

## 船橋の定理のニューラルネットワークへの応用

柿崎淳一 秋葉巴也  
弘前大学 情報科学科

多層パーセプトロン型ネットワークの学習アルゴリズムとして誤差逆伝搬法が強力であるが、この方法で学習させた時、ネットワークがコスト関数の局所的極小点に陥る可能性があるので最適解にたどり着けるとい保証はない。更に、最適解が必ずしも厳密解を与えるとは限らない。他方、3層パーセプトロンは、コンパクトなサポートを持つ任意の連続関数を任意の精度で表現できることが船橋によって証明されている。この論文では、船橋の定理をコンパクト領域で定義された2回微分可能な関数に応用し、決められた誤差の範囲で、3層ネットワークにより関数を表現するのにどれ程の隠れ層のユニット数が要求されるかについて研究する。

## Possible Application of Funahashi's Theorem to Neural Networks

JUN-ICHI KAKIZAKI AND TOMOYA AKIBA  
*Department of Information Science, Hirosaki University*

Although the back-propagation method provides a powerful learning algorithm for multilayer-perceptron-type networks, it is not guaranteed that a network can reach to the best optimal solution by this learning method. Moreover, the best optimal solution does not always give an exact solution. On the other hand, it is proved by Funahashi that a multilayer-perceptron is capable to represent any continuous function with compact supports to arbitrary accuracy. In this paper, we apply the theorem to twice differentiable functions defined in a compact region and study what number of hidden units is required to represent a function to an assigned accuracy with a three-layer network.

## 1. はじめに

1986年に、Rumelhart、Hinton、Williams<sup>[1]</sup>により誤差逆伝搬法が提案されて以来、多層ニューラルネットの識別能力は飛躍的に増大した。しかし、この学習法によってニューラルネットが要求された識別関数を厳密に表現できるという保証はない。他方、多層ニューラルネットの関数表現能力に関して、入江一三宅<sup>[2]</sup>、船橋<sup>[3]</sup>の研究がある。入江一三宅は任意の多変数関数を3層のネットワークで表現する積分公式を示したが、2つの問題点が残されていた。1つは、ユニット数が無限個必要になり、実際に実現不可能であることと、2つめは、隠れ層のユニットの出力関数に対して絶対可積分の条件が付くだけで、シグモイド関数である示唆がないことである。船橋はこの2つの問題点を解決した。しかし船橋の研究では、応用上重要と思われる隠れ層のユニット数と近似の誤差の関係について論じられていない。そこで本論文では、誤差を小さくするために必要な隠れ層のユニット数を評価し、具体例について数値的評価も行う。ただし、取り扱う関数の範囲を、コンパクト領域で定義された2回微分可能な関数に限定した。

まず2章で船橋の定理について述べ、3章で積分を有限和に直す時の誤差について論じる。4章では、関数が周期関数であるとき、フーリエ級数の部分和を用いて積分を有限和に直す方法を述べ、近似誤差を評価する。5章で具体的問題についてのシミュレーションの結果を示し、6章でまとめを行う。

## 2. 船橋の定理

船橋<sup>[3]</sup>は次の定理を証明した：

定理

$\phi(x)$ を有界単調増加連続関数とする。 $K$ を $\mathbf{R}^n$ のコンパクト集合とし、 $f(x_1, \dots, x_n)$ を $K$ 上の実数値連続関数とする。この時、任意の $\epsilon > 0$ に対し、整数 $M$ と実定数 $c_i, \theta_i$  ( $i = 1, \dots, M$ )、 $w_{ij}$  ( $i = 1, \dots, M, j = 1, \dots, n$ )が存在し、

$$\bar{f}(x_1, \dots, x_n) = \sum_{i=1}^M c_i \phi \left( \sum_{j=1}^n w_{ij} x_j - \theta_i \right) \quad (2.1)$$

が、 $\max_{x \in K} |f(x_1, \dots, x_n) - \bar{f}(x_1, \dots, x_n)| < \epsilon$ を満たす。

言い替えば、任意の $\epsilon > 0$ に対し、隠れ層の出力関数が $\phi(x)$  (具体的にはシグモイド関数)で、入力層と出力層の出力関数が線形であるような、関数 $\bar{f}(x_1, \dots, x_n)$ を表現する3層ネットワークが存在し、 $\max_{x \in K} |f(x_1, \dots, x_n) - \bar{f}(x_1, \dots, x_n)| < \epsilon$ を満たす。

関数  $\bar{f}(x_1, \dots, x_n)$  は詳しくは次の形を持つ：

$$\bar{f}(x_1, \dots, x_n) = \text{Real} \left[ \left( \frac{2A}{N} \right)^n \sum_{k_0=0}^{N-1} C_{k_0} \sum_{k_1 \dots k_n=0}^{N-1} \psi \left( \sum_{i=1}^n x_i w_{i k_i} - (-A' + \frac{2A'}{N} k_0) \right) \right. \\ \left. \times F(w_{1 k_1}, \dots, w_{n k_n}) \right], \quad (2.2)$$

$$C_{k_0} = \frac{1}{(2\pi)^n \Psi(1)} \frac{2A'}{N} \exp \left[ i \left( -A' + \frac{2A'}{N} k_0 \right) \right], \quad w_{i k_i} = -A + \frac{2A}{N} k_i, \quad (2.3)$$

ここで、 $\psi(x) = \phi(x/\delta + \alpha) - \phi(x/\delta - \alpha)$ 、 $F$  は  $f$  のフーリエ変換、 $\alpha$ 、 $\delta$  は正の実数である。そして、 $\Psi$  は  $\psi$  のフーリエ変換、 $\text{Real}[\ ]$  は括弧内の実部を意味する。この  $\bar{f}(x_1, \dots, x_n)$  は次の積分を有限和で近似したものである：

$$I_{A', A}(x_1, \dots, x_n) = \int_{-A}^A \dots \int_{-A}^A \left[ \int_{-A'}^{A'} \psi \left( \sum_{i=1}^n x_i w_i - w_0 \right) \frac{1}{(2\pi)^n \Psi(1)} \right. \\ \left. \times F(w_1, \dots, w_n) e^{i w_0} dw_0 \right] dw_1 \dots dw_n. \quad (2.4)$$

### 3. 積分を和で近似したことによる誤差評価

ここでは前述の  $I_{A', A}(x_1, \dots, x_n)$  を和で近似した時の誤差を大雑把に評価しよう。まず簡単の為、区間  $[0, L]^2$  上で定義された 2 変数関数  $f(x_1, x_2)$  の区間全体にわたる 2 重積分を有限和で表した時の誤差を評価する。

$x_1$ 、 $x_2$  それぞれについて積分区間を十分大きな正の整数  $N$  で分割し  $\Delta x = \Delta y = \frac{L}{N}$  と置けば、格子点  $(x_i, y_j)$ 、 $(i, j = 1, \dots, N)$  での誤差は平均値の定理とシュワルツの不等式から、

$$\epsilon_{ij} \leq |\nabla f(\mathbf{r}_{ij} + \Delta \mathbf{r}_{ij}(\theta - 1))| |\Delta \mathbf{r}_{ij}| \Delta x \Delta y \\ = |\nabla f(\mathbf{r}_{ij} + \Delta \mathbf{r}_{ij}(\theta - 1))| \left( \frac{L}{N} \right)^3 \sqrt{2}. \quad (3.1)$$

ここで、 $\theta$  は  $f(\mathbf{r}_{ij}) - f(\mathbf{r}_{ij} - \Delta \mathbf{r}_{ij}) = \nabla f(\mathbf{r}_{ij} + \Delta \mathbf{r}_{ij}(\theta - 1)) \cdot \Delta \mathbf{r}_{ij}$ 、 $\{\Delta \mathbf{r}_{ij} = (\Delta x_i, \Delta y_j)\}$  を満たす実数で、 $0 < \theta < 1$  である。 $\frac{L}{N} \ll 1$  であるから、 $\frac{L}{N}$  の最低次までとれば全誤差は  $\epsilon_{ij}$  の 2 つの添え字の和を取り次のように評価できる：

$$E = \sum_{i=1}^N \sum_{j=1}^N \epsilon_{ij} \sim O\left(\frac{1}{N}\right).$$

(2.4) に対しても、その積分区間をそれぞれ  $N$  分割してやれば、同様の評価で全誤差は次のように求まる：

$$\begin{aligned} E &\leq \left(\frac{2}{N}\right)^{n+2} A' A^n \sqrt{(A')^2 + nA^2} \\ &\times \sum_{i_0 i_1 \cdots i_n = 1}^N \left[ \left(\frac{\partial g}{\partial w_0}(w_{0i_0}, \dots, w_{ni_n})\right)^2 + \cdots + \left(\frac{\partial g}{\partial w_n}(w_{0i_0}, \dots, w_{ni_n})\right)^2 \right]^{\frac{1}{2}} \quad (3.2) \\ &\sim O\left(\frac{1}{N}\right). \end{aligned}$$

ただし、 $g(w_0, \dots, w_n) = \psi(\sum_{i=1}^n x_i w_i - w_0) F(w_1, \dots, w_n) e^{iw_0} / (2\pi)^n \Psi(1)$  である。

#### 4. フーリエ級数の部分和に基づく誤差評価

ここでは、関数  $f(x_1, \dots, x_n)$  が周期条件を満たすと仮定して、(2.4) をフーリエ級数の部分和で近似する方法を述べる。まず、次の式で与えられる誤差を評価しよう：

$$\epsilon = \int_{-\infty}^{\infty} \psi\left(\sum_{i=1}^n x_i w_i - w_0\right) e^{iw_0} dw_0 - \int_{-A'}^{A'} \psi\left(\sum_{i=1}^n x_i w_i - w_0\right) e^{iw_0} dw_0. \quad (4.1)$$

これは、 $A'/\delta$  が十分大きい時には、 $|\epsilon| < 2\delta \sinh \alpha \cosh [(\sum_i x_i w_i)/\delta] e^{-A'/\delta}$  と評価できることから、十分大きな  $A'/\delta$  に対して次の関係が満たされる：

$$\int_{-A'}^{A'} \psi\left(\sum_{i=1}^n x_i w_i - w_0\right) e^{iw_0} dw_0 \sim \Psi(1) \exp\left(i \sum x_i w_i\right). \quad (4.2)$$

次に、(2.4) の  $w_0$  に関する積分に (4.2) の近似を適用しよう。すると、(2.4) 式で  $w_1, \dots, w_n$  に関する積分を有限和で置き換えるのに、フーリエ級数展開の部分和をとる方法が考えられる。

簡単の為、区間  $[-\ell, \ell]$  で周期条件を満たす 1 変数関数  $f(x)$  について、フーリエ級数の部分和の誤差を評価する。そのフーリエ級数の部分和を  $S_N(x)$  とすれば、誤差は、

$$f(x) - S_N(x) = \frac{1}{2\ell} \int_{-\ell}^{\ell} [f(x) - f(x-t)] D_N(t) dt, \quad (4.3)$$

で与えられる。ここで平均値の定理から、 $f(x) - f(x-t) = f'(x-\theta t)t$  を満たす  $\theta$  ( $0 < \theta < 1$ ) が存在する。ここで、 $t$  は  $x$  に関する微分を示す。これを (4.3) に適用し、部分積分を実行すれば、十分大きな  $n$  に対し、 $f(x) - S_N(x) \sim c/N$  と評価できる。ここで、

$$c = \frac{1}{2\pi} \int_{-\ell}^{\ell} \frac{d}{dt} \left[ \frac{f'(x-\theta t)t}{\sin \frac{\pi t}{2\ell}} \right] \cos \frac{\pi t}{\ell} \left( N + \frac{1}{2} \right) dt. \quad (4.4)$$

$I_{A',A}(\mathbf{x})$  の積分をフーリエ級数の部分和で近似した式を  $S_{N_1 \dots N_n}(\mathbf{x})$  とおいて、同様の評価をすれば十分大きな  $N_1 N_2 \dots N_n$  に対し、

$$f(\mathbf{x}) - S_{N_1 \dots N_n}(\mathbf{x}) \sim \frac{c_1}{N_1} + \frac{c_2}{N_2} + \dots + \frac{c_n}{N_n}, \quad (4.5)$$

$$c_1 = \frac{1}{2\pi} \int_{-\ell}^{\ell} \dots \int_{-\ell}^{\ell} \cos \frac{\pi t_1}{\ell} \left( N_1 + \frac{1}{2} \right) \times \frac{d}{dt_1} \left[ \frac{\partial f}{\partial x_1}(x_1 - \theta t_1, x_2, \dots, x_n) t_1 \left( 2 \sin \frac{\pi t_1}{2\ell} \right)^{-1} \right] dt_1, \quad (4.6)$$

⋮

$$c_n = \frac{1}{2\pi} \int_{-\ell}^{\ell} \dots \int_{-\ell}^{\ell} \cos \frac{\pi t_n}{\ell} \left( N_n + \frac{1}{2} \right) \times \frac{d}{dt_n} \left[ \frac{\partial f}{\partial x_n}(x_1, \dots, x_n - \theta t_n) t_n \left( 2 \sin \frac{\pi t_n}{2\ell} \right)^{-1} \right] dt_n. \quad (4.7)$$

と表せる。

誤差  $|f(\mathbf{x}) - \bar{f}(\mathbf{x})|$  は、(4.5) に (4.2) の左辺の  $w_0$  に関する積分を有限和にしたときの誤差を足したものである。従って 3 章の内容を適用すれば、(4.5) に  $O(1/N_0)$  の誤差を加えたものである。

## 5. 数値シミュレーション

3章で議論した内容を1変数と2変数の具体的な関数に応用する。そのため、船橋の式(2.1)を求めるアルゴリズムを書き、誤差と隠れユニット数の関係を計算した。

図1、図2は1変数、2変数関数についてのシミュレーションの結果を示している。それぞれグラフ中の実線は、4次の多項式及び、XORを表現するような連続関数を近似した時の誤差と隠れユニット数の関係を表している。また、点線は関数  $y = c/N$  によってフィットした曲線を示す。4次の多項式として用いた関数は、区間  $[0, 5.0]$  上で定義された  $f(x) = (x-1)(x-2)(x-3)(x-4)$  である。XORの方は、区間  $[0, 1.0]^2$  上で定義された関数、 $f(x, y) = (1 - \cos \pi x \cos \pi y)/2$  である。図1の方の縦軸は区間の中の10個の点での誤差の和を表している。また、図2の縦軸は、 $(x, y) = (0, 0)$ 、 $(0, 1)$ 、 $(1, 0)$ 、 $(1, 1)$  の4つの点での誤差の和を示している。多項式のほうは、同じ隠れユニット数に対し、4章で示した結果よりも良い精度が得られている。またXORの方は、ほぼ4章の結果と同じ精度が得られている。いずれも、 $N$ が大きくなるにつれて誤差は0に近づいているが、隠れユニット数を  $M$  とすれば、多項式の方では  $M = 2N^2$  となり、XORの方では  $M = 2N^3$  となるので、非常に多くの隠れユニットを必要とする。ただし4章の議論によれば、関数が周期条件を満たし、かつ、フーリエ級数展開が有限項で終わるとき、 $M \sim O(N)$  の隠れユニット数で誤差は  $\sim O(1/N)$  となる。

## 6. まとめ

本論文では、船橋の定理における誤差と隠れユニット数の間の関係を評価した。船橋の定理では、(2.4)の各積分を和に直す時の積分区間の分割数が直接隠れユニット数になる為に、隠れ層のユニット数は、各積分の積分区間を  $N$  で分割した時  $2N^k$  で増えていく。ここで  $k = (\text{独立変数の数}) + 1$  で、因子2は隠れ層と出力層間の結合として、絶対値が等しくかつ符号が逆の結合が存在することから生じる。我々の数値シミュレーション(図1、図2)では、高速フーリエ変換を用いる為、フーリエ変換するデータを2のべき乗個にしなければならない。1つの積分の分割数を  $N = 2^l$  とすれば、 $D$ 次元の関数の場合、それを表現するパーセプトロンの隠れユニット数は  $2^{l(D+1)+1}$  個必要とする。

Barton<sup>[4]</sup> は、目標パターンとネットワークの出力パターン間の2乗誤差を最小にするように、隠れ層と出力層の間の重みを入力層と隠れ層の間の重みの関数で表す式を提案している。ここでは隠れ層の出力値で構成される行列の逆行列を用いている為、その行列式が0でないという条件が付けられている。彼の数値シミュレーションでは、2乗誤差を最小にするような、入力層と隠れ層の間の最適な重みを求めている。その結果によると、 $d$ 次多項式の近似では隠れ層のユニット数が  $d$ 個で十分な精度が得られると報告されている。又、パリティ問題として12次元までのXOR問題を解き、これについても  $D$ 次元の場合、ただだか  $D$ 個の隠れユニットを用いればほぼ厳密な値を出力すると主張している。いずれの場合も、船橋の定理から推定されるよりもはるかに少ない隠れユニット数で十分な精度が実現されている。

これまで見てきたように船橋の定理は、閉区間上で定義された関数を厳密に表現する3層パーセプトロンの存在を示すものとして注目すべき結果であるが、実用上有効といえない。今後は、少ないユニット数で十分な精度を得る事が一般的に可能かどうか、可能とすればそのネットワークの具体的な構成について調べる必要がある。

## 参考文献

[1] Rumelhart, D. E. , Hinton, G. E., and Williams, R. J. 1986. Learning internal representations by error propagation. In *Parallel Distributed Processing*, Vol.1, D.E.Rumelhart, J.L.McClelland, and the PDP Research Group, eds., pp.318-362. MIT Press, Cambridge, MA.

また、以下の文献を参照のこと。

Bryson, A.E., and Ho, Y.C. 1969. *Applied Optimal Control*. New York: Blaisdell.

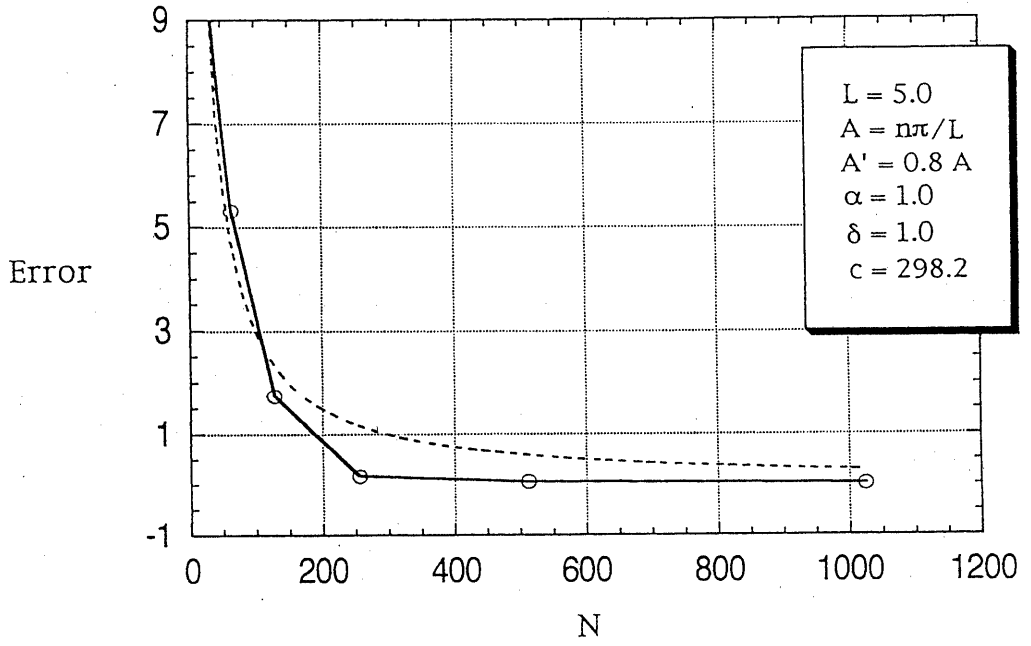
Werbos, P. 1974. Beyond Regression : New Tools for Prediction and Analysis in the Behavioral Science. Ph.D. Thesis, Harvard University.

Parker, D.B. 1985. Learning Logic. Technical Report TR - 47, Center for Computational Research in Economics and Management Science, Massachusetts Institute of Technology, Cambridge, MA.

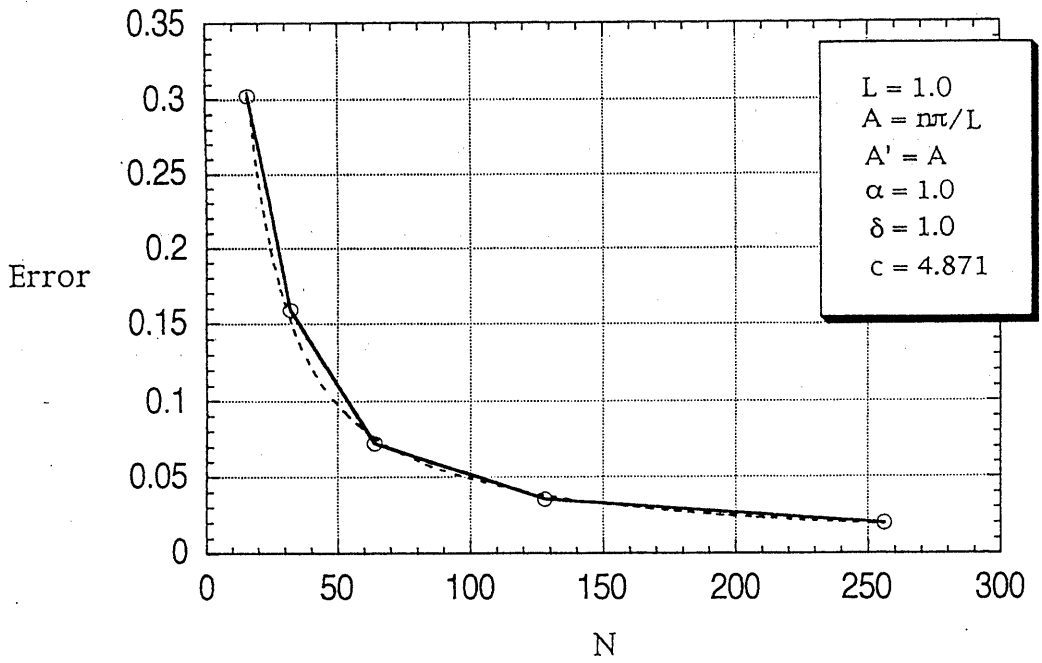
[2] Irie, B., and Miyake, S. 1988. Capabilities of three-layered perceptrons. *IEEE International Conference on Neural Networks*,1, 641-648.

[3] Funahashi, K. 1989. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2,183-192.

[4] Barton, S. 1991 A Matrix Method for Optimizing a Neural Network . *Neural Comp.*, 3 , 450-459.



⊗ 1



⊗ 2