3C−4

競争学習によるニューラルネット自己組織化アルゴリズム

松山 泰男
茨城大学 工学部 情報工学科

1．はじめに

競争学習によるニューラルネット自己組織化のアルゴリズムを与える．アルゴリズムには，大別して一括更新型と逐次更新型とがあり，両者はそれぞれ有用である．また，両者の混在も許される．これらのアルゴリズムは，従来のパターンマッチングや情報圧縮の他に，自己組織化の学習過程を利用して，組合わせ最適化問題を扱うこともできる．これは，Hopfield net や，それに雑音を加えたモデル等とは異なり，コスト関数を自ら最適化していくことを利用するものである．

2．準備

以下のような記号を用いる．

$\{\mathbf{x}_i\}_{i=0}^{T-1}$ ・・・トレーニングデータ，
$\{\mathbf{v}_j\}_{j=0}^{J-1}$ ・・・グループ化されたトレーニングデータ，
Φ・・・コストを減少させるグループ化写像の集合，
$\prod_{q=0}^{Q-1}\mathcal{C}^{(q)}[old]$・・・直積型 neuro-node の集合，
Ψ_q・・・$\mathcal{C}^{(q)}[old]$ に対して，一般化された重心を更新する写像（またはその有限近似写像）の集合．

他の記号については，アルゴリズム中で説明する．

3．一括更新型アルゴリズム

写像スケジューラ

写像スケジューラは次のような規則を有しており，Step 0 〜 Step 5 における各種のパラメータ調節や選択を行う．

[グループ化写像選択規則]

Φの中から写像を選ぶ．ただし，最適化写像の集合（あるいはその有限近似写像の集合）$\tilde{\Phi}$ の各要素は無限回生起 (i.o.) が起こるように計画されている．

[neuro-node 位相の規則]

自己組織化の過程での neuro-node 間の位相的制約．

[結合 $\mathcal{L}$ の規則]

最優位 neuro-node が与える他の neuro-node への興奮性および抑制性結合の強さとその変更規則．

[更新近傍の変更規則]

最優位 neuro-node の update 近傍の変更規則．

[更新確率の変更規則]

最優位 neuro-node とその結合 neuro-node，そしてそれらの近傍 neuro-node を更新する確率とその変更規則．

[公正競争バイアスの変更規則]

最優位 neuro-node を決めるときの公平競争バイアスの変更規則．

[繰り返し回数]　$K_{\max}$

写像スケジューラにより管理される6つのステップは，次の通りである．

Step 0 (初期状態 k = 0)

次のような初期値が与えられている．

- neuro-node の集合　$\prod_{q=0}^{Q-1}\mathcal{C}^{(q)}[old]$
- グループ化パターン　$u[old]$
- neuro-node 近傍　$\mathcal{O}^{(q)}(\mathbf{c}_{n_q}^{(q)}[old],[old])$
  $n_q=0,\cdots,Nq-1,$
  $q=0,\cdots,Q-1.$
- neuro-node 更新確率　$p^{(q)}(\mathbf{b}^{(q)}[old],\mathcal{O}^{(q)}(\mathbf{c}_{n_q}^{(q)}[old],[old])),$
  $\mathbf{b}^{(q)}[old]\in\mathcal{O}^{(q)}(\mathbf{c}_{n_q}^{(q)}[old],[old]).$
- neuro-node 選択回数　$h_{n_q}[old]=0$，その集合を$\mathcal{H}$で表わす．
- 結合量　$\alpha_{\mathcal{L}}[old].$

Step 1 (グループ化)

スケジューラはΦからφを選択し，そのφを

$$\left(\prod_{q=0}^{Q-1}\mathcal{C}^{(q)}[old],u[old]\right)$$

に適用して $u[new]$ を得る．

Step 2 (停止判定)

もし前回の停止判定以後，全ての$\tilde{\Phi}$と$\tilde{\Psi}_q,(q=0,\cdots,Q-1)$，すなわち最適写像またはその近似の全てが選択された場合には，k:=k+1 とする．$k\geq K_{max}$になっていれば終了し，u と $\prod_{q=0}^{Q-1}\mathcal{C}_{n_q}^{(q)}$を得る．

Step3 (公正競争に対する最優位neuro − node)

$$d_{\mathcal{H}}(\cdot,\prod_{q=0}^{Q-1}\mathbf{c}_{n_q}^{(q)}[old])=f(d(\cdot,\prod_{q=0}^{Q-1}\mathbf{c}_{n_q}^{(q)}[old]),\mathcal{H})$$

を用いて，$\mathbf{v}_j$に対する最優位 neuro-node を求める．そして，

$$h_{n_q}^{(q)}[new]=h_{n_q}[old]+(\mathbf{c}_{n_q}^{(q)}[old]\text{の人口数})$$

を行う $(\forall n_q,\forall q)$．

Step 4 (neuro − node更新)

Step 4.1 (人口調整)

- 各 neuro-node $\mathbf{c}_{n_q}^{(q)}[old]$ について，$\mathcal{O}^{(q)}(\mathbf{c}_{n_q}^{(q)}[old],[old])$ 中の neuro-node を，確率 $p^{(q)}(\mathbf{c}_{n_q}^{(q)}[old],\mathcal{O}^{(q)}(\mathbf{c}_{n_q}^{(q)}[old],[old]))$ で $\mathbf{c}_{n_q}^{(q)}[old]$ のメンバーに加える．
- 各 neuro-node $\mathbf{c}_{n_q}^{(q)}[new]$ が興奮性結合を与える neuro-node $\mathbf{e}^{(q)}[old]$ と，その近傍$\mathcal{O}^{(q)}(\mathbf{e}^{(q)}[old],[old])$ 中の $\mathbf{b}^{(q)}[old]$ に対して，$\mathbf{c}_{n_q}^{(q)}[old]$ を $\mathbf{c}_{n_q}^{(q)}[old]$ 中の人口数だけの重複度をもって加える．その確率は，$\alpha_{\mathcal{L}}[old]p^{(q)}(\mathbf{b}^{(q)}[old],\mathcal{O}^{(q)}(\mathbf{e}^{(q)}[old],[old]))$ とする．
- 各 neuro-node $\mathbf{c}_{n_q}^{(q)}[old]$ が，抑制性結合を与える neuro-node $\mathbf{e}^{(q)}[old]$ とその近傍 $\mathcal{O}^{(q)}(\mathbf{e}^{(q)}[old],[old])$ 中の近傍 $\mathcal{O}^{(q)}(\mathbf{e}^{(q)}[old],[old])$ 中の $\mathbf{b}^{(q)}[old]$ に対して，抑制性結合の効果を増す neuro-node のメンバーを加える．それぞれの確率は

$$\alpha_{\mathcal{L}}[old]p^{(q)}(\mathbf{e}^{(q)}[old],\mathcal{O}^{(q)}(\mathbf{e}^{(q)}[old],[old])),$$

$$\alpha_{\mathcal{L}}[old]p^{(q)}(\mathbf{b}^{(q)}[old],\mathcal{O}^{(q)}(\mathbf{e}^{(q)}[old],[old]))$$

である．

Generalized neural net self-organization by competitive learning,
Yasuo Matsuyama, Dept. of Computer and Information Sciences, Ibaraki University, Hitachi 316, Japan

Step 4.2 (neuro − nodeの更新)

ψ_pをΨ_pから選び，$\psi_p(\prod_{q=0}^{Q-1}\mathcal{C}^{(q)}[old], u[new])$ を計算する ($\forall p$).

Step 5 (各種の変更規則の実行)

●$\mathcal{O}^{(q)}(\cdot,[old])$ を近傍縮小規則に従って縮小し，$\mathcal{O}^{(q)}(\cdot,[new])$ とする.
●更新確率を変更確率に従って調節する.
●結合重み$\alpha_{\mathcal{L}}[new]$ を得る.
●公正競争バイアスを変更し，$\mathcal{H}[new]$ を得る.
●$[old] \leftarrow [new]$ とする.
次いで，Step 1へ戻る.

4．逐次更新型アルゴリズム

「逐次」とは，直列処理計算を意味するものではなくて，並列処理計算になじむものであることをあらかじめ注意しておく.

写像スケジューラ

写像スケジューラは次のような規則を有しており．Step 0 ～ Step 5 における各種のパラメータ調節や選択を行なう.

[グループ化写像選択規則]

$\tilde{\Phi}$の中から写像$\tilde{\varphi}$を選び，その$\tilde{\varphi}$か恒等写像φ^0のどちらかを選択し，それをφとする．ただし$\tilde{\Phi}$の各要素は無限界生起 (i.o.) するように計画されている.

[neuro-node 位相の定義]，[結合$\mathcal{L}$の定義]，[更新近傍の変更規則]，[更新確率の変更規則]，[公正競争バイアスの変更規則]，[繰返し回数] については，一括更新型アルゴリズムと同じとする.

スケジューラは次の 6 つのブロックを管理する.

Step 0 (初期状態 k = 0)

次のような初期値が与えられている.

● neuro-node	$\prod_{q=0}^{Q-1}\mathcal{C}^{(q)}[old]$
●グループ化パターン	$u[old]$
● neuro-node 近傍	$\mathcal{O}^{(q)}(\mathbf{c}_{n_q}^{(q)}[old],[old])$
● neuro-node 更新確率	$p^{(q)}(\mathbf{b}^{(q)}[old],$ $\mathcal{O}^{(q)}(\mathbf{c}_{n_q}^{(q)}[old],[old]))$, $\mathbf{b}^{(q)}[old] \in \mathcal{O}^{(q)}(\mathbf{c}_{n_q}^{(q)}[old],[old])$.
● neuro-node 選択回数	$h_{n_q}^{(q)}=0$, その集合を$\mathcal{H}$と表す.
●結合量	$\alpha_{\mathcal{L}}[old]$.
●学習率	$\epsilon^{(q)}[old]$.

Step 1 (グループ化 k := k + 1)

グループ化写像選択規則により選ばれたφ をデータに適用し，

$$B[new] = \varphi(B[old]) = \{\mathbf{v}_j\}$$

を求める．すなわち，部分最適化により $u[new]$ を求める.

Step 2 (停止判定)

$k \geq K_{max}$になっていれば終了し，u と$\prod_{q=0}^{Q-1}\mathcal{C}^{(q)}$を得る.

Step 3 (最優位neuro − nodeの発見)

$$d_{\mathcal{H}}(\cdot,\prod_{q=0}^{Q-1}\mathbf{c}_{n_q}^{(q)}[old]) = f(d(\cdot,\prod_{q=0}^{Q-1}\mathbf{c}_{n_q}^{(q)}[old]),\mathcal{H}[old])$$

を用いて，$\mathbf{v}_j$に対する最優位 neuro-node

$$\prod_{q=0}^{Q-1}\mathbf{c}_{v_j}^{(q)}[old], \quad \forall\mathbf{v}_j \in B[new]$$

を求める．そして，

$$h_{n_q}^{(q)}[new] = h_{n_q}^{(q)}[old]+1$$

を行なう.

Step 4 (neuro − node更新)

Step 4.1 (最優位neuro − nodeとその近傍neuro − nodeの更新)

$$\begin{cases} \mathbf{c}_{v_j}^{(q)}[new] & = \mathbf{c}_{v_j}^{(q)}[old] + \epsilon^{(q)}[old]\mathbf{r}^{(q)}(\mathbf{v}_j,\mathbf{c}_{v_j}^{(q)}[old]), \\ & w.p.\ p^{(q)}(\mathbf{c}_{v_j}^{(q)}[old],\mathcal{O}^{(q)}(\mathbf{c}_{v_j}^{(q)}[old],[old])), \\ & \forall q, \forall\mathbf{v}_j \in B[new]. \end{cases}$$

$$\begin{cases} \mathbf{b}^{(q)}[new] & = \mathbf{b}_{v_j}^{(q)}[old] + \epsilon^{(q)}[old]\mathbf{r}^{(q)}(\mathbf{v}_j,\mathbf{b}^{(q)}[old]), \\ & w.p.\ p^{(q)}(\mathbf{b}^{(q)}[old],\mathcal{O}^{(q)}(\mathbf{c}_{v_j}^{(q)}[old],[old])), \\ & \forall\mathbf{b}^{(q)} \in \mathcal{O}(\mathbf{c}_{v_j}^{(q)}[old],[old]), \\ & \forall q, \forall\mathbf{v}_j \in B[new]. \end{cases}$$

を実行する.

Step 4.2 (興奮性および抑制性結合に関連した更新)

$\mathbf{e}^{(q)}[old]$ は$\mathbf{c}_{v_j}^{(q)}[new]$ との興奮性/抑制性結合が決められている neuro-node とする．このとき，次のように更新する.

$$\mathbf{b}^{(q)}[new] = \mathbf{b}^{(q)}[old] \pm \epsilon^{(q)}[old]\mathbf{r}^{(q)}(\mathbf{v}_j,\mathbf{b}^{(q)}[old]),$$

$$w.p.\ \alpha_{\mathcal{L}}p^{(q)}(\mathbf{b}^{(q)}[old],\mathcal{O}^{(q)}(\mathbf{e}^{(q)}[old],[old])),$$

$$\forall\mathbf{b}^{(q)}[old] \in \mathcal{O}^{(q)}(\mathbf{e}^{(q)}[old],[old]), \forall\mathbf{v}_j \in B[new], \forall q.$$

ただし，$+\epsilon^{(q)}[old]$ は興奮性結合，$-\epsilon^{(q)}[old]$ は抑制性結合を意味する.

Step 5 (各種の変更規則の実行)

学習率を更新する他は一括更新型と同じで， Step 1 に戻る.

5．収束性

収束性については次の事がいえる.
〔定理〕一括更新型アルゴリズムも逐次更新型アルゴリズムも $K_{max} \to \infty$ のとき収束する.
〔注〕 一括更新型については，条件を厳しくすると K_{max}が有限で収束する.

6．修正ベクトルの例

巡回セールスマン問題や画像のような 2 次形式については，

$$\mathbf{r}^{(q)}(\mathbf{v}_j,\mathbf{c}_{v_j}^{(q)}[old]) = \bar{\mathbf{w}}^{(q)}(\mathbf{v}_j) - \mathbf{c}_{v_j}^{(q)}[old],$$

音声については，

$$\mathbf{r}_i^{(q)}(\mathbf{v}_j,\mathbf{c}_{v_j}^{(q)}[\mathbf{old}]) = \mathbf{2}(\mathbf{k}_{v_j}^{(i)} - \mathbf{k}(\mathbf{i}))/\mathbf{f}(\mathbf{k}_{v_j}^{(i)},\mathbf{k}(\mathbf{i})),\ \forall\mathbf{i}.$$

となる．ここに， $f(\cdot,\cdot)$ は，線形予測理論により同出される重み量である.

7．考察

LGB ベクトル量子化は，一括更新型による $\varphi=\varphi^0$で$Q=1$ かつ $\mathcal{O}$，$\mathcal{L}$，$\alpha_{\mathcal{L}}$，$\mathcal{H}$のない場合である．Kohonen らによるベクトル量子化は，$\varphi \equiv \varphi^{\circ}$，Q=1，2 乗ノルムコストであり，$\mathcal{L}$，$\alpha_{\mathcal{L}}$，$\mathcal{H}$のない場合である.
可変領域ベクトル量子化[1]は一括更新型で，$\mathcal{O}$，$\mathcal{L}$，$\alpha_{\mathcal{L}}$，$\mathcal{H}$のない場合である.

適用例については，本大会の別稿として与えてある.

文献

[1] 松山，可変領域ベクトル量子化，信学論，Vol. J70-A, p.1830 (1987).
[2] Matsuyama, Y., Multiple descent cost algorithms, IJCNN-WASH-90 (1990).

[3] Kohonen, T., Self-Organization and Associative Memory, Springer (1984 and 1988).