

ニューラルネットワークの関数近似における シグモイド関数の問題点

田口 功*

The Problem of a Sigmoid Function in the Function Approximation of the Neural Network

Isao TAGUCHI

In this paper I show how the generalizing ability of the neural network is used, and function approximation is realized. What kind of change in learning does a change in temperature of a sigmoid function produce? Does it influence the output of the neural network? It is known that it aids learning speed. Here I show that the trigonometric function $\cos \theta$ was input, and $\cos (\theta/2)$ was output. When the temperature was about 1, generalizing ability as a neural network could be shown as a result of the experiment. And when the temperature was low at the time of network learning, the mean squared error curve sometimes showed a vibration. When temperature was lowered further, accurate learning stopped, and all output became equal. The reason that all output becomes equal at the time of 1 input 1 output is demonstrated at the end based on backpropagation

*たぐち・いさお：敬愛大学国際学部助教授 ニューロ・コンピューティング
Associate Professor of Computer Science, Faculty of International Studies Keiai
University; neurocomputing.

learning laws from the viewpoint of ceremony. In other words, the reason all output becomes equal depends on all output being decided only by the input when the part of the linear of a sigmoid function is used and the weight coefficient renewal of neurons is done when learning is finished.

関数近似は、ニューラルネットワークの汎化能力を利用して実現できる。シグモイド関数の温度変化は、学習にどのような変化を与えるか。ニューラルネットワークの出力に影響を与えるか？ 学習速度に影響することは、良く知られている。本稿では、三角関数 $\cos \theta$ を入力とし $\cos(\theta/2)$ を出力としたネットワークを考えた。実験の結果、温度が1程度のときは、ニューラルネットワークとしての汎化能力は、発揮できた。また、ネットワーク学習時、温度が低いときには、平均2乗誤差曲線が振動性を示すことがある。さらに温度が低くなると、正しい学習が行われなくなり、すべての出力が等しくなった。最後に、1入力1出力時、すべての出力が等しくなることの理由を逆伝搬学習則を基に式的に示した。すなわち、学習の終了時、シグモイド関数の線形部分が使われてニューロンの重み係数更新が行われる場合、すべての出力が等しくなってしまう理由は、入力のみによってすべての出力が決定されてしまうことによる。

1. はじめに

シグモイド関数が、ニューラルネットワークにおいてよく使用される。本研究においては、ニューラルネットワークの汎化能力を利用して関数を近似する問題を考える。汎化の定義は明白ではないが、本論文では $\cos \theta$ を入力とし、 $\cos(\theta/2)$ を実現出来る能力を汎化能力と定義する。シグモイド関数の温度（本稿では、シグモイド関数を $f(u) = 1/(1+e^{-ku})$ としたときの k を温度と呼ぶこととする）の変化によって学習の変化は起こるものであろうかという問題を考察した。温度の大きさによる学習の可、不可そして汎化

能力については余り検討されていないようである。本稿では、問題を簡単化するためにニューロンモデルは中間層、出力層共に同じ温度に設定するものとして温度の大小変化を考えた。そして、正しい汎化学習が行われなくなることは、実験的、理論的に検討されていない。本研究では、シグモイド関数を用い、温度が1程度のときは、ニューラルネットワークとしての汎化能力機能が発揮できることを実験で示した。また、温度が低いとき(0.08程度)には、学習の際に、誤差曲線が学習回数に対して振動性を示すこと、さらに、温度が低くなる時(0.008程度)には、正しい学習が行われなくなることを実験的に示した。特に、すべての出力が等しくなることの理由を逆伝搬学習則を表す式を基に理論的に示した。本稿でのニューラルネットワーク構成は1入力1出力とし、中間層のニューロン数は15程度とした。

2. シグモイド関数の温度による線形性と非線形性

ニューラルネットワークを考えると、シグモイド関数や汎化能力は、非常に基本的に知られている。従来から、温度は主に学習速度に影響を与えるものと考えられていた。温度による学習への可、不可の影響はあるかという問題はあまり考察されていない。本研究では、汎化能力を利用して基本関数(ここでは、三角関数 $\cos \theta$ を入力とし、 $\cos(\theta/2)$ を出力とした)を覚え込ませることができるかどうかを検討した。学習時において、シグモイド関数の温度を小さくすればするほど、入出力関係が線形性を示す。この様子を図 2.1、図 2.2 に示す。 $k=0.008$ および $k=0.08$ のときのグラフをみると、入力 $\pm x$ に対して強い線形性を示す。図 2.2 をみると、 $k=0.8$ のときは原点付近は、線形性を示すが、原点から離れるにしたがって非線形性が徐々に出てくる曲線となっている。 $k=1.0$ を超えるにしたがって原点付近の線形性はなくなる。

図2.1 $k=0.008$ および $k=0.08$ の時のシグモイド関数

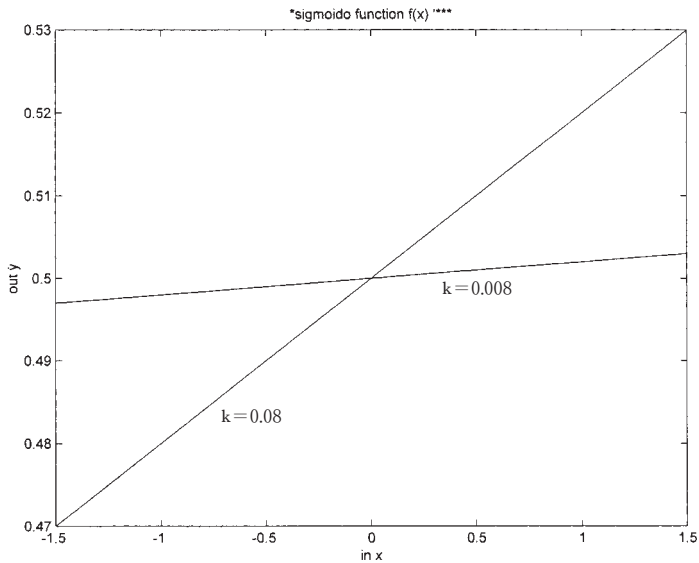
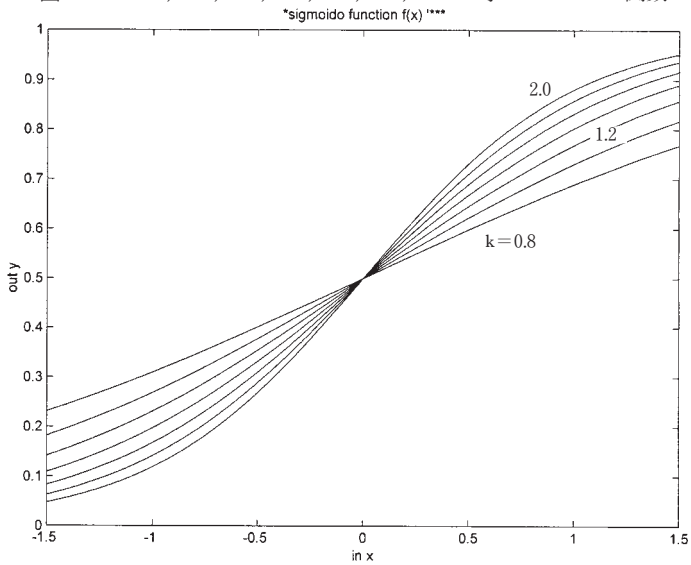


図2.2 $k=0.8, 1.0, 1.2, 1.4, 1.6, 1.8, 2.0$ の時のシグモイド関数



3. シグモイド関数による重み係数更新と同一出力性

最初に、図3.1に示すような3層ニューラルネットワークを考える。ここで、 $\mathbf{x}^{(p)}$ は p 番目の入力パターンベクトル、 $t^{(p)}$ は、それに対応する目標値である。 $z^{(p)}$ は、実際の出力とする。 $W_{ji}^{(1)}$ は、入力層と中間層の間の重み係数、 $W_{kj}^{(h)}$ は、中間層と出力層の間の重み係数である。出力層と中間層にある素子は、入力 u に対して、シグモイド関数を使用すると、

$$f(u) = 1 / (1 + e^{-ku}) \quad (3-1)$$

となる。学習は、

$$E = \sum E^{(p)} = \sum \sum \{ (t_k^{(p)} - z_k^{(p)})^2 / 2 \} \quad (3-2)$$

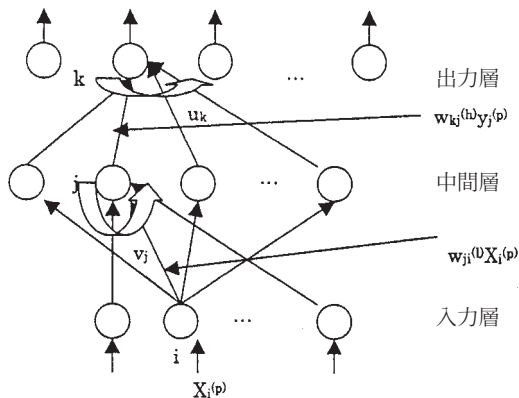
の総誤差 E を最小化することである。パターン $\mathbf{x}^{(p)}$ が入力として加わった場合、出力誤差は、

$$E^{(p)} = \sum \{ (t_k^{(p)} - z_k^{(p)})^2 / 2 \} \quad (3-3)$$

となり、これを減少させるように $W_{kj}^{(h)}$ と $W_{ji}^{(1)}$ を調節することになる。そのために、 $W_{kj}^{(h)}$ と $W_{ji}^{(1)}$ にそれぞれ

$$\Delta W_{kj}^{(h)} = - \varepsilon \partial E^{(p)} / \partial W_{kj}^{(h)} \quad (3-4)$$

図3.1 3層ニューラルネットワーク
 $z_k^{(p)}$ 出力



と

$$\Delta W_{ji}^{(1)} = -\varepsilon \partial E^{(p)} / \partial W_{ji}^{(1)} \quad (3-5)$$

を加える（最急降下法）。ここで、 ε は、学習係数である。

出力層にある素子 k の入力 u_k は、

$$u_k = \sum W_{kj}^{(h)} y_j^{(p)} \quad (3-6)$$

となり、出力は、

$$z_k^{(p)} = f(u_k) \quad (3-7)$$

となる。また、偏微分の連鎖律を考慮すると、

$$\begin{aligned} -\partial E^{(p)} / \partial W_{kj}^{(h)} &= -E^{(p)} / \partial u_k \cdot \partial u_k / \partial W_{kj}^{(h)} \\ &= -E^{(p)} / \partial z_k^{(p)} \cdot \partial z_k^{(p)} / \partial u_k \cdot \partial u_k / \partial W_{kj}^{(h)} \\ &= (t_k^{(p)} - z_k^{(p)}) z_k^{(p)} (1 - z_k^{(p)}) \cdot y_j^{(p)} \end{aligned} \quad (3-8)$$

が成り立つ。ここで、 $\delta_k^{(h)} = (t_k^{(p)} - z_k^{(p)}) z_k^{(p)} (1 - z_k^{(p)})$ とおくと、

$$-\partial E^{(p)} / \partial W_{kj}^{(h)} = \delta_k^{(h)} \cdot y_j^{(p)} \quad (3-9)$$

したがって、

$$\begin{aligned} \Delta W_{kj}^{(h)} &= \varepsilon \delta_k^{(h)} \cdot y_j^{(p)} \\ &= \varepsilon \cdot (t_k^{(p)} - z_k^{(p)}) z_k^{(p)} (1 - z_k^{(p)}) y_j^{(p)} \end{aligned} \quad (3-10)$$

となり、出力層と中間層の間の重み係数の更新を終える。次に、入力層と中間層との間の重み係数更新を述べる。中間層にある素子 j の入力 v_j は、

$$v_j = \sum W_{ji}^{(1)} x_i^{(p)} \quad (3-11)$$

となり、シグモイド関数を使用した場合の出力は、

$$y_j^{(p)} = f(v_j) \quad (3-12)$$

となる。同様に偏微分の連鎖律を考慮すると、

$$\begin{aligned} -\partial E^{(p)} / \partial W_{ji}^{(1)} &= -E^{(p)} / \partial v_j \cdot \partial v_j / \partial W_{ji}^{(1)} \\ &= -E^{(p)} / \partial v_j \cdot x_i^{(p)} \end{aligned} \quad (3-13)$$

となる。ここで、 $\delta_j^{(1)} = -\partial E^{(p)} / \partial v_j$ とおくと

$$\begin{aligned} \delta_j^{(1)} &= -\partial E^{(p)} / \partial y_j^{(p)} \cdot \partial y_j^{(p)} / \partial v_j \\ &= -\partial E^{(p)} / \partial y_j^{(p)} \cdot y_j^{(p)} (1 - y_j^{(p)}) \end{aligned} \quad (3-14)$$

であり、さらに $\partial E^{(p)} / \partial y_j^{(p)}$ は、

$$\partial E^{(p)} / \partial y_j^{(p)} = \sum \partial E^{(p)} / \partial u_k \cdot \partial u_k / \partial y_j^{(p)}$$

$$= \sum \delta_k^{(h)} W_{kj}^{(h)} \quad (3-15)$$

となるから、

$$\delta_j^{(1)} = (\sum \delta_k^{(h)} W_{kj}^{(h)}) y_j^{(p)} (1 - y_j^{(p)}) \quad (3-16)$$

とおき、まとめると、

$$\begin{aligned} \Delta W_{ji}^{(1)} &= \varepsilon \delta_j^{(1)} x_i^{(p)} \\ &= \varepsilon (\sum \delta_k^{(h)} W_{kj}^{(h)}) y_j^{(p)} (1 - y_j^{(p)}) x_i^{(p)} \end{aligned} \quad (3-17)$$

となる。ここでシグモイド関数の線形性を仮定し、

$$\begin{aligned} f(u) &= 1 / (1 + e^{-ku}) \\ &\doteq au \end{aligned} \quad (3-18)$$

とおく。a は、k の値が小さくなればなるほど小さくなる。また、5 パターンの入力を仮定し、入出力関係を検討する。

5 パターンの入力に対する出力層へのシグモイド関数入力 $u_k^{(1)} u_k^{(2)} u_k^{(3)} u_k^{(4)} u_k^{(5)}$ は、それぞれ、

$$\begin{aligned} u_k^{(1)} &= \sum W_{kj}^{(h)} y_j^{(1)} \\ u_k^{(2)} &= \sum W_{kj}^{(h)} y_j^{(2)} \\ u_k^{(3)} &= \sum W_{kj}^{(h)} y_j^{(3)} \\ u_k^{(4)} &= \sum W_{kj}^{(h)} y_j^{(4)} \\ u_k^{(5)} &= \sum W_{kj}^{(h)} y_j^{(5)} \end{aligned} \quad (3-19)$$

となる。ここで、j は、中間層の数であり、 $y_j^{(1)}$ 、 $y_j^{(2)}$ 、 $y_j^{(3)}$ 、 $y_j^{(4)}$ 、 $y_j^{(5)}$ は 5 パターンに対する中間層出力である。入力の大きさ（本稿の場合は、1 入力 1 出力）に応じて、 $y_j^{(1)}$ 、 $y_j^{(2)}$ 、 $y_j^{(3)}$ 、 $y_j^{(4)}$ 、 $y_j^{(5)}$ の値が異なっている。1 回 1 回学習の更新を行うのであるが、重み係数はすべてのパターンに対して更新後は変わらない。出力層のシグモイド関数入力 $u_k^{(1)} u_k^{(2)} u_k^{(3)} u_k^{(4)} u_k^{(5)}$ は、当然異なる。ここで $f(u) \doteq au$ の関係を代入すると、

$$\begin{aligned} f(u_k^{(1)}) &= f(\sum W_{kj}^{(h)} y_j^{(1)}) = a \sum W_{kj}^{(h)} y_j^{(1)} \\ f(u_k^{(2)}) &= f(\sum W_{kj}^{(h)} y_j^{(2)}) = a \sum W_{kj}^{(h)} y_j^{(2)} \\ f(u_k^{(3)}) &= f(\sum W_{kj}^{(h)} y_j^{(3)}) = a \sum W_{kj}^{(h)} y_j^{(3)} \\ f(u_k^{(4)}) &= f(\sum W_{kj}^{(h)} y_j^{(4)}) = a \sum W_{kj}^{(h)} y_j^{(4)} \\ f(u_k^{(5)}) &= f(\sum W_{kj}^{(h)} y_j^{(5)}) = a \sum W_{kj}^{(h)} y_j^{(5)} \end{aligned} \quad (3-20)$$

となる。ここで、中間層にある素子 j の入力 v_j は、 $v_j = \sum W_{ji}^{(1)} x_i^{(p)}$ という関係にあるから、

$$\begin{aligned} v_j^{(1)} &= \sum W_{ji}^{(1)} x_i^{(1)} \\ v_j^{(2)} &= \sum W_{ji}^{(1)} x_i^{(2)} \\ v_j^{(3)} &= \sum W_{ji}^{(1)} x_i^{(3)} \\ v_j^{(4)} &= \sum W_{ji}^{(1)} x_i^{(4)} \\ v_j^{(5)} &= \sum W_{ji}^{(1)} x_i^{(5)} \end{aligned} \quad (3-21)$$

が成立する。同様に、 $y_j^{(p)} = f(v_j) = f(\sum W_{ji}^{(1)} x_i^{(p)})$ の関係にあるから $f(v) = av$ の関係を当てはめれば、

$$\begin{aligned} y_j^{(1)} &= f(v_j^{(1)}) = f(\sum W_{ji}^{(1)} x_i^{(1)}) = a \sum W_{ji}^{(1)} x_i^{(1)} \\ y_j^{(2)} &= f(v_j^{(2)}) = f(\sum W_{ji}^{(1)} x_i^{(2)}) = a \sum W_{ji}^{(1)} x_i^{(2)} \\ y_j^{(3)} &= f(v_j^{(3)}) = f(\sum W_{ji}^{(1)} x_i^{(3)}) = a \sum W_{ji}^{(1)} x_i^{(3)} \\ y_j^{(4)} &= f(v_j^{(4)}) = f(\sum W_{ji}^{(1)} x_i^{(4)}) = a \sum W_{ji}^{(1)} x_i^{(4)} \\ y_j^{(5)} &= f(v_j^{(5)}) = f(\sum W_{ji}^{(1)} x_i^{(5)}) = a \sum W_{ji}^{(1)} x_i^{(5)} \end{aligned} \quad (3-22)$$

となる。これらの関係を、

$$\begin{aligned} f(u_k^{(1)}) &= f(\sum W_{kj}^{(h)} y_j^{(1)}) = a \sum W_{kj}^{(h)} y_j^{(1)} \\ f(u_k^{(2)}) &= f(\sum W_{kj}^{(h)} y_j^{(2)}) = a \sum W_{kj}^{(h)} y_j^{(2)} \\ f(u_k^{(3)}) &= f(\sum W_{kj}^{(h)} y_j^{(3)}) = a \sum W_{kj}^{(h)} y_j^{(3)} \\ f(u_k^{(4)}) &= f(\sum W_{kj}^{(h)} y_j^{(4)}) = a \sum W_{kj}^{(h)} y_j^{(4)} \\ f(u_k^{(5)}) &= f(\sum W_{kj}^{(h)} y_j^{(5)}) = a \sum W_{kj}^{(h)} y_j^{(5)} \end{aligned} \quad (3-23)$$

に代入すると、

$$\begin{aligned} f(u_k^{(1)}) &= f(\sum W_{kj}^{(h)} y_j^{(1)}) = a \sum W_{kj}^{(h)} (a \sum W_{ji}^{(1)} x_i^{(1)}) \\ &= a^2 \sum \sum W_{kj}^{(h)} W_{ji}^{(1)} x_i^{(1)} \\ f(u_k^{(2)}) &= f(\sum W_{kj}^{(h)} y_j^{(2)}) = a \sum W_{kj}^{(h)} (a \sum W_{ji}^{(1)} x_i^{(2)}) \\ &= a^2 \sum \sum W_{kj}^{(h)} W_{ji}^{(1)} x_i^{(2)} \\ f(u_k^{(3)}) &= f(\sum W_{kj}^{(h)} y_j^{(3)}) = a \sum W_{kj}^{(h)} (a \sum W_{ji}^{(1)} x_i^{(3)}) \\ &= a^2 \sum \sum W_{kj}^{(h)} W_{ji}^{(1)} x_i^{(3)} \\ f(u_k^{(4)}) &= f(\sum W_{kj}^{(h)} y_j^{(4)}) = a \sum W_{kj}^{(h)} (a \sum W_{ji}^{(1)} x_i^{(4)}) \\ &= a^2 \sum \sum W_{kj}^{(h)} W_{ji}^{(1)} x_i^{(4)} \end{aligned}$$

$$\begin{aligned}
 f(\mathbf{u}_k^{(5)}) &= f(\sum W_{kj}^{(h)} y_j^{(5)}) = a \sum W_{kj}^{(h)} (a \sum W_{ji}^{(1)} x_i^{(5)}) \\
 &= a^2 \sum \sum W_{kj}^{(h)} W_{ji}^{(1)} x_i^{(5)}
 \end{aligned}
 \tag{3-24}$$

となる。したがって、 $f(\mathbf{u}) \approx a\mathbf{u}$ および $f(\mathbf{v}) \approx a\mathbf{v}$ の関係が成立するとき、各入力パターンに対する各出力は、入力のみによって変化することとなる。ここで、重み係数は逆伝搬学習則により更新がなされる。ここでは、各パターンの重み係数の補正を行うのであるが、一般的に更新量は、各パターンの最急降下分の平均値が使われている。本稿でも平均値を使用した。ここで、 $\mathbf{C} = a^2 \sum \sum W_{kj}^{(h)} W_{ji}^{(1)}$ とおくと、

$$\begin{aligned}
 f(\mathbf{u}_k^{(1)}) &= \mathbf{C}\mathbf{x}_i^{(1)} \\
 f(\mathbf{u}_k^{(2)}) &= \mathbf{C}\mathbf{x}_i^{(2)} \\
 f(\mathbf{u}_k^{(3)}) &= \mathbf{C}\mathbf{x}_i^{(3)} \\
 f(\mathbf{u}_k^{(4)}) &= \mathbf{C}\mathbf{x}_i^{(4)} \\
 f(\mathbf{u}_k^{(5)}) &= \mathbf{C}\mathbf{x}_i^{(5)}
 \end{aligned}
 \tag{3-25}$$

となる。さらに、 a^2 が非常に小さくなる時、入力の大きさ $\mathbf{x}_i^{(1)} \sim \mathbf{x}_i^{(5)}$ に比較して a^2 が大きく出力に影響を与えるものと考えられる。

4. 計算機実験

すべてのニューロンに基本的なシグモイド関数のみを用いた場合の温度変化と学習シミュレーション実験を行った。ネットワーク構成は中間層のニューロン数を15とし、1入力1出力の学習を行った。三角関数 $\cos \theta$ を入力とし、 $\cos(\theta/2)$ を出力とした学習を行った。学習則としては、逆伝搬学習則を用いた。目標値は、1.0000、0.9553、0.8253、0.6216、0.3624とし、学習係数は、0.008一定とした。5個の入力パターンに対して、目標値からの差の2乗の合計を誤差とした。すべてのニューロンの温度を0.7、1.0、0.08、0.008と変化させた時の誤差曲線を学習回数に対して描いた。その結果、温度が0.7の時、24,741回の学習で、出力は0.9551、0.9341、0.8450、0.6124、0.3756となり、1.0の時は、16,546回の学習で、0.9618、0.9446、0.8532、0.6073、0.3705となった。温度が0.7の時の各々の目標値に

対する2乗誤差曲線を図4.1.1から図4.1.5に示す。同様に温度が1.0の目標値に対する2乗誤差曲線を図4.2.1から図4.2.5に示す。温度が0.08の時は、各誤差曲線がすべての入力に対して振動することとなった。187回の学習で、出力は、0.8700、0.8699、0.8695、0.8693、0.8689となった。目標値に対する2乗誤差曲線は、図4.3.1から図4.3.5に示す結果となった。

また、温度の値を小さくし、その値を、0.008とした時は、すべての異なる入力に対して、出力が0.9547となり等しくなってしまう、正しい学習が行われなくなった。2乗誤差曲線を図4.4.1から図4.4.5に示す。

図4.1.1 目標値1.0に対する
2乗誤差曲線 (k=0.7)

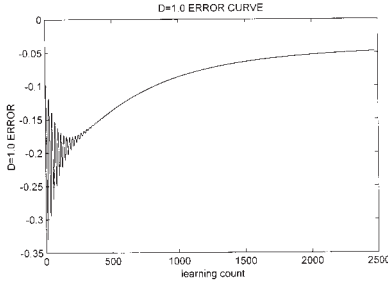


図4.1.2 目標値0.9553に対する
2乗誤差曲線 (k=0.7)

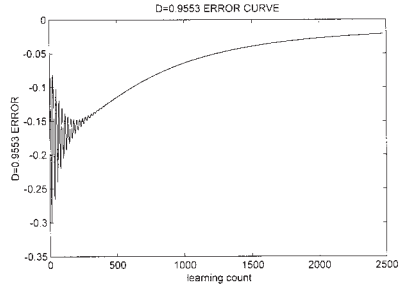


図4.1.3 目標値0.8253に対する
2乗誤差曲線 (k=0.7)

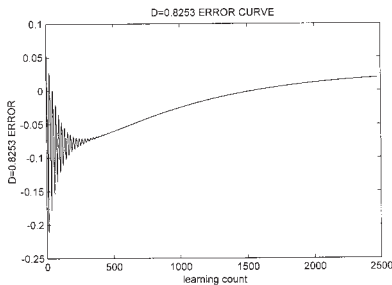


図4.1.4 目標値0.6216に対する
2乗誤差曲線 (k=0.7)

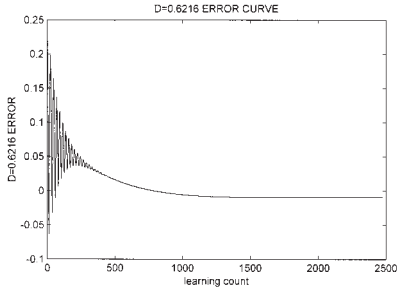


図4.1.5 目標値0.3624に対する
2乗誤差曲線 ($k=0.7$)

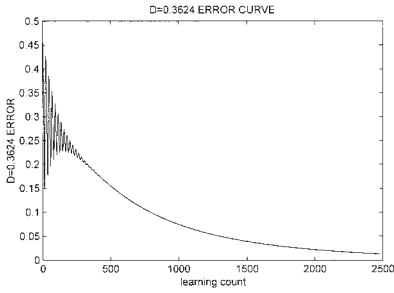


図4.2.1 目標値1.0に対する
2乗誤差曲線 ($k=1.0$)

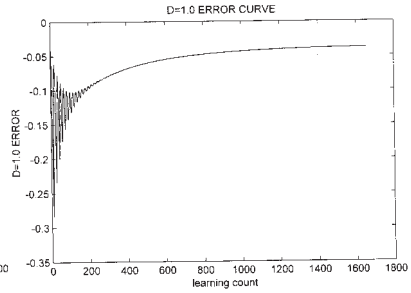


図4.2.2 目標値0.9553に対する
2乗誤差曲線 ($k=1.0$)

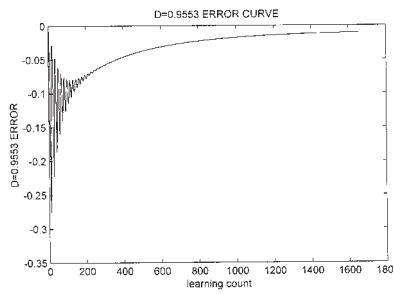


図4.2.3 目標値0.8253に対する
2乗誤差曲線 ($k=1.0$)

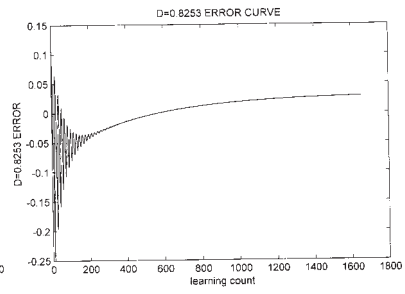


図4.2.4 目標値0.6216に対する
2乗誤差曲線 ($k=1.0$)

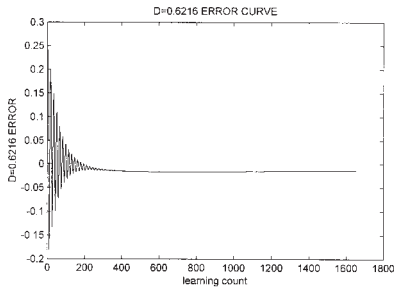


図4.2.5 目標値0.3624に対する
2乗誤差曲線 ($k=1.0$)

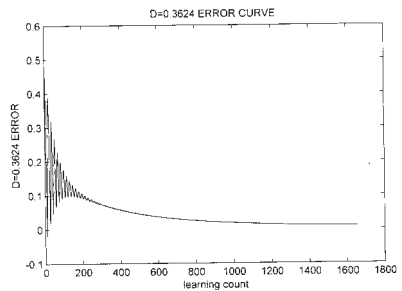


図4.3.1 目標値1.0に対する
2乗誤差曲線 ($k=0.08$)

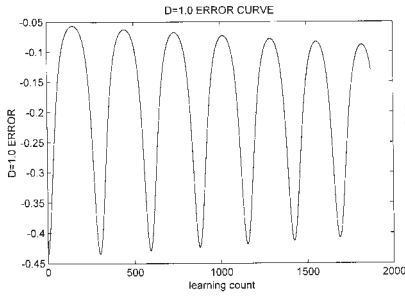


図4.3.2 目標値0.9553に対する
2乗誤差曲線 ($k=0.08$)

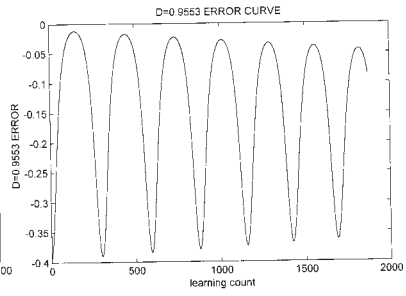


図4.3.3 目標値0.8253に対する
2乗誤差曲線 ($k=0.08$)

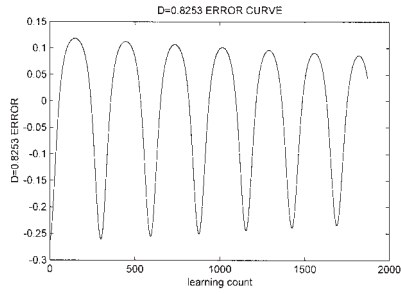


図4.3.4 目標値0.6216に対する
2乗誤差曲線 ($k=0.08$)

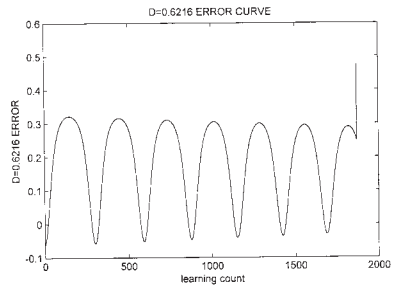


図4.3.5 目標値0.3624に対する
2乗誤差曲線 ($k=0.08$)

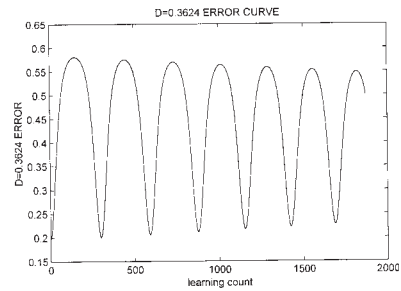


図4.4.1 目標値1.0に対する
2乗誤差曲線 ($k=0.008$)

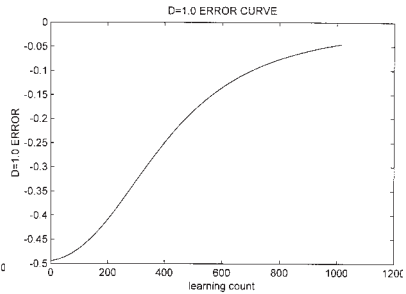


図4.4.2 目標値0.9553に対する
2乗誤差曲線 ($k=0.008$)

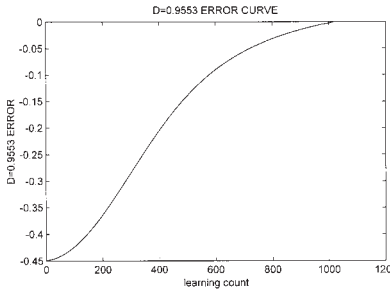


図4.4.3 目標値0.8253に対する
2乗誤差曲線 ($k=0.008$)

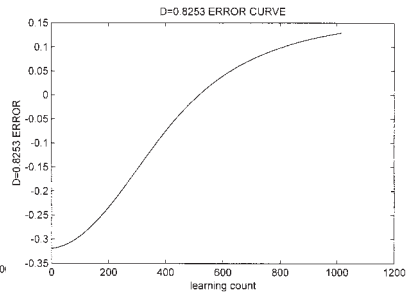


図4.4.4 目標値0.6216に対する
2乗誤差曲線 ($k=0.008$)

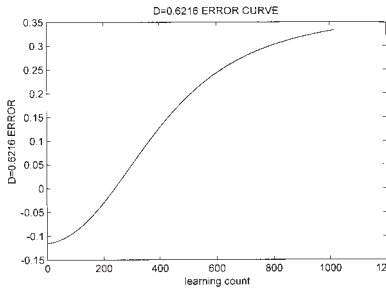
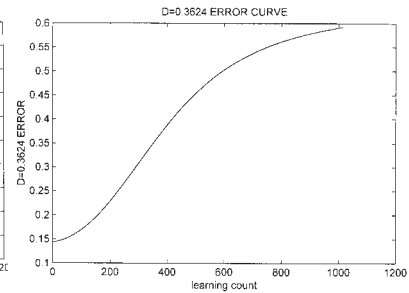


図4.4.5 目標値0.3624に対する
2乗誤差曲線 ($k=0.008$)



5. 考 察

(3-24) 式の関係、すなわち、各出力が入力に比例した関係となるとき、正しい学習は行われなくなる。 k が非常に小さいとき、シグモイド関数は比例関係を示す。実際行った計算機実験でも $k=0.008$ のとき正しい学習は、行われなく等しい出力となった。この場合は、入力に変化しても出力がそれほど変化できないために、同一出力になると考えられる。温度の大きさによる学習の可、不可そして汎化能力については余り検討されていないようであるが汎化能力が低下するという結果となった。本研究では、シグモイド関数を用い、温度が1程度のときは、ニューラルネットワークと

して関数学習をした場合、汎化能力機能が発揮できることを実験で示した。また、温度が低いとき（0.08程度）には、学習の際に、誤差曲線が学習回数に対して振動性を示すこと、さらに、温度が低くなるとき（0.008程度）には、正しい学習が行われなくなることを実験で確かめた。特に、すべての出力が等しくなることの理由も逆伝搬学習則を表す式を基に理論的にまとめた。この結果は、ニューロンモデルを設定するときに学習を行う場合の目安になると考えられる。すなわち、シグモイド関数の欠点は、線形性にあるので、線形性部分をなくすことがひとつの改良点になるという結論を得た。

〔謝辞〕 本研究に対して、研究方法から細かくご指導いただいた、慶応大学理工学部の相吉英太郎教授に感謝いたします。

（参考文献）

R・ビール／T・ジャクソン著（八名和夫監訳）『ニューラルコンピューティング入門』、海文堂、1993年、122-125ページ。

Robert Hecht Nielsen, *Neurocomputing*, 1990, pp.128-137, Addison-wesley.

八名和夫・鈴木義武共著『ニューロ情報処理技術』、海文堂、1992年、73-85ページ。