

# ニューロンの近傍関係を考慮した自己組織化 Self-Organization Considering Neighborhood Relationship between Neurons

松下春奈<sup>†</sup>      西尾芳文<sup>†</sup>  
<sup>†</sup>徳島大学 工学部 電気電子工学科

Haruna MATSUSHITA<sup>†</sup>

Yoshifumi NISHIO<sup>†</sup>

<sup>†</sup>Department of Electrical and Electronic Engineering, Tokushima University

## 1 はじめに

自己組織化マップ (Self-Organizing Map: SOM) [1] は教師無しニューラルネットワークの一つで、入力統計的な特徴を反映したマップを自己組織的に得られる性質を持つ。このことから、クラスタリングの分野において非常に注目されている。SOMの学習過程においては、最も入力ベクトルに近い重みベクトルを持つニューロンを勝者ニューロンとし、勝者ニューロンとその近傍ニューロンの重みベクトルが更新される。このとき、近傍ニューロンは入力ベクトルとの距離で決定されるのではなく、勝者ニューロンとのマップ上での結合関係の距離で選ばれる。このことから、離れた場所にいくつかのクラスタがあるような入力に対してSOMを適用した場合、そのクラスタ間に不活性ニューロンが存在してしまう。入力データが存在しないエリアに不活性ニューロンがあるために、クラスタ間にデータがあると勘違いしてしまうことが懸念される。

その一方で、実社会において、隣の家とはいえ距離的には自分の家の近くではない場合が多々ある。つまり、“近傍”がいつも“本当の近傍”であるとは限らないことを意味している。さらに、近傍関係は一定ではなく、時間と共に常に変化し続けていることから、状況に応じて近傍関係を柔軟に変化させることが重要であると考えられる。

また、実際の脳においてもシナプスの強さは一定ではない。1985年に発表されたGrowing Grid [2]では、ニューロン数を増やすことによりニューロン間の近傍距離を増加させている。このように、ニューロン数を増加させたりライバルニューロンを考慮するアルゴリズム [3] はあっても、シナプスの強さ自体

を変化させた研究はあまり報告されていない。

そこで本研究では、ニューロン間の誤り近傍度を考慮する自己組織化マップ (SOM with False-Neighbor degree between neurons: FN-SOM) を提案する。誤り近傍度はFN-SOMのニューロンの行間と列間に割り当てられている。アルゴリズムでは、一度も勝者になったことのないニューロン $q$ を探し、 $q$ と1近傍で直接結合されているニューロンのうち、最も $q$ から離れているニューロンを $q$ の“誤り近傍ニューロン”とする。誤り近傍度の初期値は0に設定されるが、学習と共に徐々に増加し、重みベクトル更新の際にマップ上のノード間の距離の重みとして作用する。つまり、FN-SOMは状況と入力データの形によって近傍関係を柔軟に変化させて学習する。

また、FN-SOMを2次元データ、3次元データ、実データに対して適用し、FN-SOMの振る舞いを調査する。またその結果を数値的にも評価し、FN-SOMが入力の分布状態を最も反映したマップを得ることを確認する。

## 2 誤り近傍度を考慮した SOM (FN-SOM)

提案手法であるFN-SOMについて詳しく説明する。FN-SOMは2次元グリッド上に配置された $n \times m$ 個のニューロンによって構成されており、各ニューロン $i$ は $d$ 次元の重みベクトル $w_i = (w_{i1}, w_{i2}, \dots, w_{id})$  ( $i = 1, 2, \dots, nm$ ) を備えている。FN-SOMの行間に誤り近傍度 $R_r$  ( $1 \leq r \leq n-1$ ) を、列間に誤り近傍度 $C_k$  ( $1 \leq k \leq m-1$ ) を割り当て、それぞれの初期値は0に設定する。学習のフローチャートを図1に示す。

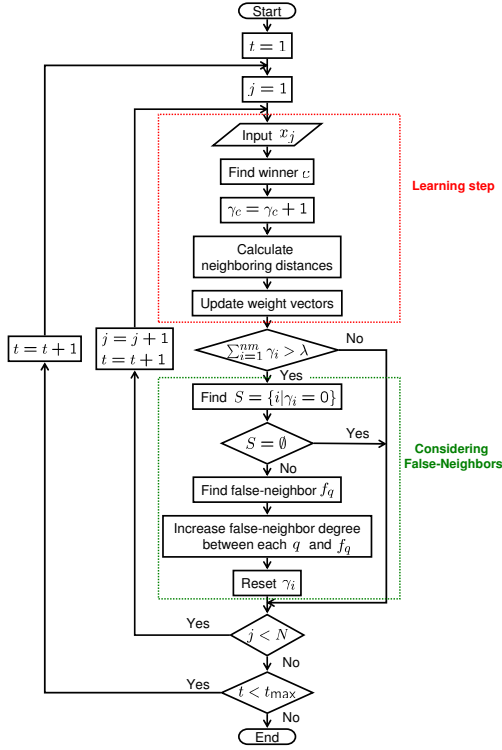


図1: FN-SOMの学習アルゴリズムのフローチャート

### Learning Step

(FN-SOM1)  $d$ 次元入力ベクトル  $\mathbf{x}_j = (x_{j1}, x_{j2}, \dots, x_{jd})$  ( $j = 1, 2, \dots, N$ ) を1つ選び、全ニューロンに入力する。

(FN-SOM2) 全ての重みベクトルと入力ベクトル  $\mathbf{x}_j$  間の距離を計算し、最も  $\mathbf{x}_j$  に近い重みベクトルを持つニューロンを勝者ニューロン  $c$  とする。

$$c = \arg \min_i \{\|\mathbf{w}_i - \mathbf{x}_j\|\} \quad (1)$$

ここで、 $\|\cdot\|$  はユークリッド距離を表す。

(FN-SOM3) 勝者ニューロン  $c$  の勝者回数を増加させる:  $\gamma_c^{\text{new}} = \gamma_c^{\text{old}} + 1$ 。

(FN-SOM4) 勝者ニューロン  $c$  と各ニューロンとの近傍距離を計算する。例えば、 $r_1$ 行  $k_1$ 列に位置するニューロン  $s_1$  と、 $r_2$ 行  $k_2$ 列に位置するニューロン

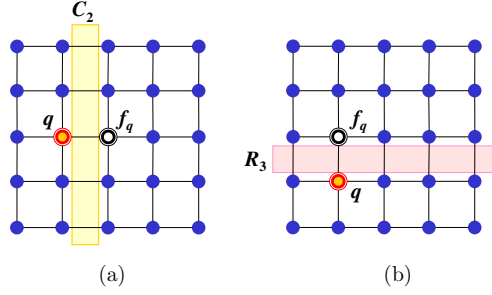


図2: 誤り近傍度の増加. (a)  $q$ が3行2列に、その誤り近傍  $f_q$ が3行3列に位置する場合、2列と3列間の誤り近傍度  $C_2$ が式(6)に従って増加される. (b)  $q$ が4行2列に、その誤り近傍  $f_q$ が3行2列に位置する場合、3行と4行間の誤り近傍度  $R_3$ が式(7)に従って増加される。

$s_2$ 間の近傍距離は、次の式によって定義される。

$$d_f(s_1, s_2) = \left( |r_1 - r_2| + \sum_{r=r_1}^{r_2-1} R_r \right)^2 + \left( |k_1 - k_2| + \sum_{k=k_1}^{k_2-1} C_k \right)^2 \quad (2)$$

このとき、 $r_1 < r_2$ 、 $k_1 < k_2$  とする。つまり、 $\sum_{r=r_1}^{r_2-1} R_r$  は  $r_1$ 行と  $r_2$ 行間の誤り近傍度の合計を、 $\sum_{k=k_1}^{k_2-1} C_k$  は  $k_1$ 列と  $k_2$ 列間の誤り近傍度の合計を、それぞれ意味している。

(FN-SOM5) ニューロンの重みベクトルを更新する。

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + h_{F_{c,i}}(t)(\mathbf{x}_j - \mathbf{w}_i(t)) \quad (3)$$

$t$ は学習回数、 $h_{F_{c,i}}(t)$ はFN-SOMの近傍関数である。

$$h_{F_{c,i}}(t) = \alpha(t) \exp\left(-\frac{d_f(c, i)}{2\sigma^2(t)}\right) \quad (4)$$

$\alpha(t)$ と  $\sigma(t)$ は、時間と共に減少する。

(FN-SOM6) もし  $\sum_{i=1}^m \gamma_i \geq \lambda$ であれば、(FN-SOM7)から(FN-SOM10)のステップに従い誤り近傍度を増加させる。そうでない場合は(FN-SOM11)へ進む。つまり、 $\lambda$ 個の入力データに対して学習が行なわれる度に、誤り近傍を考慮する。

### Considering False-Neighbors

(FN-SOM7) 一度も勝者にならなかったことのないニューロンの集合  $S$ をみつける:  $S = \{i \mid \gamma_i = 0\}$ 。もし、勝

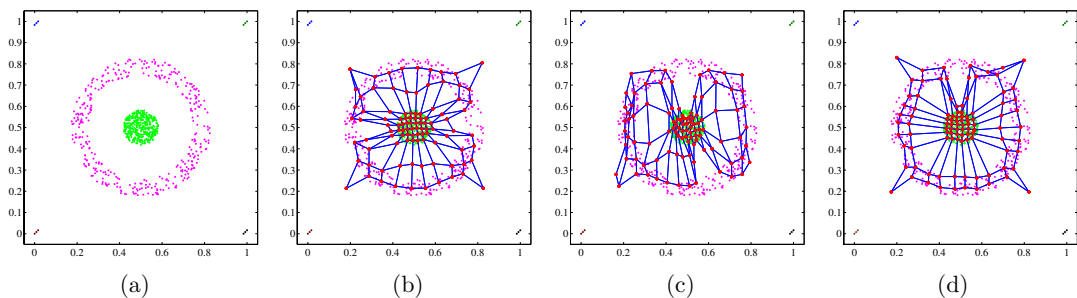


図 3: Target データに対する 3 手法の学習結果. (a) 入力データ. (b) 従来の SOM. (c) Growing Grid. (d) FN-SOM.

者になったことのないニューロンが存在しない, つまり  $S = \emptyset$  である場合, 誤り近傍度は増加させず (FN-SOM11) へ進む.

**(FN-SOM8)** 集合  $S$  内の各ニューロン  $q$  に対する誤り近傍  $f_q$  を選ぶ.  $f_q$  は  $q$  の 1 近傍ニューロン  $N_{q_1}$  のうち, 最も  $q$  から遠い重みベクトルを持つニューロンとする.

$$f_q = \arg \max_i \{\|\mathbf{w}_i - \mathbf{w}_q\|\}, \quad q \in S, i \in N_{q_1} \quad (5)$$

**(FN-SOM9)**  $q$  とその誤り近傍  $f_q$  間の誤り近傍度  $R_r$ , もしくは  $C_k$  を増加させる. 図 2(a) のように,  $q$  が  $r$  行  $k$  列,  $f_q$  が  $r$  行  $(k+1)$  列に位置する場合,  $k$  列と  $(k+1)$  列間の誤り近傍度  $C_k$  を次式に従い増加させる.

$$C_k^{\text{new}} = C_k^{\text{old}} + \frac{n+m}{2nm} \quad (6)$$

同様に, 図 2(b) のように,  $q$  が  $(r+1)$  行  $k$  列,  $f_q$  が  $r$  行  $k$  列に位置する場合,  $r$  列と  $(r+1)$  列間の誤り近傍度  $R_r$  を次式に従い増加させる.

$$R_r^{\text{new}} = R_r^{\text{old}} + \frac{n+m}{2nm} \quad (7)$$

これらの誤り近傍度の増加量は固定値であり, ニューロン数から数値的に計算される.

**(FN-SOM10)** 全ニューロンの勝者回数を 0 回にリセットする:  $\gamma_i = 0$ .

**(FN-SOM11)** 全入力データに対して (FN-SOM1) から (FN-SOM10) を繰り返す.

### 3 シミュレーション結果

本研究では FN-SOM を様々な入力データに適用し, その結果を従来の SOM, 及び Growing Grid と

比較する.

#### 3.1 2次元入力データ

まず, 図 3(a) に示されるような, Target データ [4] と呼ばれる 2次元入力データを考える. 全入力数  $N$  は 770 個で, 4つの離れたクラスタを含む全部で 6つのクラスタがある. データはランダムに並べ替えている.

学習条件を以下に示す. 従来の SOM 及び FN-SOM は  $nm = 100$  個 ( $10 \times 10$ ) のニューロンを持ち, Growing Grid は  $2 \times 2$  個のニューロンから学習を始め,  $nm_{\text{max}} = 100$  以下である限り新しい行と列を挿入する. 学習は全入力データに対して 20 回繰り返す ( $t_{\text{max}} = 15400$ ). FN-SOM の学習では  $\lambda = 3000$  とし, 誤り近傍度の効果を確認するために, SOM と FN-SOM に対して同じ  $\alpha$  と  $\sigma$  の減少式を用いる.

従来の SOM と Growing Grid の学習結果を図 3(b) と (c) に示す. これを見ると, クラスタ間に複数の不活性ニューロンが存在することがわかる. 一方, 図 3(d) に示される FN-SOM の結果を見ると, クラスタ間にはほとんど不活性ニューロンは存在せず, 従来の SOM や Growing Grid に比べて, より効果的に入力の分布状態を反映したマップを得られていることがわかる.

FN-SOM の学習の性能を数値的にも比較するために, 以下のような 3 つの評価値を用いる.

**量子化誤差 (Quantization Error)**  $Q_e$  は, 各入力ベクトルとそれに対する勝者ニューロン間の距離の平均を計算する [1]. より 0 に近い値であるほど良い.

$$Q_e = \frac{1}{N} \sum_{j=1}^N \|\mathbf{x}_j - \bar{\mathbf{w}}_j\|, \quad (8)$$

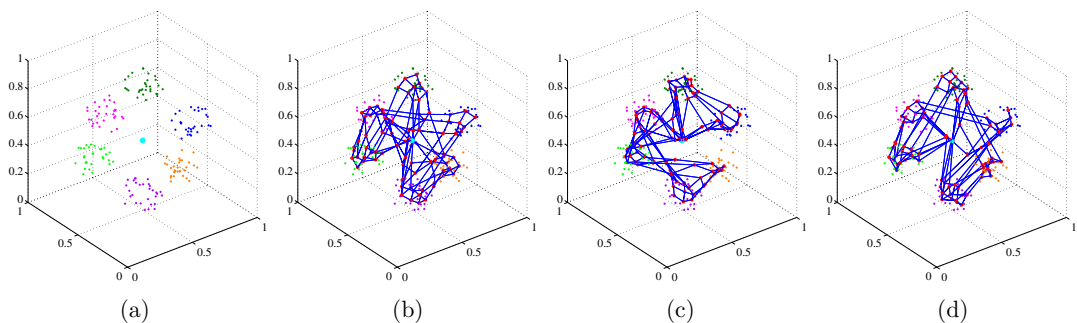


図 4: Hepta データに対する 3 手法の学習結果. (a) 入力データ. (b) 従来の SOM. (c) Growing Grid. (d) FN-SOM.

$\bar{w}_j$  は,  $\mathbf{x}_j$  に対する勝者ニューロンの重みベクトルを示す.

**トポロジーエラー (Topographic Error)  $T_e$**  は, 入力データの接続形態をどれくらい維持しているかを計算し, より 0 に近い値であるほど良い.

$$T_e = \frac{1}{N} \sum_{j=1}^N u(\mathbf{x}_j), \quad (9)$$

ここで,  $\mathbf{x}_j$  に対する勝者ニューロンと 2 番目の勝者が, マップ上の 1 近傍であれば,  $u(\mathbf{x}_j) = 1$ , それ以外なら  $u(\mathbf{x}_j) = 0$  とする.  $T_e$  は量子化誤差とは異なり, マップの構造を評価する. 不自然にねじれたマップに対しては, 量子化誤差が小さい値であったとしてもトポロジーエラーは大きくなる.

**ニューロンの利用度 (Neuron Utilization)  $U$**  は, 1 つ以上の入力ベクトルに対して勝者ニューロンとなっているニューロンの割合を計算する [3]. より 1.0 に近い値であるほど良い.

$$U = \frac{1}{nm} \sum_{i=1}^{nm} u_i, \quad (10)$$

ニューロン  $i$  が, 1 つ以上の入力に対する勝者であれば  $u_i = 1$ , そうでなければ  $u_i = 0$  とする.

計算した 3 つの評価値を表 1 に示す. FN-SOM の量子化誤差  $Q_e$  は, 3 つのアルゴリズムの中で最も小さい値であり, 従来の SOM を用いた場合よりも 7.7% 改善されている. これは, FN-SOM の結果にはほとんど不活性ニューロンがないため, より多くのニューロンが入力データを自己組織化できているからである. この結果は, ニューロンの利用度  $U$  の

表 1: Target データに対する, 量子化誤差  $Q_e$ , トポロジーエラー  $T_e$  及びニューロンの利用度  $U$ .

	SOM	Growing Grid	FN-SOM
$Q_e$	0.0207	0.0237	0.0191
$T_e$	0.0740	0.2455	0.0442
$U$	0.8100	0.8137	0.9100

値からも確認できる. FN-SOM の  $U$  の値は全アルゴリズム内で最も大きい 0.91 であり, これは 91% のニューロンが 1 つ以上の入力の勝者ニューロンとなっている, つまり, 不活性ニューロンが少ないことを示している. また, FN-SOM の  $Q_e$  と  $U$  は最も良い値であるにもかかわらず, トポロジーエラー  $T_e$  においても FN-SOM が最小値となっている. これらの結果は, FN-SOM がマップの接続形態を正確に維持しつつ, 最も効果的な自己組織化をしていることを意味している.

### 3.2 3次元入力データ

次に, 図 4(a) に示されるような, Hepta データ [4] と呼ばれる 3 次元入力データに対して学習を行う. 全データ数  $N$  は 212 であり, 密度の違う 7 つのクラスタを含んでいる. 学習は全データに対して 70 回繰り返し ( $t_{\max} = 14840$ ), 各パラメータは図 3 に対するシミュレーションと同じ値を用いる.

図 4(b), (c) 及び (d) に示された 3 手法の学習結果より, FN-SOM はクラスタ間の不活性ニューロンが最も少ないことがわかる. また, この結果に対する 3 つの評価値を表 2 に示す. これを見ると, FN-SOM

表 2: Hepta データに対する, 量子化誤差  $Q_e$ , トポロジーエラー  $T_e$  及びニューロンの利用率  $U$ .

	SOM	Growing Grid	FN-SOM
$Q_e$	0.0360	0.0409	0.0306
$T_e$	0.1698	0.1509	0.1462
$U$	0.6500	0.6863	0.8000

は全ての値において最も良い結果を得ており, 量子化誤差  $Q_e$  は従来の SOM を用いる場合より 15.0% も改善されている. それにもかかわらず, FN-SOM はトポロジーエラー  $T_e$  においても最小値を示し, 従来の SOM よりも 13.9%, Growing Grid よりも 3.1% エラーを減少できている. また, ニューロンの利用率  $U$  も最も良い値であり, 従来の SOM 及び Growing Grid を用いる場合よりも, それぞれ 23.1% と 16.6% 改善している. これらの結果より, 3次元入力データに対しても FN-SOM が最も有効であるといえる.

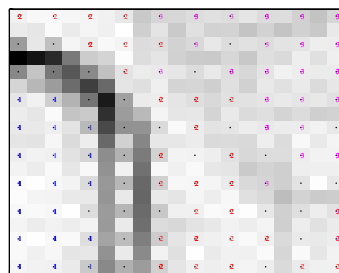
### 3.3 アヤメデータ

さらに, 実データのクラスタリング問題に対しても適用する. 今回用いる実データはアヤメデータ [7] で, パターン認識の分野で最もよく知られているデータの 1 つである. このデータは, それぞれ 50 個のデータで成り立つ 3 つのクラスを含んでおり, それぞれのクラスはアヤメの種類である Iris setosa, Iris versicolor, Iris virginica に対応する [6]. 入力 は 4 次元で, それぞれ, がくの長さ, がくの幅, 花びらの長さ, 花びらの幅である. Iris setosa は線形的に他の 2 つと区別できるが, Iris versicolor と Iris virginica はそれぞれ線形的には区別できない.

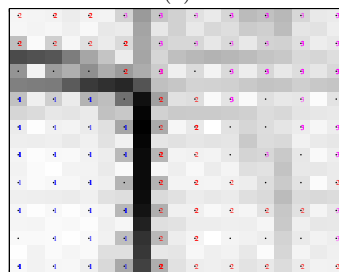
学習は全データに対して 100 回繰り返し行い ( $t_{\max} = 15000$ ), 各パラメータは図 3 に対するシミュレーションと同じ値を用いる.

図 5 はクラスタ構造を示した U-Matrix であり [4], 図 5 の各セルは近傍ニューロン間の距離を表している. この図より, 従来の SOM に比較して, FN-SOM のクラスタ境界の方がはっきりしていることが見てとれる.

量子化誤差  $Q_e$ , トポロジーエラー  $T_e$ , ニューロンの利用率  $U$  の結果を表 3 に示す. FN-SOM の  $Q_e$  及び  $T_e$  は, 3 手法の中で最小値であり,  $Q_e$  は従来の SOM よりも 13.6%,  $T_e$  は 30.5% も改善されていることが確認できる. これは, FN-SOM の結果には



(a)



(b)

図 5: アヤメデータのシミュレーション結果に対する U-Matrix. ラベル 1, 2, 3 は Iris setosa, Iris versicolor, Iris virginica にそれぞれ対応する. (a) 従来の SOM. (b) FN-SOM.

Iris setosa と他の 2 つのクラス間に不活性ニューロンがほとんどないため, より多くのニューロンが Iris versicolor と Iris virginica のデータを自己組織化できるからである. これらの結果から, FN-SOM の有効性が確認できる.

## 4 おわりに

本研究では, ニューロン間の誤り近傍度を考慮する自己組織化マップ (SOM with False-Neighbor degree between neurons: FN-SOM) を提案した. 誤り近傍度は FN-SOM のニューロンの行間と列間に割

表 3: アヤメデータに対する, 量子化誤差  $Q_e$ , トポロジーエラー  $T_e$  及びニューロンの利用率  $U$ .

	SOM	Growing Grid	FN-SOM
$Q_e$	0.0374	0.0452	0.0323
$T_e$	0.2400	0.2267	0.1667
$U$	0.7200	0.7048	0.8200

り当てられており、初期値は0であるが学習と共に徐々に増加し、重みベクトル更新の際にマップ上のノード間の距離の重みとして作用する。FN-SOMを2次元データ、3次元データ、アヤマデータに対して適用し、FN-SOMの学習における振る舞いを調査した。また、FN-SOMの結果を、3つの評価値を用いて数値的に従来のSOMとGrowing Gridに対して比較した。全ての結果でFN-SOMは最も良い値を示したことから、状況と入力データの形によって近傍関係を柔軟に変化させて学習するFN-SOMが、他の2手法と比べて、より効果的なマップを得られることを確認した。

### 謝辞

本研究は、財団法人ヤマハ音楽振興会の助成を受けて実施されたものである。

### 参考文献

- [1] T. Kohonen, *Self-organizing Maps*, Berlin, Springer, 1995.
- [2] B. Fritzke, “Growing Grid – a self-organizing network with constant neighborhood range and adaptation strength,” *Neural Processing Letters*, vol. 2, no. 5, pp. 9–13, 1995.
- [3] Y. Cheung and L. Law, “Rival-Model Penalized Self-Organizing Map,” *IEEE Trans. Neural Networks*, vol. 18, no. 1, pp. 289–295, 2007.
- [4] A. Ultsch, “Clustering with SOM: U\*C,” *Proc. of WSOM’05*, pp. 75–82, 2005.
- [5] K. Kiviluoto, “Topology Preservation in Self-Organizing Maps,” *Proc. of ICNN*, pp. 294–299, 1996.
- [6] D. J. Newman, S. Hettich, C. L. Blake and C. J. Merz, UCI Repository of Machine Learning Database, 1998.
- [7] R. A. Fisher, “The Use of Multiple Measurements in Taxonomic Problems,” *Annual Eugenics*, no.7, part II, pp. 179–188, 1936.