

EBP学習アルゴリズムの逐次修正法と一括処理法

山本祥弘・佐藤 武^{*1}・松田直樹^{*2}

知能情報工学科・^{*1}富士機械製造株式会社・^{*2}鳥取大学大学院

Sequential and Batch Processing Methods of the EBP Learning Algorithm

Yoshihiro YAMAMOTO, Takeshi SATO^{*1} and Naoki MATSUDA^{*2}

Department of Information and Knowledge Engineering, Faculty of Engineering

Tottori University, Tottori, 680 Japan

E-mail yamamoto@ike.tottori-u.ac.jp

^{*1} Fuji Machine Production Corporation, Ltd, Aichi, 472 Japan

^{*2} Graduate School of Engineering, Tottori University, 680 Japan

Abstract: A new error back propagation learning algorithm called EBP is examined for the cases of a sequential and a batch processing methods using a XOR problem as an example. Many methods can be considered for the batch processing method. Among these, one method is most effective with 100% convergent rate, which treats a neural network as a combination of two layered of each units.

Key words: Error back propagation, Learning algorithm, Multilayer neural network, Sequential method, Batch processing method

1. はじめに

1986年、階層型ニューラルネットワーク(NN)に対する学習アルゴリズム(誤差逆伝搬法:BP法)がRumelhartら[1]により発表されていらい、NNとその応用に関する研究が非常に活発に行われている。しかし、BP法には多くの欠陥が指摘されており、その改良に多くの努力が払われているが、勾配法によるBP法を基にした改良では、限界があると思われる。そこで、この勾配法から脱却した新しい学習アルゴリズム[2]~[4]を著者の一人が開発した。これは最小二乗法による解であり、この流れは、適応制御の歴史とまさに一致する。すなわち、適応制御においても初期の頃には勾配法がその適応アルゴリズムに用いられていたが、勾配法ゆえに派生する局所性の問題から脱皮する一連の研究を経て、現在では、最小二乗法によるアルゴリズムが知られている[5]。問題は、どのように最小二乗法を適用するかである。

提案する方法は次の二つから構成される。

- 各中間層への仮の教師信号を、NNの出力誤差から順に決定する(誤差逆伝搬法EBP)。
- 与えられた仮の教師信号との誤差を零とするように重みパラメータを修正する(最小2乗

法)。

このアルゴリズムは一般論としては完成されておらず、本論では、その具体的方法のいくつかを検討する。XOR問題を例題とするシミュレーションの結果から、各ユニットに対する2層NN問題として解く方法が最も優れていると判断される。この結果は現時点での結論であり、B)の最小2乗法として、推定法で知られている指数重み付き最小2乗法がさらに有効ではないかと現在検討中である。

2. ニューラルネットワーク

本論では、以下の3層NNを考える。

$$c_p = f(z_p), \quad z_p = w_{p-1}^T a_p, \quad (1-1)$$

$$a_p = f(y_p), \quad y_p = V_{p-1}^T b_p \quad (1-2)$$

ここに c , b はNNの出力と入力であり、簡単のためスカラ出力とする。また、 w , a , y , b はベクトル、 V は行列であり、そのサイズは適切なものとする。添え字 p は学習データの番号であるが、動的システムを対象のときは時刻と考えてよい。関数 $f(\cdot)$ は以下で定義する非線形関数である

が、これが線形であり、かつ2層回路とすれば、(1-1)式は通常のパラメータ推定の式と同じである。(1)式のNNにおいて w と V が可調整パラメータである。この調整は、 d_p を教師信号(目標信号)として、

$$E = \frac{1}{2} (d_p - c_p)^2 \quad (2)$$

を最小とすることを目的としている。すでに良く知られている誤差逆伝搬法(BP法)は

$$\Delta w_{p-1} = - \frac{\partial E}{\partial w_{p-1}} \quad (3-1)$$

$$\Delta V_{p-1} = - \frac{\partial E}{\partial V_{p-1}} \quad (3-2)$$

であり、(3-2)式右辺の計算に、出力側からの逆伝搬が実行されている。この(3)式は勾配法そのものであり、収束が遅い。

3. 学習アルゴリズム

3.1 逐次修正法

(1)式において w, V の修正量をそれぞれ $\Delta w, \Delta V$ とする。このとき(1-2)式から、ベクトル a に Δa の修正がなされ、(1-1)式より

$$c_p + \Delta c_p = f(z_p + \Delta z_p), \quad (4-1)$$

$$\begin{aligned} z_p + \Delta z_p \\ = (w_{p-1} + \Delta w_{p-1})^T (a_p + \Delta a_p) \end{aligned} \quad (4-2)$$

となる。すなわち、 z 、従って c の値は w と a の修正を受けることになる。逆に、出力(4-1)式の値が(2)式を最小にするため、すなわち d_p と一致するために必要な修正量 Δc_p から、(4-2)式を満たす Δw_{p-1} と Δa_p を求めようとするのが、提案するアルゴリズムの特徴の一つである。また本論では、(2)式を減少させるのではなく、その値をゼロとするように決めるのが第二の特徴である。そのためには、

$$c_p + \Delta c_p = d_p \quad (5)$$

を満たすことであるが、これはまた、

$$z_p + \Delta z_p = f^{-1}(d_p) \quad (6)$$

と等価である。ただし、関数 f の逆関数の存在を仮定している。以上より、

$$(w_{p-1} + \Delta w_{p-1})^T (a_p + \Delta a_p) = f^{-1}(d_p) \quad (7)$$

が条件式となる。ここで、 Δw は任意であるが、 Δa には制限が付く。すなわち、新しい a_{Np} 、

$$a_{Np} = a_p + \Delta a_p \quad (8)$$

は次の層(1-2)式にたいする教師信号の役割をすることになるので、 a_{Np} は非線形関数 $f(\cdot)$ の値域に属することが必要となる。それでも(7)式を満たす解は種々考えられる[3]が、そのうちの一つが次のアルゴリズムである。

[逐次学習アルゴリズム[4]]

$$1) \Delta a_p = w_{p-1} (w_{p-1}^T w_{p-1})^{-1} \{ f^{-1}(d_p) - w_{p-1}^T a_p \} \quad (9)$$

$$2) a_{Np}' = a_p + \Delta a_p \quad (10)$$

$$3) a_{Np} = H * a_{Np}' \quad (11)$$

$$4) \Delta V_{p-1} = b_p (b_p^T b_p)^{-1} \{ f^{-1}(a_{Np}) - V_{p-1}^T b_p \} \quad (12)$$

$$5) V_p = V_{p-1} + \Delta V_{p-1} \quad (13)$$

$$6) \Delta w_{p-1} = a_{Np} (a_{Np}^T a_{Np})^{-1} \{ f^{-1}(d_p) - w_{p-1}^T a_p \} \quad (14)$$

$$7) w_p = w_{p-1} + \Delta w_{p-1} \quad (15)$$

このアルゴリズムにおいて(12),(14)式は現時点のみのデータに基づく場合であり、直交射影法と呼んでいる。すなわち(14)式は、超平面

$$f^{-1}(d_p) = w^T a_p \quad (16)$$

への直交射影となっている。(12)式も同様である。また、蓄積された過去のデータをも利用する方法として、正規直交化法[4]を提案している。一方、過去のデータを一括して処理する方法は次節で示す。ところで、(11)式の意味は、

$$a_{Np} \in \text{range}(f(y)) \quad (17)$$

とすることである。 $f(y)$ の値域は通常区間で与えられるので、その方法として、

H1) 値域を越えた分をカットする方法。

H2) $f(0)$ を中心にして a_{Np} を縮小する方法。

の二つが考えられる。また、非線形関数 f として、

$$F1) f(y) = \frac{1}{1 + \exp(-my)} \quad (18)$$

$$F2) g(y) = \frac{1 - \exp(-my)}{1 + \exp(-my)} \quad (19)$$

を考える。ただし通常は $m = 1$ とする。(18)式はシグモイド関数と呼ばれており、その値域は(0,1)開区間である。一方(19)式は双曲線関数の一つで $\tanh(my/2)$ であり、その値域は(-1, 1)の開区間である。これより組み合わせとして4種類考えられる。ここに、 $g(y) = 2f(y) - 1$ なる関係がある。文献[4]では H1), F1) の場合を扱っている。

3.2 一括処理法

(1)式は p 番目の学習データにたいする応答を表しているが、本節では、このデータを M 組一括して処理することを考える。そのために以下の記号を導入する。

$$c_p = (c_{p1}, c_{p2}, \dots, c_{p(M+1)}) \quad (20-1)$$

$$z_p = (z_{p1}, z_{p2}, \dots, z_{p(M+1)}) \quad (20-2)$$

$$A_p = (a_{p1}, a_{p2}, \dots, a_{p(M+1)}) \quad (20-3)$$

$$Y_p = (y_{p1}, y_{p2}, \dots, y_{p(M+1)}) \quad (20-4)$$

$$B_p = (b_{p1}, b_{p2}, \dots, b_{p(M+1)}) \quad (20-5)$$

$$d_p = (d_{p1}, d_{p2}, \dots, d_{p(M+1)}) \quad (20-6)$$

このとき、(1)式に対応して、

$$c_p = f(z_p), \quad z_p = w_{p-1}^T A_p, \quad (21-1)$$

$$A_p = f(Y_p), \quad Y_p = V_{p-1}^T B_p \quad (21-2)$$

と表され、このときの学習の目的は

$$d_p = c_p \quad (22)$$

あるいは等価な

$$f^{-1}(d_p) = z_p \quad (23)$$

である。このとき逐次学習アルゴリズムと同様にして、以下の結果が導かれる。

[一括学習アルゴリズム]

$$1) \Delta A_p = w_{p-1} (w_{p-1}^T w_{p-1})^{-1} \{ f^{-1}(d_p) - w_{p-1}^T A_p \} \quad (24)$$

$$2) A_{N_p}' = A_p + \Delta A_p \quad (25)$$

$$3) A_{N_p} = H * A_{N_p}' \quad (26)$$

$$4) \Delta V_{p-1} = B_p (B_p^T B_p)^{-1} \{ f^{-1}(A_{N_p})$$

$$- V_{p-1}^T B_p \}^T \quad (27)$$

$$5) V_p = V_{p-1} + \Delta V_{p-1} \quad (28)$$

$$6) \Delta w_{p-1} = A_{N_p} (A_{N_p}^T A_{N_p})^{-1} \{ f^{-1}(d_p) - w_{p-1}^T A_p \}^T \quad (29)$$

$$7) w_p = w_{p-1} + \Delta w_{p-1} \quad (30)$$

(17)式以下の注意がこの場合にも適用される。ただしこの場合には、行列 B_p のフルランクを仮定している。従って、 $B_p^T B_p$ の逆行列が存在しない場合には、(27)式を次式で置き換えるものとする。

$$4') \Delta V_{p-1} = (B_p B_p^T)^{-1} B_p \{ f^{-1}(A_{N_p}) - V_{p-1}^T B_p \}^T \quad (27')$$

同様に(29)式に対しても

$$6') \Delta w_{p-1} = (A_{N_p}^T A_{N_p})^{-1} A_{N_p} \{ f^{-1}(d_p) - w_{p-1}^T A_p \}^T \quad (29')$$

の置き換えを行うが、問題は行列 A_p のフルランクを仮定できないことである。そのために、このフルランク性を満たすように、学習データの選別が必要となる。一括学習で $M = 1$ とすれば逐次学習となることは当然である。また、2層NNで線形ユニットの場合には、この一括学習アルゴリズムは著者の一人が提案している T L S M 適応アルゴリズム[6]と一致している。

4. 数値例による検討

提案するアルゴリズムのなかで $M = 1$ の逐次修正が最も収束が遅いのは当然であり、正規直交化を用いると大きく改善されることが文献[4]で示されている。そこで以下では、一括処理の具体的方法を例題を用いて検討する。

4.1 2層NN

2層NNは(1-1)式のみで表され、 a を入力とした場合である。ここに f が線形であり、 $c = z$ とすれば、推定理論で用いられる回帰モデルと一致し、このときの学習アルゴリズムは文献[6]の適応アルゴリズムと一致している。2層NNで解ける問題は線形分離可能な場合であり、AND 回路、OR 回路を解いてみた。重みの初期値は ± 1 に限定した6通りとし、その結果を以下に示す。

Table 1 AND circuit

	逐次		全入力		M=N+1		M可変	
	F1	F2	F1	F2	F1	F2	F1	F2
C	6	6	6	6	6	6	6	6
A	40.7	10.8	4	5	2	2	2	2
L	38	10	4	5	2	2	2	2
U	42	12	4	5	2	2	2	2

Table 2 OR circuit

	逐次		全入力		M=N+1		M可変	
	F1	F2	F1	F2	F1	F2	F1	F2
C	6	6	6	6	6	6	6	6
A	16.3	11.5	3.8	5	1	1	2	2
L	15	10	3	5	1	1	2	2
U	19	13	4	5	1	1	2	2

C: 収束数、A: 平均収束回数、

L: 最小収束回数、U: 最大収束回数

これらの回路はどちらも4組の入力データがあり、これらをすべて入力(M=4)としたのが「全入力」であり、1組ずつ(M=1)入れて学習させるのが「逐次」である。これに対して、入力変数をNとし、しきい値を加えて「M=N+1」とする方法が考えられる。この例題ではN=2, M=3である。これは文献[4]の正規直交化法と類似しており、毎回改めて入力を入れ直す点が異なっている。一方、全データを対象とするが、出力誤差が0とならないデータの組のみを入力するのが「M可変」である。

ところで出力誤差は

$$e_p = f^{-1}(d_p) - \mathbf{w}_{p-1}^T \mathbf{a}_p \quad (31)$$

で考えても等価である。このとき、例えば $d_p = 1$ のとき逆関数は無限大となるので、 $d_p = 0.9$ で代用している。従って、

$$f^{-1}(d_p) e_p \leq 0 \rightarrow e_p = 0 \quad (32)$$

なる操作を加えている。

このように一括処理にも種々の方法が考えられるが、M=1の逐次修正が最も悪いのは当然である。それでも従来のBP法よりは大きな改善となっている。上記表の結果からは「M=N+1」が最も良く、「M可変」も同程度であるといえる。しかし、これらの結果は2層NNに対してであり、一般の多層NNに対してはどうかを次節で検討する。

4.2 3層NN

2層NNでは解けない問題として、排他的論理和(XOR問題)が知られている。この問題を以下の2入力1出力3層NNで解いてみる。

$$c = f(z), \quad z = \sum_{j=0}^2 w_j a_j \quad (31-1)$$

$$a_j = f(y_j), \quad y_j = \sum_{k=0}^2 v_{jk} b_k \quad (31-2)$$

ここに添え字pは省略する。またj, k=0はしきい値を表し、 $a_0 = b_0 = 1$ とする。このようにしきい値が入ると先のアルゴリズムに若干の修正が必要となる。すなわち、(31)式をベクトル行列表現すると

$$c = f(z), \quad z = \bar{\mathbf{w}}^T \bar{\mathbf{a}} + w_0 a_0 = \mathbf{w}^T \mathbf{a}, \quad (32-1)$$

$$\bar{\mathbf{a}} = \mathbf{f}(\mathbf{y}), \quad \mathbf{y} = \mathbf{V}^T \mathbf{b} + v_0 b_0 = \mathbf{V}^T \mathbf{b} \quad (32-2)$$

となり、例えば(9)式は

$$1) \triangle \bar{\mathbf{a}}_p = \bar{\mathbf{w}}_{p-1} (\bar{\mathbf{w}}_{p-1}^T \bar{\mathbf{w}}_{p-1})^{-1} \{ f^{-1}(d_p) - \bar{\mathbf{w}}_{p-1}^T \mathbf{a}_p \} \quad (9')$$

と修正する。

【初期値の設定】：従来のBP法では初期値の設定に問題があり、通常乱数を用いている。本論では、任意の初期値からの収束を目指して、 w_j 、 v_{jk} の9個の初期値を±1に限定する。すると場合の数は 2^9 通りとなるが、NNの対称性を考慮した176通りを考える。

【傾きmの変動】：176通りの初期値の中で、24通りはNNが対称形となって基本的には収束不可能である。この対称形を崩すために(18)あるいは(19)式のmの値をフラフラ変動させることを $f(y_2)$ に対して行った。

【周期】：3層NNに対して、2層NNと全く同じ方法を行っても上手くいかない。その理由として、仮の教師信号が文字通り仮のものだからである。そのため、 \mathbf{w} と \mathbf{V} はそれぞれ解に近づこうとしても、そのタイミングがずれることが考えられる。例えば、ある時点で \mathbf{w} が解となっても、 \mathbf{V} が解でないとき、次の時点で \mathbf{w} が解から外れてしまうことが考えられる。この問題を解決する方法として、 \mathbf{V} の値を固定して、 \mathbf{w} のみを学習

する期間を設ける。すると、 w と V の学習および w のみの学習が周期的に繰り返される。これを周期と呼ぶことにする。

以上の準備のもとにシミュレーションを行った結果の一部を Table 3～6 に記す。この中では、Table 4 の「 $M=N+1$ 」の方法が最もよく、 m 変動の場合の収束率は 100% を含み、すべて 90% 以上である。つぎに Table 6 の M 可変 Π と続くが、これらの方法はすべて周期の設定が面倒である。そこで 5 番目の方法として、出力誤差が 0 とならないデータの組のみを入力する「 M 可変」の考え方を各ユニットそれぞれに対して行う方法を提案する。例えば XOR 問題では Fig. 1 の点線で示すように 3 入力 1 出力の 2 層 NN が 3 組存在する。これらの各ユニットには仮の教師信号が与えられているので、この仮の教師信号との誤差が 0 とならないデータの組のみを一括処理する方法であり、その結果を Table 7 に示す。

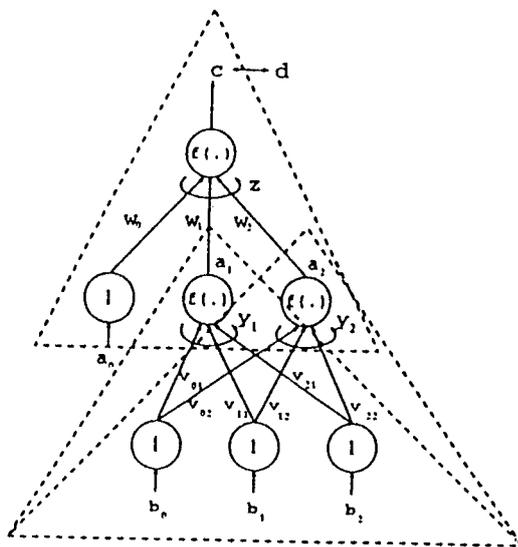


Fig. 1 NN of the Proposed Method

この結果は、Table 4 の「 $M=N+1$ 」の方法よりも収束率を上げることができ、また周期を必要としないことから、本論文で検討した方法の中では最良であると言える。また、XOR のようなテスト問題でなく、一般の大規模な問題を考えたとき、周期の決定はますます複雑になると考えられるが、提案する方法は任意の問題に適用可能である。

Table 7 XOR-problem: Proposed Method

F	F 1: シグモイド				F 2: 双曲線			
	固定		変動		固定		変動	
H	H1	H2	H1	H2	H1	H2	H1	H2
C	87	69	176	170	110	101	176	175
%	49	39	100	97	63	57	100	99
A	9.9	4.2	3.6	9.4	3.2	7.5	5.0	7.0
L	1	1	1	1	1	1	1	1
U	20	39	25	48	22	44	42	39

C: 収束数、A: 平均収束回数、

L: 最小収束回数、U: 最大収束回数

5 おわりに

本論文では、多層ニューラルネットワークの学習アルゴリズムとして新しく開発している EBP 学習アルゴリズムの具体的な計算法として、逐次修正法と一括処理法を検討した。主たる目的は初期値にたいする収束率の向上であり、100% の収束率に成功した。一括処理法には種々の方法が考えられるが、NN を各ユニットの 2 層 NN の結合とする方法が、収束率およびアルゴリズムの容易さからも最も優れていると考えられる。

参考文献

- 1) D.E.Rumelhart and J.L.McClelland Eds.: Parallel Distributed Processing, Vol.1, MIT Press, 1986.
- 2) 山本: ニューロ回路の学習規則と適応アルゴリズム、システム制御情報学会論文誌、7 巻、12 号、pp47-49, 1994.
- 3) 山本: 多層ニューロ回路の新しい誤差逆伝搬法とその代数的性質、システム制御情報学会論文誌、9 巻、5 号、pp1-9, 1996.
- 4) 山本: 正規直交化法による多層ニューラルネットワークの学習アルゴリズム、システム制御情報学会論文誌、9 巻、10 号、pp33-39, 1996.
- 5) K.J.Astrom and B.Wittenmark: Adaptive Control, Addison-Wesley, 1989.
- 6) 山本: 修正最小 2 乗法による適応アルゴリズム、計測自動制御学会論文集、26 巻、12 号、pp22-27, 1990.
- 7) 山本、佐藤: EBP 学習アルゴリズムの逐次修正法と一括処理法、第 17 回適応制御シンポジウム資料、pp37-40, 1997.

(受理 平成 9 年 8 月 30 日)

Table 3 XOR-problem : 全入力、M = 4

F	周期4								周期5							
	F1 : シグモイド				F2 : 双曲線				F1 : シグモイド				F2 : 双曲線			
	固定		変動		固定		変動		固定		変動		固定		変動	
H	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
C	69	86	77	85	97	108	115	94	69	82	78	76	97	108	109	96
%	39	49	44	48	55	61	65	53	39	47	44	43	55	61	62	55
A	3.4	8.2	3.1	4.5	2.3	9.4	2.1	3.9	4.0	7.9	3.7	2.7	2.5	11.4	2.0	8.5
L	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
U	13	39	9	46	5	25	5	17	16	46	12	11	6	31	6	41

Table 4 XOR-problem : M = N + 1 = 3

F	周期4								周期5							
	F1 : シグモイド				F2 : 双曲線				F1 : シグモイド				F2 : 双曲線			
	固定		変動		固定		変動		固定		変動		固定		変動	
H	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
C	42	97	176	174	80	82	176	169	106	151	174	161	128	140	174	165
%	24	55	100	99	45	47	100	96	60	86	99	91	73	80	99	94
A	3.3	5.2	4.6	5.9	3.1	6.9	5.6	7.1	6.7	6.0	4.6	5.0	5.4	7.8	4.8	6.7
L	3	3	2	2	3	3	2	2	3	3	2	2	3	3	2	2
U	7	27	30	39	4	35	23	23	34	24	32	32	23	13	17	24

Table 5 XOR-problem : M可変 I (全→可変)

F	周期4								周期5							
	F1 : シグモイド				F2 : 双曲線				F1 : シグモイド				F2 : 双曲線			
	固定		変動		固定		変動		固定		変動		固定		変動	
H	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
C	69	93	78	94	97	108	115	134	69	107	76	94	97	108	114	127
%	39	53	44	53	55	61	65	76	39	61	43	53	55	61	65	72
A	3.3	8.2	3.2	5.7	2.0	5.3	2.0	6.8	4.4	9.7	3.4	8.0	2.2	6.3	3.2	9.2
L	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
U	13	41	14	34	5	14	5	26	32	46	6	49	6	17	41	44

Table 6 XOR-problem : M可変 II (N + 1 → 可変)

F	周期4								周期5							
	F1 : シグモイド				F2 : 双曲線				F1 : シグモイド				F2 : 双曲線			
	固定		変動		固定		変動		固定		変動		固定		変動	
H	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2	H1	H2
C	41	75	164	127	84	56	154	127	106	151	174	169	127	140	170	176
%	23	43	93	72	48	32	88	71	60	86	99	96	72	80	97	100
A	3.1	2.7	2.7	3.5	2.3	2.0	3.1	5.8	6.8	5.8	3.1	5.1	4.7	7.2	3.4	7.0
L	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
U	36	36	39	45	7	2	14	44	32	22	38	38	23	12	19	33

C : 収束数、 A : 平均収束回数、 L : 最小収束回数、 U : 最大収束回数