

密度可変型自己組織化マップによる追加学習法

島田, 敬士
九州大学システム情報科学研究院知能システム学部門

谷口, 倫一郎
九州大学システム情報科学研究院知能システム学部門

<https://hdl.handle.net/2324/5949>

出版情報 : 電子情報通信学会技術研究報告. NC, ニューロコンピューティング. 106 (407), pp.97-102, 2006-12. 電子情報通信学会

バージョン :

権利関係 :

密度可変型自己組織化マップによる追加学習法

島田 敬士[†] 谷口倫一郎[†]

[†]九州大学大学院 システム情報科学府 〒 819-0395 福岡県西区元岡 744 番地

E-mail: †{atsushi,rin}@limu.is.kyushu-u.ac.jp

あらまし 本稿では、密度可変型自己組織化マップによる追加学習法を提案する。自己組織化マップにおいて追加学習を行なう際には、1. ニューロンの枯渇問題、2. 過去に学習したデータの忘却問題、3. 位相の保存問題を解決する必要がある。山田らが提案している「シナプス硬直条件」ならびに「シナプス半硬直条件」を利用することで、過去に学習したデータの忘却を防ぐことができるが、その他の問題が生じてしまう。そこで本研究では、シナプス硬直条件とシナプス半硬直条件を用いた自己組織化マップの追加学習法を改良し、位相を保持したまま効率的にニューロンを追加する密度可変型自己組織化マップを考案した。

キーワード 自己組織化マップ, 追加学習, シナプス硬直条件, シナプス半硬直条件, ニューロンの追加, 密度可変型自己組織化マップ

Variable Density Self-Organizing Map for Incremental Learning

Atsushi SHIMADA[†] and Rin-ichiro TANIGUCHI[†]

[†] Department of Intelligent Systems, Kyushu University

744, Motoooka, Nishi-ku, Fukuoka, 819-0395 Japan

E-mail: †{atsushi,rin}@limu.is.kyushu-u.ac.jp

Abstract We propose a new incremental learning method of Self-Organizing Map. There are three problems in the incremental learning of Self-Organizing Map; 1. neuron depletion, 2. forgetting previous training data, 3. keeping topology. Weights fixed neurons and weights semi-fixed neurons are very effective for the second problem. However the other problems remains. Therefore, we improve the incremental learning method with weights fixed neurons and weights semi-fixed neurons. Our approach can increment neurons effectively in the incremental learning process.

Key words Self-Organizing Map, Incremental Learning, Weights Fixed Neuron, Weights Semi-fixed Neuron, Neuron Addition, Variable Density Self-Organizing Map

1. はじめに

Kohonen らが提案した自己組織化マップ (Self-Organizing Map:SOM) は、ニューラルネットワークのひとつで、中間層のない2層型の教師なし競合近傍学習モデルである [1], [2]。SOM は、多次元情報を可視化することができるため、多変量データ解析への応用をはじめ、近年では画像や音声の認識にも利用されるようになってきた。SOM の学習は、大きく分けて「一括学習型」と「追加学習型」に分けることができる。一般に SOM の学習と言えば、前者の一括学習型を指すことが多く、後に入力されるデータほど学習後に形成されるマップに大きな影響を与える学習方式となっている。このデータの入力順序への依存性を排除する方法として、バッチ型 SOM (Batch-Learning Self-Organizing Map) が有名である [3]。

一方で、追加学習型の学習方式をとる SOM は、逐次的にデータが入力される場合に利用される。一括学習型とは異なり、学習するデータ数が未知であるため、ニューロンの数を決定することが困難である。また、SOM は後に入力されるデータほど学習後に形成されるマップに大きな影響を与えることから、前に学習した内容を忘却する恐れがある。また、どのようなデータが入力されるか分からないため、適切な位相が保持されたマップができあがるとは限らない。このように、追加学習型の SOM には多くの問題があり、それらの解決手法がいくつか提案されている。本稿では、SOM の追加学習問題に対する先行研究を整理したうえで、追加学習を効率的に行なう新たな学習法を提案する。

2. 自己組織化マップにおける追加学習

SOM の追加学習における問題点として次の3つが挙げられる。

問題1 追加学習過程でニューロンが枯渇する

問題2 過去に学習したデータを忘却する

問題3 追加学習過程で学習するデータと既学習データ間の位相が破壊される

問題1に関して、追加学習するデータ(カテゴリ)数が増加するとそれらをマップ上に記憶するためのニューロンが学習前に設定した数では不足してしまうことが起こりうる。学習初期に必要なニューロンの数を把握することは困難であり、追加学習の過程でニューロンの数を増やしていくことが適切である。また、初めから追加学習に十分(莫大)な数のニューロンを用意しておくことも考えられるが、これでは学習データ数が少ない追加学習初期の段階では無駄なニューロンが多数存在することになり、データ間の位相的近傍性を保持したマップが形成されなくなってしまうおそれがある。

問題2に関して、SOM は後に入力されるデータほど学習後に形成されるマップに大きな影響を与えることから、追加学習前に学習した内容を忘却する恐れがある。また、追加学習するデータが既学習データと似たものであっても、既学習データを忘却しては、既学習データと追加学習データ間の位相的近傍性を適切に表現できるマップが形成されるとは限らない。

問題3に関して、SOM は似たものは近いところに記憶するという概念があるが、追加学習過程で勝利ニューロンとして選択できるニューロンが少なくなると、元のデータ空間内での位相を反映したマップが構成されづらくなってしまふ。

道端らは、SOM の学習過程で入力データの変化に伴うマップの反応から異常状態を検出し、追加学習を行なう判断基準として利用する方法を提案している[4],[5]。異常状態には、次の3つの指標が用意されている。

指標1 入力データと勝利ニューロンの結合ベクトル間の距離が異常に離れているものが多数存在する。

指標2 全く勝利しないニューロンが多数存在する。

指標3 勝利頻度が異常に多いニューロンが存在する。

これらの指標により学習過程で異常を検出し、異常が見つければマップ上のニューロンの結合ベクトルの初期化を行わずに、これまでに学習したデータを再学習するという手続きをとっている。追加学習後のマップでは、入力データに対応するニューロンが変わってしまうため、ニューロン自身は過去に学習した内容を事実上忘却していることになる。また、この方法では学習データが増加していく場合にニューロンが枯渇するという問題に関しては取り扱っていない。また、位相保存の問題については、ほぼ再学習に近い学習法をとっているため一応位相は保存されると考えられる。

島崎らは、読み取り型追加学習と入力ベクトル保存型追加学習の2種類の追加学習法を提案している[6]。読み取り型追加学習では、学習データのラベルが割り当てられたニューロンの結合係数と追加学習データを新しいマップ上で学習する方法であ

表1 SOMにおける追加学習問題への対処法の比較

	問題1	問題2	問題3
異常検出型		×	
読み取り型		×	
入力保存型		×	
シナプス硬直型	×		×

る。また、入力ベクトル保存型追加学習は、学習データのラベルが割り当てられたニューロンを参照した入力データの平均値と追加学習データを新しいマップ上で学習する方法である。いずれの方法も、新たなマップ上で学習するためのデータ選択法を提案しているのもであり、実際には再学習を行なうことと大差ない。また、ニューロンの枯渇問題に関しては特に言及されていない。

山田らは、SOM の追加学習のための「シナプス硬直条件」と「シナプス半硬直条件」を提案している[7]。シナプス硬直条件は、入力データが割り当てられたニューロンについては、追加学習時に結合係数ベクトルの更新を行わなくするものである。この作用で、過去に学習したデータの忘却を防ぐことができる。また、シナプス半硬直条件は、シナプス硬直条件が適用されたニューロンの周りのニューロンに対して、結合係数ベクトルの更新を行いつつ更新するものである。これにより、シナプス硬直条件が適用されるニューロンがマップ全体に分散ようになる。これら2つの条件を適用することで、過去の学習データの忘却を防ぎつつ、データを記憶したニューロンを全体に分散させることができるが、SOM の基本概念である「似たものは近いところに記憶する」に反したマップが構成される。また、学習データが増加したときにニューロンが枯渇するという問題に関しては取り扱われていない。

以上の3つの問題点に対する先行研究のアプローチを整理すると、表1のようになる。

問題1について、異常検出型、読み取り型、入力保存型では再学習をする際にニューロンの数を増やすことで対処可能であるが、ニューロン不足をどのように検知するかという問題や、ニューロンが不足する度にマップを作り直し再学習を行なうという方法をとる必要があるため効率的ではない。また、問題2については、異常検出型、読み取り型、入力保存型は、マップを一新する学習法であるため追加学習よりも再学習に近いと言えるため、追加学習前のニューロンは事実上過去に学習した内容を忘却している。追加学習時に新たなマップ上で学習を行なう必要がないという点では、シナプス硬直条件とシナプス半硬直条件を用いた追加学習法が適している。しかし、追加学習データに対して勝利ニューロンとして選択できるニューロンに制約があるため、学習データ間の本来の位相性を近似したマップが構成できないという問題3が生じてしまい、自己組織化の概念に反してしまう。

そこで次節では、SOM における追加学習時の3つの問題を解決するための新たな追加学習法を提案する。

3. 密度可変型自己組織化マップ

シナプス硬直条件とシナプス半硬直条件を利用した SOM の追加学習法は、他の学習法と比較して唯一過去に学習したデータの忘却問題（問題 2）に対処している。従って、シナプス硬直条件とシナプス半硬直条件を利用しつつ問題 1 と問題 3 を解決することができれば追加学習がうまくいくことになる。そこでまず、シナプス硬直条件とシナプス半硬直条件について説明する。

3.1 シナプス硬直条件・シナプス半硬直条件

シナプス硬直条件・シナプス半硬直条件を利用した学習法を以下に示す。

Step1 重みの初期値をランダムに設定する。

Step2 入力ベクトル I に対し、式 (1) を満足するニューロン c を選択し、入力ベクトルとニューロン c の結合係数ベクトルとの距離を U とする。

$$\|I - W_c\| = \min_u (\|I - W_u\|) \quad (1)$$

Step3 式 (2) により、ニューロン $u (1 \leq u \leq N)$ の結合係数ベクトルを更新する。ここで、 N はニューロンの数である。ただし、シナプス硬直条件が適用されたニューロン f の重みは更新しない。

$$W_u(t+1) = W_u(t) + H(d) \times \alpha(t) h_{cu}(t) (I(t) - W_u(t)) \quad (2)$$

ここで、 t は離散時間を表し、 $h_{cu}(t)$ は次式に示す近傍関数である。

$$h_{cu}(t) = \exp\left(-\frac{\|x_c - x_u\|^2}{2\sigma^2(t)}\right) \quad (3)$$

また、 $\alpha(t)$ 、 $\sigma(t)$ は次式に示す単調減少関数である。

$$\alpha(t) = \frac{-\alpha_0(t-T)}{T} \quad (4)$$

$$\sigma(t) = \sigma_i \left(\frac{\sigma_f}{\sigma_i}\right)^{t/T} \quad (5)$$

上式で、 T は学習回数の上限、 α_0 は学習率の初期値、 σ_i 、 σ_f はそれぞれ $\sigma(t)$ の初期値と最終値を表す。最後に、 $H(d)$ はシナプス半硬直条件のための関数で、次式で表される。

$$H(d) = \frac{1 - \exp(-d \cdot k)}{1 + \exp(-d \cdot k)} \quad (6)$$

ここで、 d はマップ上のニューロン u と、このニューロンから最も近くにあるシナプス硬直条件が適用されたニューロンとのマップ上の距離を表す。また、 k は $H(d)$ の傾きを決定する係数である。この関数は、 d が小さいほど 0 に近い値をとるため、シナプス硬直条件が適用されたニューロン f の近くに位置するニューロンの結合係数ベクトルは更新されづらくなる。また、 d が大きいニューロンでは、関数 $H(d)$ の値が 1 に近い値をとるため、従来の SOM と同じように結合係数ベクトルの更新が行われる。この条件により、シナプス硬直条件が適用されたニューロン f の近傍のニューロンは、新しい学習データを記憶しづらくなり、マップ全体に既学習ニューロンが分散して配置されることになる。

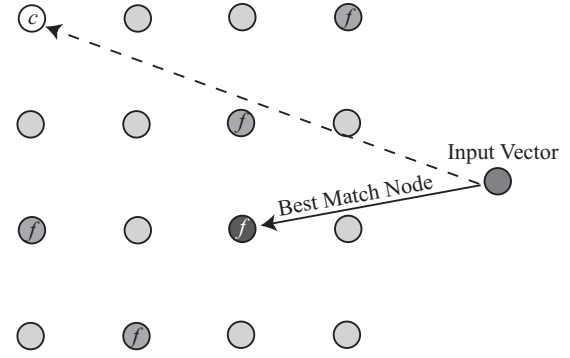


図 1 SOM の位相破壊の例

Step4 シナプス硬直条件 $U < d_f$ を満たすまで、同じ学習データに対して Step2 ~ Step3 の学習を繰り返す。

Step5 シナプス硬直条件を満たすニューロンが現れたら次の学習データを入力し、Step2 ~ Step4 の学習を繰り返す。

以上の手続きを行なうことで、シナプス硬直条件とシナプス半硬直条件を利用した SOM の追加学習が達成される。シナプス硬直条件が適用されたニューロン f はその後の学習過程で結合係数ベクトルの更新が行われなため、既学習データの忘却は起こらない。従って、問題 2 に関しては十分な対処がなされていると言える。しかしながら、学習データ数がマップ内のニューロン数に近づくると直ちにニューロンの枯渇問題（問題 1）が発生してしまう。さらに、シナプス硬直条件が適用されたニューロンとその付近のニューロンは、新たな学習データに対して最大活性化ニューロンとして選択されにくいため、本来であればシナプス硬直条件が適用されたニューロンの近くに割り当てられるべき学習データが、実際には遠くのニューロンに割り当てられることが起こり、位相関係が破壊される。この様子を、図 1 で説明する。現在の学習データに対して、式 (1) を満足するニューロンが、シナプス硬直条件が適用されたニューロン f （図中では、最も黒い色で示したニューロン）である場合であっても、そのニューロンの結合係数ベクトルは更新されることはなく、さらにニューロン f の付近のニューロンの結合係数ベクトルもシナプス半硬直条件により更新されづらいため Step2 ~ Step3 の学習を繰り返す過程で、最大活性化ニューロン c は、図に示すようにシナプス硬直条件が適用されたニューロン f から遠い位置のニューロンとなってしまふ。

以上のことから、シナプス硬直条件およびシナプス半硬直条件を利用した追加学習を行う際に、必要に応じてシナプス硬直条件が適用されたニューロン f の近くに新たにニューロンを追加することができれば、これらの問題を解消できると考えられる。

3.2 密度可変型自己組織化マップにおける追加学習法

密度可変型 SOM における追加学習では、シナプス硬直条件とシナプス半硬直条件を利用した学習法の Step2 と Step3 の間に以下の Step2 - A と Step2 - B を追加する。

Step2 - A 式 (1) を満足するニューロン c にシナプス硬直条件が適用されている場合、そのニューロンの 8 近傍に新たなニューロンを追加する。

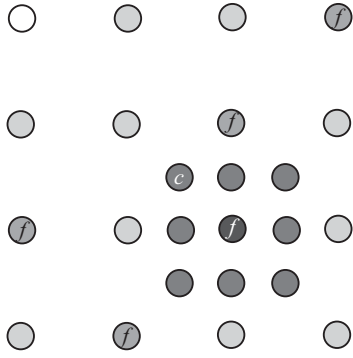


図 2 密度可変型 SOM

Step2 – B 新たに追加されたニューロン n の結合係数ベクトルの初期値を次のように決める．ニューロン n から最も近くにあるシナプス硬直条件が適用されたニューロン f がニューロン c であるとき，

$$W_n = h_{c,n} \times W_c \quad (7)$$

そうでないとき，

$$W_n = \frac{h_{c,n} \times W_c + h_{f,n} \times W_f}{h_{c,n} + h_{f,n}} \quad (8)$$

とする．

このように，ニューロンを全体ではなく局所に追加することで，図 2 のように局所的に密度の異なる SOM が形成される．学習データがどれだけ増えても適応的にニューロンが増加するので，ニューロンの枯渇問題（問題 1）は解消される．さらに，シナプス硬直条件とシナプス半硬直条件のみを利用していたときは，最大活性化ニューロンとして選択可能なニューロンは，ニューロン f から遠くに位置するニューロンであるために位相破壊が起こっていたが，このようにニューロン f の近くに最大活性化ニューロンとして選択可能なニューロンを追加し，それらのニューロンにニューロン f に似た結合係数ベクトルを初期値に持たせることで位相保存の問題（問題 3）も解消される．

4. 実験と結果

提案する密度可変型 SOM による追加学習後のマップの評価を行なうために実験を行なった．初期のマップサイズは 8×8 （ニューロンの数 64）とし， $\alpha_0 = 0.3$ ， $\sigma_i = 1.5$ ， $\sigma_f = 0.3$ ， $k = 0.6$ とした．また，ひとつの学習データに対する学習回数の上限は 1000 とした．実験では比較のため，以下の 3 つの学習法を用いた．

- 学習法 1：一括学習（追加学習なし）
 - 学習法 2：シナプス硬直条件とシナプス半硬直条件による追加学習
 - 学習法 3：密度可変型 SOM による追加学習
- 追加学習を行なう学習法 2, 3 では，ひとつの学習データに対して学習が終了してから次のデータを学習させた．

山田ら [7] と同様に，図 3 に示すアルファベット 26 文字を学習データをして用意した．学習画像のサイズは 7×7 で，それを 49 次元ベクトルとして入力した．学習法は先と同様に，学



図 3 学習に用いたアルファベット画像

習法 1，学習法 2，学習法 3 を利用し，学習後に形成されたマップを比較した．学習は，A, B, ..., Y, Z の順に行なった．

図 4 は，学習法 1 によって形成されたマップである．学習データが記憶されているニューロンの上にそのラベルを付加して表示している．「E と F」，「H と U」，「M と N」，「Q と V」についてはそれぞれの組がひとつのニューロンに記憶されていた．これはすなわち，ニューロンが不足していたことを意味する．このようなラベルの重複がなくなるまで，マップサイズを $9 \times 9 \dots$ と増やしていったときに，マップサイズが 10×10 （ニューロン数 100）のときに重複がなくなった．このときのマップの様子を図 5 に示す．

図 6 は，学習法 2 によって形成されたマップである．学習データが右下の方に偏って分布しているのがわかる．また「C」，「G」，「O」の 3 つのアルファベットに注目すると，図 5 ではこれらのアルファベットはマップ上で近い位置に記憶されているのに対して，図 6 では，マップ全体に広がって記憶されている．このことから，学習法 2 では学習データ間の位相がうまく保持できないまま追加学習が行なわれていることが分かる．

図 7 は，学習法 3 によって形成されたマップである．学習後のニューロンの数は 104 個であった．これは，学習法 1 でラベルの重複がなくなるまでマップサイズを増やしたときのニューロン数とほぼ同数であることから，追加されたニューロンの数の妥当性も示された．また「C」，「G」，「O」の 3 つのアルファベットについても，マップ上で近い位置に記憶することができることから位相保存も良好に行なえていることが分かる．最後に，学習法 3 について，学習順序をランダムにしたときに形成されたマップを図 8 に示す．この場合でも，適所にニューロンが追加されて位相保存も適切に行なえていることがわかる．以上のことから，密度可変型 SOM による追加学習は，学習の順序にもさほど影響されることなく位相を保持した追加学習が可能であることが明らかになった．

次に，追加学習に要する学習時間の比較を行なった結果を図 9 に示す．図中の赤い点は，学習法 1 による学習時間である．全ての学習データを一括して学習したいためこのように表示している．一方，追加学習を行なった場合は，各学習データを学習し終えたときまでの累積時間を表示している．また，毎回再学習を行なった場合の学習時間についても比較用に表示している．毎回再学習を行なった場合は，他の学習法に比べて大幅に学習時間がかかっている．従って学習時間を考慮すると，再



図 4 学習法 1 によるアルファベット学習後のマップ

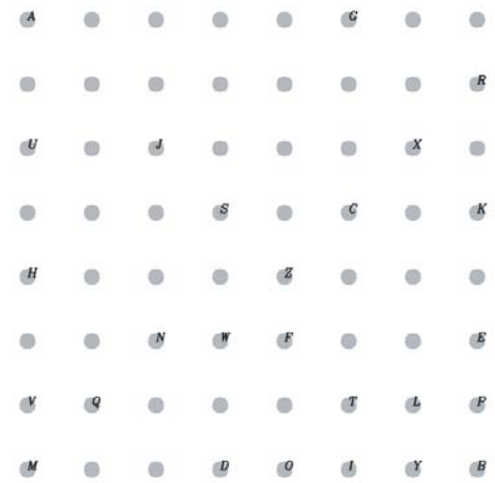


図 6 学習法 2 によるアルファベット学習後のマップ

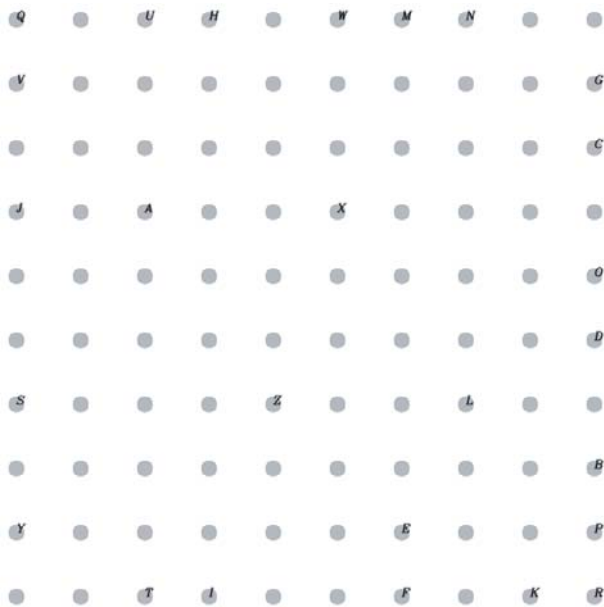


図 5 学習法 1 によるアルファベット学習後のマップ (ニューロン数 100)

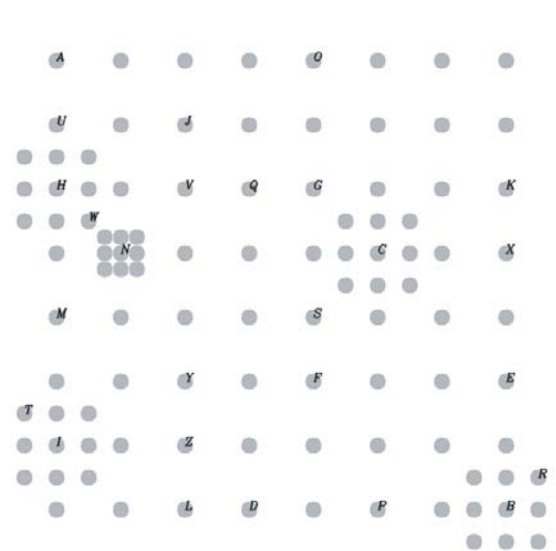


図 7 学習法 3 によるアルファベット学習後のマップ

学習することよりも追加学習の方が効率的であると言える。また、学習法 3 のほうが学習法 2 よりもニューロンの数が多いにもかかわらず、学習にかかった時間は短かった。これは、追加学習過程で追加されたニューロンに与える初期値がそのときの学習データ類似したものであるために、シナプス硬直条件が適用されるまでの繰り返し計算が少なかったためと考えられる。これは、各学習においてシナプス硬直条件が適用されるまでの繰り返し学習回数を示した図 10 から明らかで、学習法 2 では学習の後半ほど繰り返し計算回数は増加傾向にあるが、学習法 3 では、ニューロンが増加しているにもかかわらず、繰り返し計算数は学習法 2 と比べて激減していることが分かる。

最後に、各学習データがどれだけの数のニューロンに記憶されているかを評価した。図 11 は、学習データを記憶しているニューロンの分布状況を示している。分散値が大きいものほ

ど、多くのニューロンによってその学習データが記憶されていることになり、逆に分散値が 0 の場合は、ひとつのニューロンによってのみ、その学習データが記憶されていることになる。なお学習法 1 については、学習データを記憶しているニューロンに重複がなくなったときのマップにおける分布状況を解析している。学習法 1 では、全体的に平均して学習データが複数のニューロンによって記憶されているのに対して、学習法 2 および学習法 3 では、初期に学習したデータほどひとつのニューロンによって記憶されている場合が多く、その傾向は学習法 2 のほうが強かった。学習法 2 を用いる方が後に学習したデータがマップに大きく分散していることから、過去に学習したデータの忘却性が高いと言える。これに対して、学習法 3 は比較的過去に学習したデータも複数のニューロンで記憶することができている。図 10 に示したように、シナプス硬直条件が適用されるまでの繰り返し計算回数が少なかったことで、近傍学習によるニューロンの結合係数ベクトルの更新回数が少なかったため

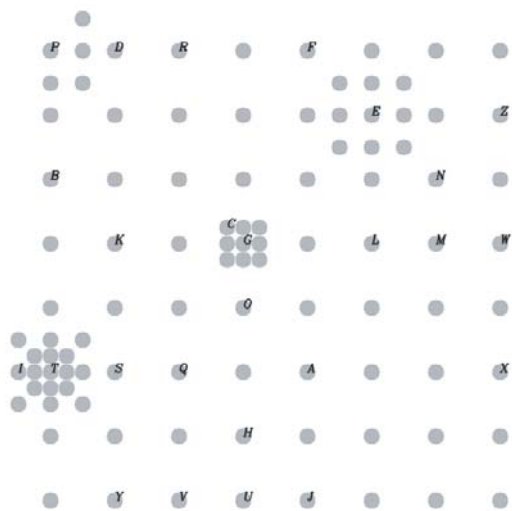


図 8 学習法 3 によるアルファベット学習後のマップ (学習順序ランダム)

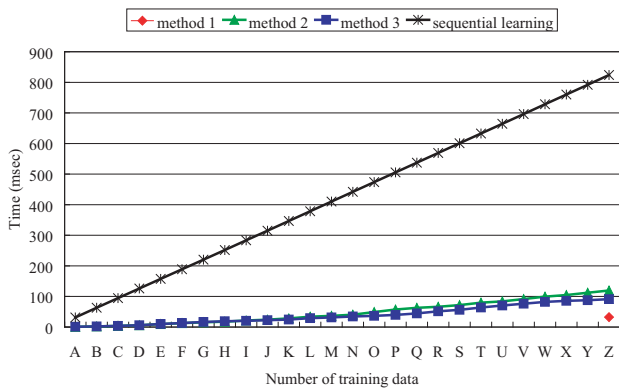


図 9 各学習法の学習時間の比較

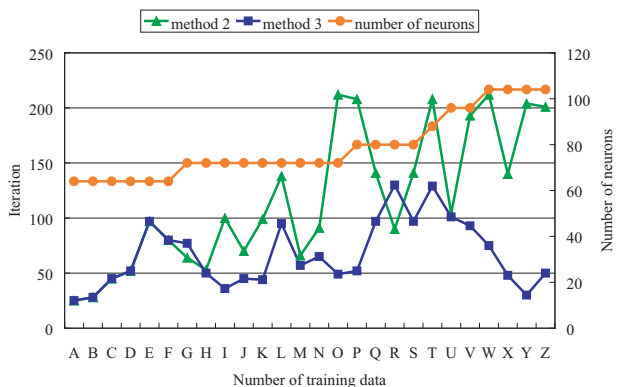


図 10 シナプス硬直条件が適用されるまでの繰り返し計算回数の比較

に忘却性が低くなったものと考察できる。

5. まとめ

密度可変型自己組織化マップによる追加学習法を提案した。従来の SOM による追加学習法では、学習データが増加するにつれてニューロンが枯渇するという問題や、過去に記憶したデータほど忘却されやすいという問題があった。さらには、忘

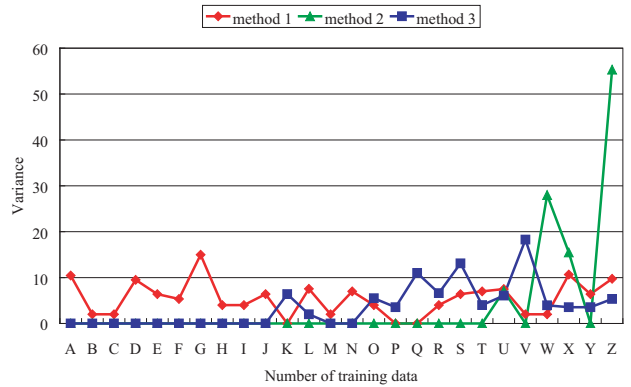


図 11 各学習データを記憶したニューロンの分布状況

却を防ぐための学習法であるシナプス硬直条件を利用した場合には、追加学習過程で学習データ空間での位相の近傍性をマップ上で反映できなくなるという問題も生じていた。提案した密度可変型自己組織化マップでは、追加学習過程でシナプス硬直条件が適用されたニューロンが最大活性化ニューロンとして選択された場合に、その近傍にニューロンを追加することで位相破壊の問題を解消すると同時に、ニューロンの枯渇問題も解消できた。さらに新たに追加したニューロンの初期値を適切に設定することで、従来法よりも学習にかかる時間を短くすることができた。

文 献

- [1] T. Kohonen. "Self-Organization and Associative Memory". Springer-Verlag, 1989.
- [2] T. Kohonen. "Self-Organizing Maps". Springer Series in Information Science, 1995.
- [3] Takashi Abe, Shigehiko Kanaya, Makoto Kinouchi, Yoshihiro Kudo, Hirotada Mori, Hideo Matsuda, Carlos Del Carpio, and Toshimichi Ikemura. "Gene Classification Method Based on Batch-Learning SOM". Genome Informatics Series, No. 10, pp. 314-315, 1999.
- [4] 道端正裕, 三好力, 増山博. "自己組織化マップの追加学習後におけるラベルに関する考察". 第 16 回ファジィシステムシンポジウム論文集 TC2-1, pp. 383-386, 2000.
- [5] 道端正裕, 三好力, 増山博. "自己組織化マップの自動追加学習システム". 日本ファジィ学会誌, Vol. 14, No. 2, pp. 147-156, 2002.
- [6] 島崎尚史, 瀬木寛人, 松浦弥三郎, 大木誠, 大北正昭. "トラス型自己組織化マップのための追加学習法の提案および検討". 第 7 回自己組織化マップ研究会, 2006.
- [7] 山田剛生, 服部元信, 森澤正之, 伊藤洋. "自己組織化特徴マップを用いた追加学習可能な連想メモリ". 電子情報通信学会 信学技法 NC98-90, pp. 93-100, 1999.