

バイナリニューラルネットの効果的な学習アルゴリズムについて

島田 政範 齋藤 利通
法政大学工学部電気電子工学科

Binary Neural Networks の新しい幾何学的学習アルゴリズムを提案する。BNN は入出力がバイナリ信号のフィードフォワードニューラルネットである。幾何学的学習アルゴリズムは超平面による、分離を用いており、簡素であるが、非常に学習速度が早く、また、ハード化に適している教師あり学習アルゴリズムである。本学習アルゴリズムでは、実際には入力されない信号を仮想入力として用いる。これによって、従来は困難であった中間層が少なく、結合パラメータのばらつきが少ないネットワークが構成できる。

1. はじめに

近年、人間の脳の優れた情報処理機能に学んだニューラルネットの研究が盛んに行われている。ニューラルネットの結合系は多く提案されているが、最も多くの応用が提案されているのが階層構造の multi-layer perceptron (MLP) であり、その学習アルゴリズムとして backpropagation (BP) アルゴリズムが非常に有効であることが数多くの文献により示されている。この BP アルゴリズムは最急降下法を用いており、高い汎化能力をもつネットワークを構成できるアルゴリズムである。

一方、最近のデジタル技術の発展に伴い、デジタル値の入出力をもつ MLP が検討されてきた。しかし、BP アルゴリズムはシグモイド関数を用いたニューロンを基に設計されたアルゴリズムであるため、単純なバイナリ信号を扱う場合にも非常に計算時間がかかるなど、能率が悪い点がある。

そこでバイナリ信号を入出力とする Binary Neural Networks (BNN) の学習アルゴリズムがいくつか提案されている [1]-[8]。その中に入力信号を N 次元超立方体の中で考え、幾何学的に超平面を求める Expand-and-Truncate Learning Algorithm (ETL) [5] がある。この ETL アルゴリズムでは、BP アルゴリズムで必要な繰り返し計算がないので、計算時間が極端に短い。最急降下法を用いていないので、ローカルミニマムに陥らず必ず学習が終了する。また、ニューロン間の結合パラメータ(重み)、しきい値が整数で求めらハード化に適しているなどの利点がある。しかし、連続空間で超平面を求められないことにより中間層の数が増加する。また、複雑な教師信号を学習して、ネットワークが大きくなる場合、中間層の数が増加し、パラメータがばらつくという問題点もある。我々の Improved Expand-and-Truncate Learning Algorithm (IETL) [7][8] では、ETL よりも中間層の数を減らすことに成功したが、パラメータのばらつきは改善されなかった。

本論文では、ETL アルゴリズムを基礎にした新しい学習アルゴリズムを提案する。本手法は教師信号に実際には入力されないバイナリでない仮想入力を加えることにより中間層の数を減らし、パラメータのばらつきを少なくできるアルゴリズムであることを計算機実験により示す。

2. Binary Neural Networks (BNN)

本論文において教師信号は N 入力 1 出力のブール関数

によって与えられるものとする：

$$Y = B(X_i), X_i = (X_i^1, \dots, X_i^N), X_i^j \in \{0,1\}, \\ Y \in \{0,1\}, i = 1 \sim 2^N$$

本稿で構成される BNN はフィードフォワード型の三層構造で次式のように定義される。また図 1 に BNN 構成図を示す。入力層は入力そのまま出力するニューロンである。

$$\begin{cases} z_i = \text{sgn}\left(\sum_{j=1}^i w_{ij}^h x_j - T_i^h\right) \\ y = \text{sgn}\left(\sum_{i=1}^i w_i^o z_i - T^o\right) \end{cases} \quad (1)$$

ただし、 y は出力、 x_i は i 番目のビットの入力、 z_i は i 番目の中間層の出力、 w_{ij}^h は i 番目の入力層から j 番目の中間層への結合パラメータ、 w_i^o は i 番目の中間層から出力層へのパラメータ、 T_i^h は i 番目の中間層のしきい値、 T^o は出力層のしきい値である。また y, x_i, z_i は binary、その他は全て整数値である。

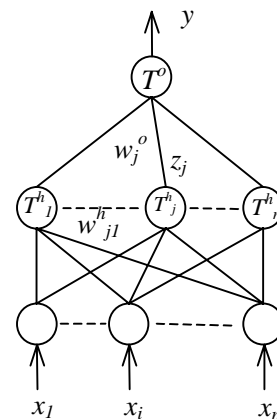


図 1. 三層 BNN

3. 学習アルゴリズム

N -ビット入力 1 出力の教師信号を N 次元超立方体

(N-dimensional hypercube ab.N-HC)として考えて、教師信号の各要素を vertex、特に出力1のものを true vertex、出力0のものを false vertex、出力がどちらでも良いものを don't care と記述する。そして true vertex のハミング距離が近いもの 2 つを幾つか選んでその中点を仮想入力として true vertex に加える。表1の教師信号[4]にはハミング距離が1違うもの間に仮想 vertex 15個を加えた。この理由については、後述する。例 01010 と 00010 の間には(0,1/2,0,1,0)が加えられる。

表1. 教師信号

出力 1			
00010	01000	01001	01010
01011	01110	11010	00100
00101	00111	01101	10000
10100	11100	10111	
出力 0			
00000	00001	00011	00110
01100	01111	10001	10011
10101	10110	11000	11001
11011	11101	11110	11111
10010			

以上の準備のもとで以下のように学習させる。

step 1: true vertex の集合を set included true vertex (SITV) [5]と定義し、その要素を $v_k=(v_k^1, \dots, v_k^n), k=1 \sim C_0$ とする。quantized center of gravity (QCG) [8]を用いて core vertex [8]を選び、その core vertex からハミング距離を基に SITV を増やしてゆき最も多くの true vertex と全ての false vertex を分離する超平面 (hyperplane ab.HP) を以下のように求める。:n-HC の重心(1/2, ..., 1/2)と SITV の重心($C_1/C_0, \dots, C_n/C_0$)それぞれを中心とした超球体 (hypersphere ab.HS) HS_1, HS_2 を考える。

$$\text{ただし } C_i = \sum_{k=1}^{C_0} v_k^i, r \text{ は半径}$$

$$HS_1 = \left(x_1 - \frac{1}{2}\right)^2 + \left(x_2 - \frac{1}{2}\right)^2 + \dots + \left(x_n - \frac{1}{2}\right)^2 = \frac{n}{4} \quad (2)$$

$$HS_2 = \left(x_1 - \frac{C_1}{C_0}\right)^2 + \left(x_2 - \frac{C_2}{C_0}\right)^2 + \dots + \left(x_n - \frac{C_n}{C_0}\right)^2 = r^2 \quad (3)$$

この HS_1 と HS_2 の交わる HP は(4)式で記述される。

$$HP: (2C_1 - C_0)x_1 + (2C_2 - C_0)x_2 + \dots + (2C_2 - C_0)x_2 - T = 0 \quad (4)$$

$$\text{ただし, } T = r^2 C_0 - \sum_{i=1}^n \frac{C_i^2}{C_0}$$

この HP によって分離される vertex の集合を以下のように記述する、

$$U = \{v_1, \dots, v_m\}, v_n = (u_{n1}, \dots, u_{nN})$$

$$D = \{\bar{v}_1, \dots, \bar{v}_l\}, \bar{v}_n = (u_{d1}, \dots, u_{dN})$$

すると

$$\begin{cases} \sum_{i=1}^n (2C_i - C_0)v_{ui} - T \geq 0 \\ \sum_{i=1}^n (2C_i - C_0)v_{di} - T < 0 \end{cases}$$

このとき、式(5)が成り立つ

$$\sum_{i=1}^n (2C_i - C_0)v_{ui} > \sum_{i=1}^n (2C_i - C_0)v_{di} \quad (5)$$

(4)においてUの最小値を t_{\min} , Dの最大値を f_{\max} とすると、UとDを分離するHPは(7)式で表される。

$$(2C_1 - C_0)x_1 + (2C_2 - C_0)x_2 + \dots + (2C_2 - C_0)x_2 - T^h = 0 \quad (7)$$

$$\text{ただし, } T^h = \left[\frac{t_{\min} + f_{\max}}{2} \right]$$

n=2 の例を図2に示す。

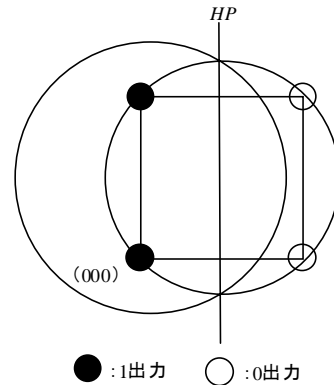


図2. n=2 のHP

step 2: 分離できた true vertex を don't care をして、true vertex から消去し、step 1 へ戻る。

step 3: 全ての true vertex が分離できたら終了し、出力層のパラメータを以下のように定める。

出力層のパラメータ

どの true vertex が入力されても、最低 1 つは 1 を出力する中間層が存在し、また、どの false vertex が入力されても、中間層は全て 0 を出力する。よって、出力層は OR 演算を実現すれば正しい結果が得られる。ゆえにパラメータは $w_j^o = T^o = 1$ となる。教師信号[4]の各 vertex が入力されたときの各中間層の出力の様子を表 2 に示す。

表 2. 各中間層の出力の様子 (*は0もしくは1)

入力信号			z1	z2	z3	z4	出力
00010	01000	01001					1
01010	01011	01110	1	*	*	*	
11010							
00100	00101	00111	0	1	*	*	
01101							
10000	10100	11100	0	0	1	*	
	10111		0	0	0	1	
すべてのfalse vertex			0	0	0	0	0

・従来法との比較

HP は n-HC と SITV の重心の 2 点によって求められるので仮想入力のない従来法[5]においては、ある true vertex が与えられた場合その個数によって求められる HP の数も限定される。本手法ではこれに対し、仮想入力を加えることによって重心と分離 HP を柔軟に変化させることができ、より好ましい分離 HP を得ることができる。仮想入力は (0, 1/2, 0, 1, 0) の様に 1/2 のビットを含んでいるが(7)式の構造より求められるパラメータは従来法を同様に整数である。図 3 は 3-bit 関数[5]の幾何学的表現による HP の様子である。仮想入力を加えることにより分離に必要な HP が減少した。

また、従来法では出力層は、中間層が増えるごとに結合パラメータのばらつきが大きくなっていくが、本手法では OR 演算を実現することによりパラメータのばらつきが改善できた。

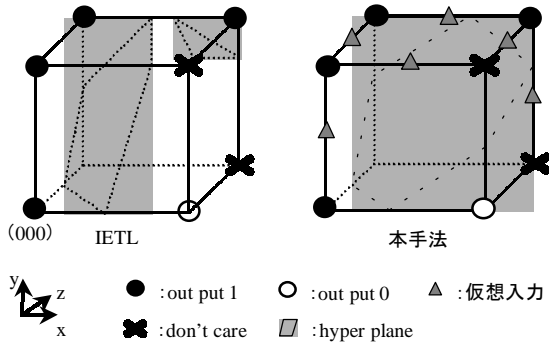


図 3. 3-ビット関数による仮想入力

4. 計算機実験

4.1.A 7-Bit Function

A 7-Bit Function[5]は 1 出力 36 個、0 出力 36 個、7 ビットの教師信号である。残りの 56 個の入力は don't care である。学習させたところ ETL では中間層 7 つが必要であり、また IETL でも 3 つとなったが、本手法では 2 つとなった。この教師信号にはハミング距離が 2 違うものの中点を仮想入力として加えた。求められた HP を示す。

1st HP:

$$7x_1 - x_2 - 12x_3 - 13x_4 - 11x_5 - 13x_6 - 6x_7 + 22 = 0$$

2nd HP:

$$2x_1 - 4x_2 + 7x_3 - 4x_4 + 7x_5 - 4x_6 + 7x_7 - 18 = 0$$

表 3. 7 ビット関数

出力 0		出力 1	
0000110	0110100	1000001	0000000
0001010	0110110	1000010	0000001
0001100	0111000	1000100	0000010
0001110	0111010	1000101	0000011
0001111	0111100	1001000	0000100
0010010	0111101	1010000	0000101
0010100	0111110	1010001	0001000
0010110	0111111	1010100	0001001
0010111	1001101	1010101	0010000
0011000	1001110	1010111	0010001
0011010	1001111	1011101	0010101
0011011	1010110	1100000	0100000
0011100	1011010	1100100	0100001
0011101	1011100	1100101	0100010
0011110	1011110	1100110	0100100
0011111	1011111	1101100	0101000
0101100	1111100	1110100	0110000
0101110	1111110	1110101	1000000

4.2. Two Spirals

Two Spirals[3]は複雑な教師信号のひとつである。今回学習させる信号は 8 ビットで 1 出力 76 個、0 出力 180 個の図 6[8]の様なパターンである。IETL, 本手法によって得られたパラメータを表 4, 5 に示す。中間層が 21 個から 19 個に減少している。また w_j^o が、IETL では大きくばらついているが、本手法では全て 1 となっている。この教師信号にはハミング距離が 2 違うものの中点を仮想入力として加えた。

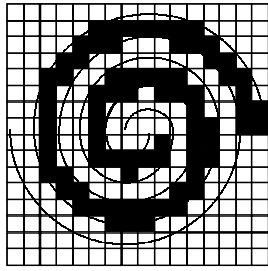


図 5. Two Spirals problem

4. むすび

BNN の新学習アルゴリズムを提案した。仮想入力を導入することにより従来法より柔軟な HP が求められることができ、中間層の数を減少させることができた。複雑な問題への学習に対して、パラメータのばらつきを少なくすることができた。今後の課題を以下に挙げる。

1. 仮想入力の適切な挿入法の考察
2. 現実的な問題への応用

参考文献

- [1] D.L.Gray & A.N.Michel, "A training algorithm for binary feed forward neural networks," IEEE Trans. Neural Networks, vol.3, no.2, pp.176-194, March.1992.
- [2] M.Muselli, "On sequential construction of binary neural networks," IEEE Trans. Neural Networks, vol.6, no.3, pp.678-690, May.1995.
- [3] F.M.Frattale & G.Martinelli, "A constructive algorithm for binary neural networks: The oil-spot algorithm," IEEE Trans. Neural Networks, vol.6, no.3, pp.794-797, May.1995.
- [4] S.K.Park & J.K.Kim, "Geometrical learning algorithm for multilayer neural networks in binary field," IEEE Trans. Computers, vol.42, no.8, pp.988-992, Aug.1993.
- [5] J.H.Kim & S.K.Park, "The geometrical learning of binary neural networks," IEEE Trans. Neural Networks, vol.6, no.1, pp.237-247, Jan.1995.
- [6] S.Young & T.Downs, "CARVE: A constructive algorithm for real-valued examples," IEEE Trans. Neural Networks, vol.9, no.6, pp1180-1190, Nov.1998.
- [7] A.Yamamoto & T.Saito, "An improved expand-and-truncate learning," Proc.IEEE International Conference on Neural Networks, ICNN'97, pp.1111-1116, 1997.
- [8] A.Yamamoto & T.Saito, "A flexible learning algorithm for binary neural networks," IEICE Fundamentals, vol. E81-A, no.9, Sep.1998.
- [9] K.Fukunaga, "Introduction to statistical pattern recognition, 2nd ed." New York: Academic, 1990.

表4. 本手法によって two spiral を学習させて得られたパラメータ

j番目の の中間層	重みとしきい値									
	w_{j1}^h	w_{j2}^h	w_{j3}^h	w_{j4}^h	w_{j5}^h	w_{j6}^h	w_{j7}^h	w_{j8}^h	T_j^h	w_j^o
1	-69	10	102	-73	76	-91	87	6	196	1
2	18	-18	-3	5	11	14	18	-18	49	
3	7	1	7	2	7	-7	-7	0	22	
4	-1	26	-22	-22	25	25	-28	1	55	
5	-42	36	19	25	4	30	35	-38	155	
6	6	-6	-5	-4	-1	-6	6	5	12	
7	-25	-49	59	30	-55	98	-60	21	158	
8	-4	4	-1	3	-4	-3	4	4	12	
9	-3	3	-3	3	3	-3	-2	-1	7	
10	-2	-2	2	-2	-2	2	2	-1	5	
11	-2	-2	2	-2	2	-2	-2	-1	3	
12	5	-8	8	0	2	8	-6	1	24	
13	2	-2	1	2	-2	2	2	2	10	
14	2	2	-2	-2	-2	2	2	-1	7	
15	-1	-2	2	1	1	2	-2	-2	6	
16	6	8	-8	-2	3	2	-7	-7	17	
17	-1	1	-1	-2	-2	2	-2	-2	3	
18	1	1	-1	1	1	-1	1	1	6	
19	1	-1	1	-1	-1	1	-1	-1	3	

表5. IETL によって two spiral を学習させて得られたパラメータ

j番目の の中間層	重みとしきい値									
	w_{j1}^h	w_{j2}^h	w_{j3}^h	w_{j4}^h	w_{j5}^h	w_{j6}^h	w_{j7}^h	w_{j8}^h	T_j^h	w_j^o
1	2	2	10	-10	10	-10	6	0	23	10946
2	-42	-42	-30	2	-14	-18	-10	0	-43	-6765
3	-11	11	5	3	1	1	11	-11	26	4181
4	11	21	13	-21	-21	1	-15	9	29	-2584
5	5	-7	13	3	-9	13	-11	7	29	1597
6	20	30	16	12	12	24	16	8	101	-987
7	6	-4	-6	2	2	-6	6	4	15	610
8	19	-9	19	1	19	-7	-3	5	30	-377
9	9	-9	-1	3	3	5	9	-9	23	233
10	-11	11	1	11	-13	-11	-11	-11	10	-144
11	4	12	-6	-6	12	8	-12	2	27	89
12	21	-31	15	-25	7	17	-3	-3	31	-55
13	-9	9	3	3	-5	-1	9	9	27	34
14	-10	22	-22	-8	16	2	12	-4	31	-21
15	2	-6	10	2	10	-10	-10	0	17	13
16	7	-19	19	19	-9	-7	11	-9	32	-8
17	-11	-13	13	-13	-13	11	13	1	16	5
18	9	7	-11	11	-11	3	-3	1	21	-3
19	-5	17	-17	-17	-17	17	-5	-15	8	2
20	-18	14	18	2	18	2	0	2	41	-1
21	-2	-2	2	2	2	2	-2	0	-5	1

キーワード.

フィードフォワードニューラルネット、バイナリニューラルネット、教師あり学習、線形分離、超平面

Summary.

An efficient learning algorithm for binary neural networks

Masanori Shimada and Toshimichi Saito
Department of Electrical and Electronic Engineering, Hosei University

This paper proposes a novel efficient geometrical learning algorithm for binary neural networks: a feed-forward neural network having binary inputs and outputs. Our supervised learning algorithm uses flexible and speedy linear separation method using virtual inputs. As compared with conventional algorithm, our algorithm can reduce the number of hidden layers and improve the variation of the connection parameters for complex teacher signal. The learned networks are suited for hardware implementation

Keywords.

binary neural networks, feed forward neural networks, supervised learning, linearly separation, hyperplane