), (7, 1), \\ (3, 1), (5, 0), (2, 8), (1, 10), (8, 6)]$$

各ノードの初期値と初期入力 x_0 を図示すると以下の様になる.

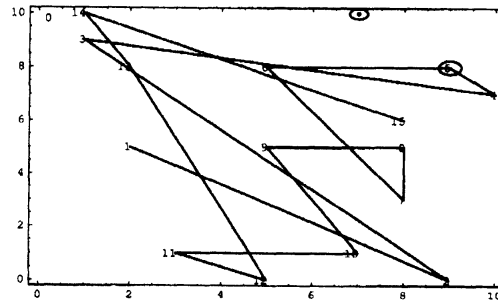


図 1: m_0

- (3) 入力として, $\{0, 1, 2, \dots, 10\} \times \{0, 1, 2, \dots, 10\}$ 上の離散一様分布に従い, 発生させた.

$$x = (7, 10), (5, 2), (7, 3), (10, 8), (2, 7), (0, 0), (8, 2), (7, 5), (8, 3), (4, 4), \\ (10, 1), (8, 2), (6, 4), (7, 2), (0, 9), (7, 2), (6, 6), (4, 10), (1, 3), (2, 8), \dots$$

- (4) 学習プロセス L_A を仮定する. (学習率: $\alpha = \frac{1}{2}$)

このとき, 上の学習プロセスにより, モデル関数は以下の様に更新される (図 2).

$$m_1 = [(2, 5), (9, 0), (1, 9), (8.5, 8.5), (8, 9), (6, 9), (8, 3), (8, 5), (5, 5), (7, 1), (3, 1), \\ (5, 0), (2, 8), (1, 10), (8, 6)]$$

$$m_2 = [(2, 5), (9, 0), (1, 9), (8.5, 8.5), (8, 9), (6, 9), (8, 3), (8, 5), (5, 5), (7, 1), (4, 1.5), \\ (5, 1), (3.5, 5), (1, 10), (8, 6)]$$

$$m_3 = [(2, 5), (9, 0), (1, 9), (8.5, 8.5), (8, 9), (6.5, 6), (7.5, 3), (7.5, 4), (5, 5), (7, 1), \\ (4, 1.5), (5, 1), (3.5, 5), (1, 10), (8, 6)]$$

.....

□

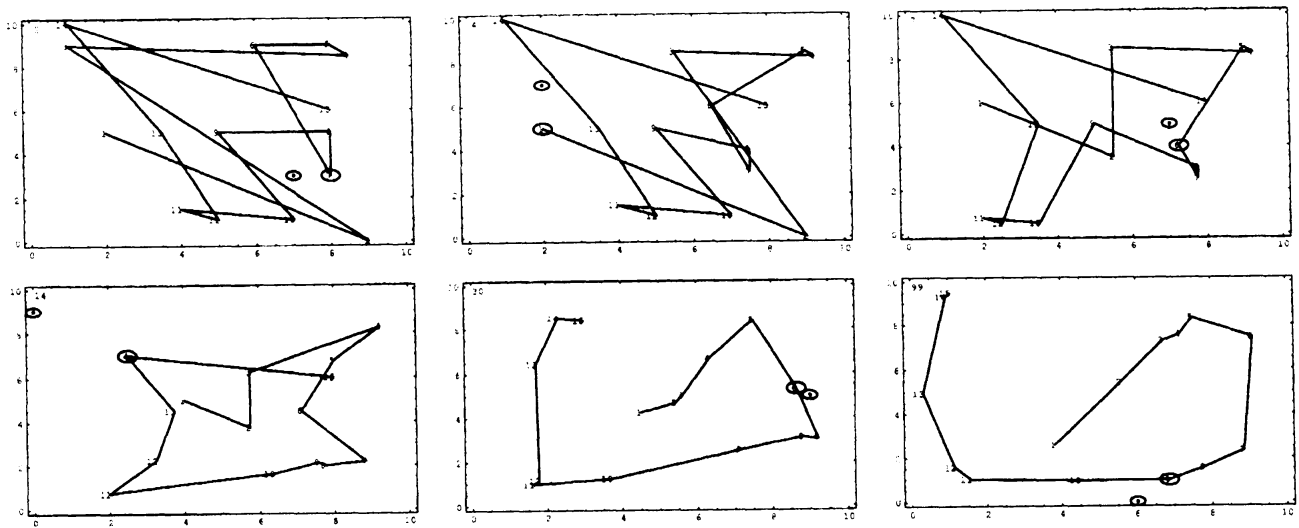


図 2: 左上から右に向かって $m_2, m_4, m_7, m_{14}, m_{30}, m_{99}$

Example 2 例 1 のモデルにおけるモデル関数 m_{14} に注目する. モデル関数

$$m_{14} = [(4, 5), (5.75, 3.75), (5.75, 6.25), (9.25, 8.25), (8, 6.75), (7.125, 4.5), \\ (8.84375, 2.1875), (7.73438, 2), (7.5625, 2.125), (6.375, 1.625), (2, 0.75), \\ (3.25, 2.25), (3.75, 4.5), (2.5, 7), (8, 6)]$$

は, 入力 $x_{14} = (2, 4)$ により, 次のように更新される.

$$m'_{14} = [(4, 5), (5.75, 3.75), (5.75, 6.25), (9.25, 8.25), (8, 6.75), (7.125, 4.5), \\ (8.84375, 2.1875), (7.73438, 2), (7.5625, 2.125), (6.375, 1.625), (2, 0.75), \\ (2.625, 3.125), (2.875, 4.25), (2.25, 5.5), (8, 6)]$$

このとき, 各ノードの値としての点 $m_{14}(1), m_{14}(2), \dots, m_{14}(15) \in \mathbb{R}^2$ を順に結合してできる折れ線において, 交点の数に注目すると, m_{14} の交点の数は 2 であるが, 更新後の値 m'_{14} でも, 交点の数は 2 のままである. 例えば, 入力を $x_{14} = (2, 6)$ にすると

$$m'_{14} = [(4, 5), (5.75, 3.75), (5.75, 6.25), (9.25, 8.25), (8, 6.75), (7.125, 4.5), \\ (8.84375, 2.1875), (7.73438, 2), (7.5625, 2.125), (6.375, 1.625), (2, 0.75), \\ (3.25, 2.25), (2.875, 5.25), (2.25, 6.5), (5, 6)]$$

と更新され, 同様に交点を数えると更新後の値 m'_{14} において, 交点の数は 0 となる. 入力を $x_{14} = (10, 6)$ にすると

$$m'_{14} = [(4, 5), (5.75, 3.75), (5.75, 6.25), (9.25, 8.25), (8, 6.75), (7.125, 4.5), \\ (8.84375, 2.1875), (7.73438, 2), (7.5625, 2.125), (6.375, 1.625), (2, 0.75), \\ (3.25, 2.25), (3.75, 4.5), (6.25, 6.5), (9, 6)]$$

と更新され、更新後の値 m'_{14} において、交点の数は3となる。

例1のモデルにおける各更新回数でのノードの値に対して、入力 x を以下の様に、各座標 0.1 刻みで変化させた場合について考える。

$$x = (p, q), \quad p, q = 0, 0.1, 0.2, \dots, 9.9, 10$$

入力 x を変化させた場合におけるモデル関数 m に対する更新後のモデル関数 m' の交点数の増減について、増加する、変化していない、減少する、それぞれの入力 x の頻度を調べると以下の様になる。

モデル関数	交点の数の増減					
	増加		変化していない		減少	
	度数	相対度数	度数	相対度数	度数	相対度数
m_0	0	0	4524	0.443486	5677	0.556514
m_1	0	0	3510	0.344084	6691	0.655916
m_2	0	0	3489	0.342025	6712	0.657975
m_3	0	0	3911	0.383394	6290	0.616606
m_4	27	0.0026468	3117	0.305558	7057	0.691795
m_{14}	447	0.0438192	6698	0.656602	3056	0.299578
m_{15}	1	0.0000980296	10200	0.999902	0	0
m_{16}	0	0	10201	1	0	0
m_{50}	0	0	10201	1	0	0

また、入力 x を変化させた場合における交点の増減頻度と更新回数の関係を調べると以下の様になる (図3)。

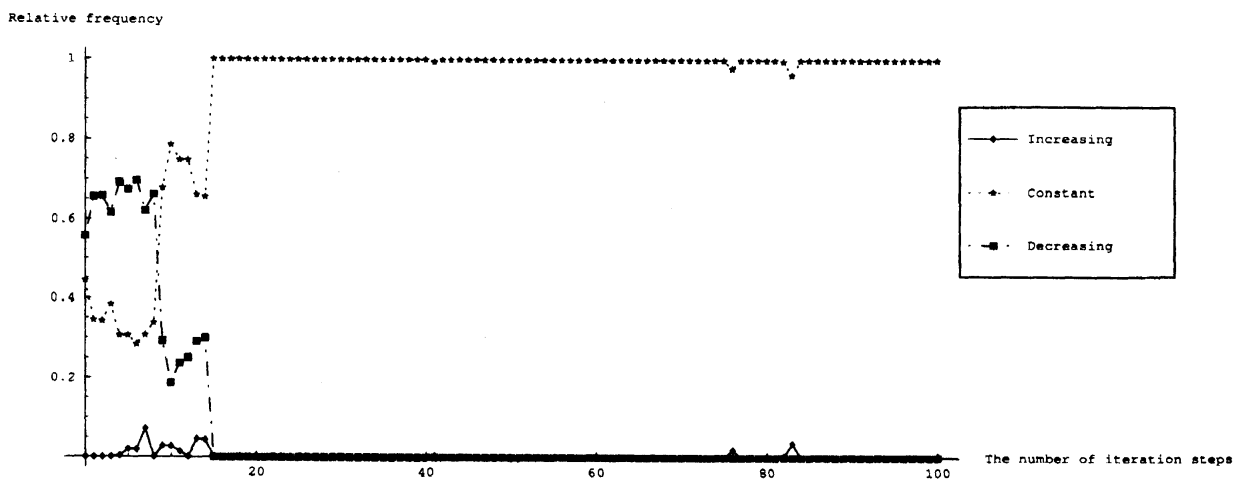


図3: 交点の増減の相対度数と更新回数の関係 (学習率 $\alpha = 0.5$)

更新回数が十分大きくなると、交点の数が増え、交点の数が増えない相対度数が殆ど1になることが観測される。

同様に、学習率が0.1の場合について調べると、学習率0.5の場合と同様の傾向となるが、学習が緩やかになるので、交点の数が増え、交点の数が増えない相対度数が（殆ど）1となる更新回数は学習率0.5の場合よりも多くなる。

また、モデル関数 m_{14} を固定して、学習率を0から1まで変化させた場合、学習率を大きくすると、交点の数が増える相対度数は増加する傾向にあるが、同時に交点の数が増える相対度数も増加してしまう傾向にあることが観測される。□

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] W. Fujiwara, E. Itou, M. Hoshino, I. Kaku, A. Sakusabe, M. Sasaki and H. Kosaka, *A study on the effective method of external inspecting using a neural network approach*, Proceedings of 6th ICIM (2002), pp. 369–375.
- [6] M. Hoshino, Y. Kimura and I. Kaku, *Quasi-convexity and monotonicity of model function of node in self-organizing maps*, Proceedings of 8th ICIM (2006), pp. 1173–1177.
- [7] T. Kohonen, *Self-Organizing Maps, Third Edition*, Springer, 2001.
- [8] W. Takahashi, *Nonlinear Functional Analysis*, Yokohama publishers, 2000.
- [9] K. Tanaka, *凸解析と最適化理論*, 牧野書店, 1994.
- [10] 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.