統計数理（1996）
第44巻 第1号 85-98

特集「計算統計学の発展」
[研究詳解]

統計モデルとしてのニューラルネットワーク

北海道大学* 佐 藤 義 治

（1996年4月 受付）

1. はじめに

ニューラルネットワークを日本語でいうならば，神経回路網ということになろうが，これらに関する研究をみるとわが国ではこの両者にはいささか相違があるように思われる．神経回路網理論は脳の生理的機能をいかに数理的に表現できるのかという問題を扱っているのに対して，ニューラルネットワークはその実用的な側面に重点をおき，主に情報識別や種々の予測に関する問題を解決する一つの道具として論じられている．

本論文では実用的な側面を中心としたニューラルネットワークについてそれを利用する立場から考察する．近年ニューラルネットワークを応用した研究が数多く見られる．種々の識別・認識問題，たとえば，文字・形状認識，音声認識など，また自然言語処理(理解)やロボット制御などの工学的な応用のみならず，社会システムにおける様々なエキスパートシステムやデータベース検索，市場調査などへの応用がある．さらに地球科学の分野では，気象予測，資源探査，地震予知などへの応用研究が行なわれている．

これらほとんどすべての問題は従来から統計科学の枠内で議論されている．すなわち，種々の統計モデルを用いた判別・識別の理論，あるいは時系列解析・予測のための統計モデルはいうまでもない．しかし，ニューラルネットワークの立場からの意見では，多くの場合，統計モデルより“うまくいく”といわれている（Cheng and Titterington (1994)）．一般にニューラルネットワークは非線形モデルとみなすことができるが，これだけが理由とは思われない．統計モデルとの本質的な違いがあるのだろうか．この意味において本報告では，ニューラルネットワークを統計モデルという立場から考察することが目的である．

統計モデルという観点から見るとニューラルネットワークは広くは多変量非線形回帰モデルと見なすことができる．このとき問題となるのは標本数をnとするとき，推定すべきパラメータ数qは形式的にみると“$n \ll q$”となっている場合が多い．もしこれを許すならば識別等はいくらでも“うまくいく”ことになる．統計モデルにおけるAICの考え方からすると問題がある．

このような背景の下で，ニューラルネットワークにおいて問題となるのは，誤差を含むデータを扱うことができるのか否か，すなわち統計モデルでいうところの予測誤差の問題が扱えるのか否かということである．特に多層ニューラルネットワークにおいて隠れ素子の個数を減少させることに意味があるのか否かを検討する必要がある．またニューラルネットワークにおいて推定すべきパラメータと統計モデルにおけるパラメータ（母数）と同一視できるかどうかは詳細に考察してみる必要がある．

ニューラルネットワークを統計モデルとして考察しているものはそれほど見受けられない．しかし，ニューラルネットワークにoverfittingの問題があることは従来から指摘されている(Ripley (1994), Smith (1993) and Haykin (1994))．それを回避するためにいわれているこ

* 工学部：〒060 北海道札幌市北区北13条西8丁目．

とは，収束条件，すなわち2乗誤差の和で表される適合度を最適（最小）にせずに途中で打ち切るということである．しかしこれでは適合度は全く無意味なことになる．そこで本報告ではまず第2章で3層ネットワークだけを取り上げ，基本的な性質から検討をする．

第3章では多変量正規混合モデル，あるいは多変量正規密度関数をカーネル関数とする確率密度関数推定と密接な関連があると思われるRBF（Radial Basis Function）ネットワークについて，その理論的背景も含めて論ずる．これらのモデルにより，ニューラルネットワークに期待する“自己組織的”とか“知的処理”ということが実現できるかどうかを明らかにしたい．

2. ニューラルネットワークモデル

多数の素子が互いに適当な重さをもって結合されたものをニューラルネットワークという．その結合のしかたによって大きくつぎの二つに分類される．その一つは相互結合型ニューラルネットワークと呼ばれるもので（図1），各素子がすべて相互に結合されたものであり，ホップフィールドネットワークやボルツマンマシンとして知られているものが代表的な例である．これに対して，素子がいくつかの層をなして配列され，各層間の素子は互いに結合されているが，層内の素子は結合されていない，階層型ニューラルネットワーク（図2）と呼ばれるものが知られている．ただし，このとき入力層と呼ばれる特殊な層をもち，この層は単にデータを入力するだけの機能を有するものである．入力層と出力を担う素子から成る出力層にはさまれている層は中間層あるいは隠れ素子層と呼ばれ，一般には多層から成る．

2.1 パーセプトロンと階層型ニューラルネットワーク

階層型ニューラルネットワークとして古くから知られているものとしてパーセプトロンがある．パーセプトロンの構成は基本的には図2と同一であるが，各素子の特性がマカロック－ピッツモデルに従うものであり，1または0の2値の入力データ $x_1, x_2, \ldots, x_p$ に対して，中間層の出力 $h_j (j = 1,2,\ldots,r)$ は

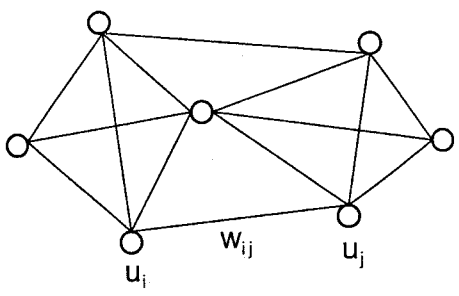

図1．相互結合型ニューラルネットワーク．

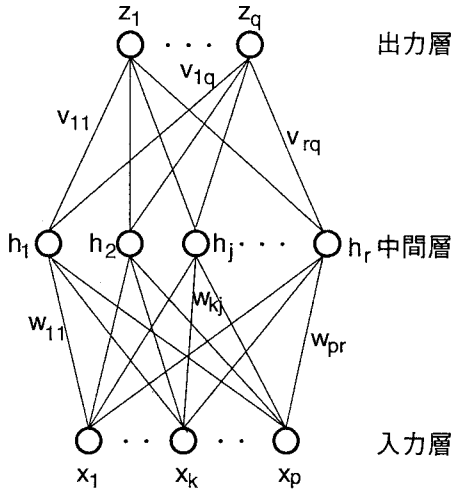

図2．階層型ニューラルネットワーク．

$$h_j = \begin{cases} 1, & \text{if } \sum_{k=1}^{p} w_{kj} x_k + w_{0j} > 0, \\ 0, & \text{if } \sum_{k=1}^{p} w_{kj} x_k + w_{0j} \leq 0 \end{cases} \tag{2.1}$$

として与えられ，出力層での出力 z_a $(a = 1, 2, \ldots, q)$ は

$$z_a = \begin{cases} 1, & \text{if } \sum_{l=1}^{r} v_{la} h_l + v_{0a} > 0, \\ 0, & \text{if } \sum_{l=1}^{r} v_{la} h_l + v_{0a} \leq 0 \end{cases} \tag{2.2}$$

これに対して，現在多く用いられている階層型ニューラルネットワークは素子の特性がロジスティックモデルによるものであり，その構成はパーセプトロンと同様である．したがって［0,1］上の連続値をとる入力変数 $x_1, x_2, \ldots, x_p$ に対して，中間層の j 番目の素子の出力 h_j $(j = 1,2,\ldots,r)$ は

$$h_j = \xi(x_1, \ldots, x_p) = 1 \Big/ \left\{ 1 + \exp\left(-w_{0j} - \sum_{k=1}^{p} w_{kj} x_k \right) \right\} \tag{2.3}$$

であり，出力層での a 番目の素子の出力 z_a $(a = 1, 2, \ldots, q)$ は

$$z_a = \eta(h_1, \ldots, h_r) = 1 \Big/ \left\{ 1 + \exp\left(-v_{0a} - \sum_{m=1}^{r} v_{ma} h_a \right) \right\} \tag{2.4}$$

として与えられる．このとき入力空間 $I_p = \{x_1, x_2, \ldots, x_p\}$ は単位区間［0,1］の p 次元デカルト積，つまり p 次元単位立方体であり，また中間層の出力からなる空間 H_r および出力層の出力からなる空間 Z_q もそれぞれ r 次元，q 次元単位立方体である．このとき（2.3），（2.4）は

$$I_p \xrightarrow{\xi(x)} H_r \xrightarrow{\eta(h)} Z_q$$

写像(関数)を意味する．ここで，$x = (x_1, x_2, \ldots, x_p)$，$h = (h_1, h_2, \ldots, h_r)$ とした．

階層ニューラルネットワークは一方では多層ニューラルネットワークとも呼ばれ，ネットワークモデルを構成するためには中間層を何層にすべきかが重要な課題であるが，ある条件の下では三層ニューラルネットワークで十分であることがつぎの定理で示されている．

定理. ［関数近似の基本定理］(Cybenko (1989), Funahashi (1989), Hornik et al. (1989)) $\varphi(\cdot)$ を有界でかつ単調増加（非定数）関数とし，I_p を p 次元単位立方体とするとき，I_p の連続関数からなる空間を $C(I_p)$ と表す．このとき，$\forall f \in C(I_p)$ および $\forall \varepsilon > 0$ に対して，ある整数 M と実数 α_i, θ_i, w_{ij} $(i = 1, \ldots, M; j = 1, \ldots, p)$ が存在して，

$$F(x_1, \ldots, x_p) = \sum_{i=1}^{M} \alpha_i \varphi\left(\sum_{j=1}^{p} w_{ij} x_j - \theta_i \right)$$

なる関数により

$$|F(x_1, \ldots, x_p) - f(x_1, \ldots, x_p)| < \varepsilon$$

がすべての $\{x_1, \ldots, x_p\} \in I_p$ に対して成り立つ．

表1．学習データ．

関数値（出力）				変数			
y_1	y_2	$\cdots$	y_q	x_1	x_2	$\cdots$	x_p
y_{11}	y_{12}	$\cdots$	y_{1q}	x_{11}	x_{12}	$\cdots$	x_{1p}
y_{21}	y_{22}	$\cdots$	y_{2q}	x_{21}	x_{22}	$\cdots$	x_{2p}
$\vdots$	$\vdots$		$\vdots$	$\vdots$	$\vdots$		$\vdots$
y_{n1}	y_{n2}	$\cdots$	y_{nq}	x_{n1}	x_{n2}	$\cdots$	x_{np}

2.2 階層型ニューラルネットワークと非線形回帰モデル

表1で与えられるデータに対して $(x_1, x_2,..., x_p)$ から $(y_1, y_2,..., y_q)$ を予測する問題は多変量重回帰分析として知られている．そのモデルは一般的に，

$$\begin{cases} y_1 = f_1(x_1, x_2,..., x_p)+\varepsilon_1 \\ y_2 = f_2(x_1, x_2,..., x_p)+\varepsilon_2 \\ \quad \vdots \\ y_q = f_q(x_1, x_2,..., x_p)+\varepsilon_q \end{cases} \tag{2.5}$$

として表される．もし f_i が線形関数ならばよく知られているようにモデルはつぎのようになる．

$$y_a = \beta_{0a}+\beta_{1a}x_1+\cdots+\beta_{pa}x_p+\varepsilon_a \ (a = 1,2,...,q).$$

階層型ニューラルネットワークにおいて，出力層の各素子の出力 z_a による関数値 y_a からの残差を ε_a とおくとモデル (2.3), (2.4) は

$$y_a = 1\Big/\Big\{1+\exp\Big\{-v_{0a}-\sum_{l=1}^{r}\Big\{1\Big/\Big\{1+\exp\Big(-w_{0l}-\sum_{k=1}^{p}w_{kl}x_k\Big)\Big\}\Big\}\Big\}\Big\}+\varepsilon_a \tag{2.6}$$

と表わされ，これは (3.8) の特別な場合である．階層型ニューラルネットワークモデルを回帰モデルとして捉えるならば，統計的モデルとして古くから論じられている回帰モデルの諸性質を継承していることになる．回帰および判別のいずれの場合においても，それらを利用する主要な目的は与えられた学習データに基づき推定して回帰式を用いて未知の（未学習の）データに対する予測を行なうことである．このとき予測誤差をできる限り小さくすることが重要であることはいうまでもない．そのためには AIC の考え方にあるように，モデルに含まれる未知パラメータを増加して当てはまりだけを良くすることは意味がなく，それは予測誤差を増加させることになることは統計的モデルの考え方として良く知られていることである．しかし階層型ニューラルネットワークでは単に当てはまりの良さのみで“うまくいく”といっているように思われる．例えば英数字活字文字認識の問題では，76文字を判別するために，各文字10標本を用いて，入力層の素子数（入力変数の個数）を99，中間層の素子数を20～80，出力層の素子数7によってネットワークを構成している．この場合モデルに含まれる未知パラメータの個数は $99\times20\times7+27 = 13887$ であり，学習データが高々1000個に対して約1万個以上のパラメータで当てはめを行なうならば，学習データはほとんど誤差なく当てはまるのは当然である．また7文字の単語の音声合成の問題においては，入力層の素子数 $29\times7 = 203$，中間層の素子数80，出力層の素子数26なるネットワークが用いられている（桐谷（1989）参照）．この場合には推定すべき未知パラメータ数は $203\times80\times20+100 = 324900$ である．入力変数の個数は203個と大きいにしても，あまりにも未知

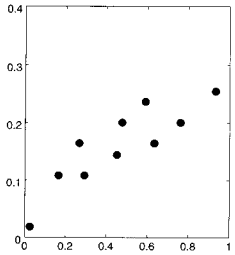

図3．学習データ．

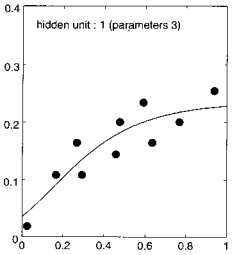

図4(a)．中間層の素子数1．

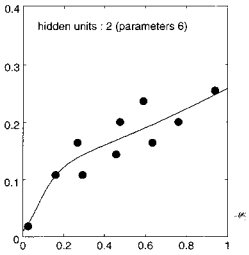

図4(b)．中間層の素子数2．

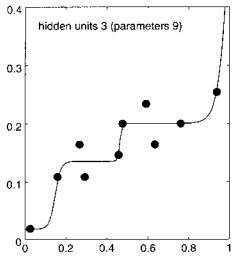

図4(c)．中間層の素子数3．

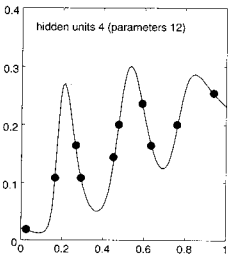

図4(d)．中間層の素子数4．

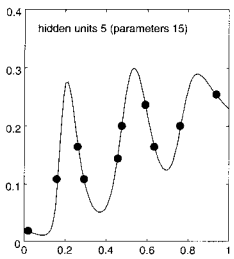

図4(e)．中間層の素子数5．

パラメータが多く，いわゆる過学習（overfitting）であることは明らかである．

階層型ニューラルネットワークにおける未知パラメータの役割が特殊なものであるとは考えられず，これを検証するために最も単純な例を用いて実験を行なった結果を以下に示す．データはいくつかの都道府県の民力指数（x）と教育指数（y）を観測し，図3にプロットしたものである．これを階層的ニューラルネットワークを用いてxからyを予測することを試みる．この場合入力層および出力層の素子数はいずれも1個である．これに対し，中間層の素子数を1から5まで変化させたときに得られる非線形回帰式を図4(a)〜(e)に示した．これはまさに多項式回帰における次数に関する問題と全く同様であることがわかる．したがって階層的ニューラルネットワークを予測問題に適用する場合にはこの点を十分に考慮しなければ，単に学習データに対してうまく当てはまっても何の意味もないこととなる．この点に関して，冒頭に述べたように中間層の素子数を十分大きくして，収束を途中で打ち切る方法が提案されているが，この主旨はできる限り線形に近い関数で近似(予測)しようということである．そうならば線形モデルを用いればよいことになり，ニューラルネットワークを用いる積極的理由にはならない．

3. RBF（Radial Basis Function）ネットワーク

3.1 RBFネットワーク

非線形判別問題を扱うことを目的としたニューラルネットワークの1つとしてRBFネットワークが提案されている．(Broomhead and Lowe (1988), Lowe (1991)) 前章でも説明したように，この判別問題はFisherの判別関数のように扱うならば非線形回帰あるいは非線形関数の近似理論と考えることもできる．このネットワークの基本的な考え方は，非線形判別問題におけるつぎのφ-分離性（φ-separability）である．簡単のために，R^pにおけるN個のデータ$X=\{x_1, x_2,..., x_N\}$が2群X^+とX^-から構成されているものとしよう．このとき，$\exists w \in R^p$であり，

$$w'x \geq 0 \quad \text{for } x \in X^+$$

$$w'x < 0 \quad \text{for } x \in X^-$$

なるとき X は線形分離可能であるという．また R^p から R^m への変換 φ を

$$\varphi(x) = \{\varphi_1(x), \varphi_2(x), \ldots, \varphi_m(x)\} \in R^m$$

とする．このとき

$$w'\varphi(x) \geq 0 \text{ for } x \in X^+$$

$$w'\varphi(x) < 0 \text{ for } x \in X^-$$

を満たす $w \in R^m$ が存在するとき，X は φ 分離性（φ-separability)をもつという．

この考え方に基づき，RBFネットワークは，図5に示すように，入力層と1層からなる中間層および出力層の3層から構成され，学習データ $\{x_i \in R^p | i = 1, 2, \ldots, N\}$ に対して関数 $F(x)$ をつぎのように与える．

$$F(x) = \sum_{i=1}^{N} w_i \varphi(x - x_i) \tag{3.1}$$

このとき $x_i \in R^p$ をRBFの中心という．中間層は N 個の素子から成りその活性化関数を $\varphi(x-x_i)$ とする．したがって入力層から中間層へは直接入力され，中間層から出力層へは重み係数 w_i による線形結合として関数値が得られる．すなわち，RBF では φ-separabilityを実現しようとするものである．

学習データとして与えられる望ましい出力を d_i とし，$\varphi_{ji} = \varphi(\|x_j - x_i\|)$ とおき，さらに $d = [d_1, \ldots, d_N]'$，$w = [w_1, \ldots, w_N]$ および $\Phi = [\varphi_{ji}]$ とおくと学習データに対して上式はつぎのようになる．

$$\Phi w = d$$

もし Φ が正則ならば，

$$w = \Phi^{-1} d$$

として，重みベクトル w を求めることができる．RBF のあるクラスで Φ が正定値であるものが存在する．これを満たすものとしてはつぎのような関数が知られている．

(1) Inverse multiquadratic

$$\varphi(r) = \frac{1}{(r^2 + c^2)^{1/2}}, \quad c > 0, \ r \geq 0$$

(2) Gaussian function

$$\varphi(r) = \exp\left\{-\frac{r^2}{2\sigma^2}\right\}, \quad \sigma > 0, \ r \geq 0$$

Φ が正則でないならばつぎにのべる regularization theory によって

$$\Phi = \Phi + \lambda I$$

として w を求める．

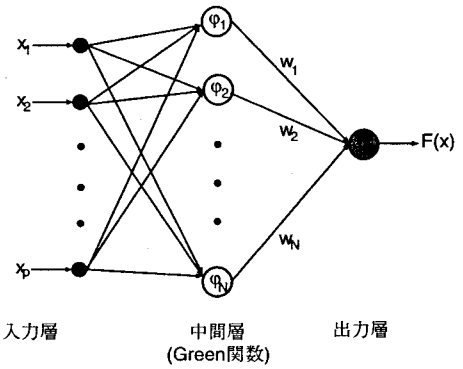

図5．Reguralization ネットワーク．

3.2 Regularization theory

写像 $F: X \to Y$ がつぎの条件を満たすとき well-posed であるという．

1．存在： $\forall x \in X$ に対して $y = F(x)$ となる $y \in Y$ が存在する
2．一意性： $\forall x, t \in X, F(x) = F(t)$ iff $x = t$
3．連続性： $\forall \varepsilon > 0, \exists \delta = \delta(\varepsilon), \rho_x(x, t) < \delta \Rightarrow \rho_y(F(x), F(t)) < \varepsilon$

この条件を満たさないとき，ill-posedと呼ばれる．前述のRBFネットワークにおいては中間層の素子数と学習データの個数が同一であり，この条件では overfitting を生じ，また ill-posed になりやすい．また学習データに入出力関係を一意に定める情報は含まれていないことと，データに含まれる雑音などにより，well-posed な関数を構成するためにはある冗長性が必要である．冗長性をもつ関数の基本的な形が滑らかさ（smoothness）である．

Tikhonov and Arsenin（1977）は ill-posed な問題を解決するための方法として Regularization 理論を提唱した．基本的な考え方は事前情報として非負の補助関数（汎関数）を導入し，解の安定化（stabilize）を計ることである．すなわち，入出力関数に滑らかさの制約を付加することによって ill-posed な問題を well-posed な問題とすることである．

入出力関数を $F(x)$ とし，入力データ $x_i \in R^p$，$(i = 1,2,..., N)$ に対する関数値を $y_i = F(x_i)$ とし，観測される出力値を d_i とする．このとき Tikhonov の regularization は F に関するあるコスト汎関数を最小にすることである．コスト関数はつぎの2つの項から成る．

(1) Standard error term：

$$\mathcal{E}_S(F) = \frac{1}{2}\sum_{i=1}^{N}(d_i - y_i)^2 = \frac{1}{2}\sum_{i=1}^{N}(d_i - F(x_i))^2$$

(2) Regularization term：

$$\mathcal{E}_C(F) = \frac{1}{2}\|PF^2\|$$

ここに P は線形（擬）微分作用素である．P は問題に応じて決められるものであり，F に関する事前情報として導入されるもので，安定化作用素と呼ばれるものである．記号 $\|\cdot\|$ は PF の含まれる関数空間でのノルムである．通常ここで用いられる関数空間は L^2- 空間とする．結局 regularization では

$$\mathcal{E}(F) = \mathcal{E}_S(F) + \lambda \mathcal{E}_C(F) \tag{3.2}$$

を最小にする関数Fを求めることになる．このとき$\lambda \in (0, \infty)$は regularization parameterと呼ばれる．

1次元データで考えると，微分作用素Pはつぎのように定義される．

$$\|PF\| = \int_{R^1} \left[\frac{d^2F(x)}{dx^2} \right]^2 dx$$

この場合，コスト関数$\mathcal{E}(F)$を最小にする$F(x)$は cubic spline である．

この最適化問題を解くためには汎関数$\mathcal{E}(F)$の微分を考えなければならない．そのためにここではつぎのようなFréchet 微分を用いる．

$$\mathcal{E}(F, h) = \left[\frac{d}{d\beta} \mathcal{E}(f + \beta h) \right]_{\beta = 0}$$

ここに$h(x)$はある固定されたxの関数である．コスト関数を最小にするための必要条件は，すべての$h(x)$に対して$F(x)$に関するFréchet 微分が0となることである．

$$d\mathcal{E}(F, h) = d\mathcal{E}_S(F, h) + \lambda d\mathcal{E}_C(F, h) = 0$$

微分作用素Pの共役作用素をP^*とおくと上式は Euler-Lagrange の方程式

$$P^*PF = \frac{1}{\lambda} \sum_{i=1}^{N} (d_i - F)\delta(x - x_i) \tag{3.3}$$

と表され，この微分方程式の解は自共役な微分作用素P^*Pに対する Green 関数によって与えられる kernel の積分変換となっている．$G(x; x_i)$を中心x_iをもつ Green 関数とする．したがって，上記の微分方程式の解$F(x)$は，

$$\varphi(\xi) = \frac{1}{\lambda} \sum_{i=1}^{N} (d_i - F)\delta(\xi - x_i)$$

とおくと

$$F(x) = \int_{R^p} G(x; \xi)\varphi(\xi) d\xi$$

と表される．上記の和と積分の順序を交換し，デルタ関数の性質（sifting property）を用いると

$$F(x) = \frac{1}{\lambda} \sum_{i=1}^{N} \{d_i - F(x_i)\} G(x; x_i)$$

が得られる．

つぎに未知の係数を決定しよう．

$$w_i = \frac{1}{\lambda} \{d_i - F(x_i)\}$$

とおくと

$$F(x) = \sum_{i=1}^{N} w_i G(x; x_i)$$

であり，$x = x_j$ においては

$$F(x_j) = \sum_{i=1}^{N} w_i G(x_j; x_i), \quad j = 1,..., N$$

であるから，

$$F = [F(x_1),..., F(x_N)]', \ d = [d_1,...,d_N]', \ w = [w_1,...,w_N]', \ G = [G(x_j; x_i)]$$

とおくと

$$w = \frac{1}{\lambda}[d-F], \quad F = Gw$$

と表される．したがって，w はつぎの方程式を解くことによって得られる．

$$(G+\lambda I)w = d.$$

P^*P は自己共役作用素であるから，それに伴う Green 関数は対称である．したがって $G(x_i; x_j) = G(x_j; x_i)$ であり上記行列 G は対称行列となる．また Green 関数のあるクラスでは $x_1,..., x_N$ がすべて異なるならば行列 G は正定値となる．たとえば前述の inverse multiquadratic 関数や Gaussian 関数などがそのクラスに含まれる．実際十分大きな λ をとると $G+\lambda I$ は正則となり，

$$w = (G+\lambda I)^{-1}d$$

として w を求めることができる．結果として，Regularization 問題の解は

$$F(x) = \sum_{i=1}^{N} w_i G(x; x_i) \tag{3.4}$$

として与えられる．

以上の Regularization によるネットワークについてまとめると以下のようになる．

1．Regularization approach は Green 関数の族での展開に等しい．またそれは stabilizer P の形や境界条件のみによって特性づけられる．
2．Green 関数の個数は学習データの個数に等しい．
3．解の一意性については $Pg = 0$ となる，すなわち作用素 P に関する Null space に属する関数を除いてということである．しかし，Gaussian や Inverse multiquadratic（bell-shape Green functions）ではその必要がないので扱いやすい．

3.3 多変量正規密度関数

回転および平行移動に関して不変な微分作用素 P は

$$\|PF\| = \sum_{k=0}^{K} a_k \|D^k F(x)\|^2 \quad \text{(multi-index)}$$

の形で表される．微分作用素 D^k のノルムはつぎのように定義される．

$$\|D^k F\|^2 = \sum_{|\alpha|=k} \int_{R^p} |\partial^\alpha F(x)|^2 dx, \quad |\alpha| = \alpha_1+\alpha_2+\cdots+\alpha_p = k$$

Green 関数 $G(x; x_i)$ はこの微分作用素について

$$\sum_{k=0}^{K}(-1)^k a_k \nabla^{2k} G(x\,;x_i)=\delta(x-x_i)$$

を満たす，ここに ∇^{2k} は k 回の p 次元作用素を意味する．

$$\nabla^2=\frac{\partial^2}{\partial x_1^2}+\cdots+\frac{\partial^2}{\partial x_p^2}$$

Gaussian radial basis function の場合には上記和の上限 k を $+\infty$ に，さらに係数 a_k を

$$a_k=\frac{\sigma_i^{2k}}{k!2^k}$$

とおくことになる．ただし σ_i は定数とする．このとき Green 関数 $G(x\,;x_i)$ はつぎの微分方程式を満たす．

$$\sum_{k=0}^{\infty}(-1)^k\frac{\sigma_i^{2k}}{k!2^k}\nabla^{2k}G(x\,;x_i)=\delta(x-x_i)$$

多次元のフーリエ変換を用いてこの微分方程式を解くと Green 関数 $G(x\,;x_i)$ は

$$G(x\,;x_i)=\exp\left(-\frac{1}{2\sigma_i^2}\|x-x_i\|^2\right)$$

として得られる．このように表される Green 関数 $G(x\,;x_i)$ を多変数 Gaussian 関数という．したがって

$$F(x)=\sum_{i=1}^{N}w_i\exp\left(-\frac{1}{2\sigma_i^2}\|x-x_i\|^2\right)$$

が得られる．

図5に示されるように，中間層の素子の活性化関数が Green 関数で表されるネットワークを Regularization ネットワークと呼ぶ．このネットワークを近似理論の立場からみるとつぎのような特徴がある．

1．Regularization ネットワークは中間層の素子数を十分大きくとるならば，R^p のコンパクト部分集合上の任意の関数を必要な精度で近似することができる．
2．このネットワークにおいては，未知のパラメータに関しては線形となっていることから，その推定量としては最良のものを容易に得ることができる．

前述の Reguralization ネットワークにおいては，入力される学習データの個数を N とするとき，中間層における Green 関数の個数も N 個であり，w を推定するためには $N\times N$ 行列の逆行列が必要である．また N が大きくなるのにつれて計算量が増大するとともに，行列が ill-condition となる可能性も増大する．そこで中間層の関数の個数(素子数)を N より小さくした M とし，

$$F^*(x)=\sum_{i=1}^{M}w_i\varphi_i(x)$$

によって近似関数を求める．RBF の考え方から，

$$\varphi_i(x)=G(\|x-t_i\|)$$

とおく，ただし中心 t_i は未知のものとする．このとき $F^*(x)$ を求める基準は

$$\mathcal{E}(F^*) = \sum_{i=1}^{N} \left(d_i - \sum_{j=1}^{M} w_j G(\|x_i - t_j\|) \right)^2 + \lambda \|PF^*\|^2 \tag{3.5}$$

として与えられる．

$\mathcal{E}(F^*)$ を最小にする解によって得られるネットワークを一般化RBFネットワークという．このネットワークにおいては出力素子においてバイアスを調整することが必要である．これはネットワークの出力層へ線形素子を1個つけ加えることによって解決することができる．従来のRBFネットワークと一般化RBFネットワークはその構成としては基本的には同じであるが，つぎの諸点においてその相違を示すことができる．

1．RBFネットワークでは中間層の素子数が学習データの個数 N であるのに対して，一般化RBFネットワークにおいては N より少ない個数 M で構成される．
2．一般化RBFネットワークでは，中間層のGreen関数の中心および線形重み w_i はいずれも未知パラメータとして扱われるが，従来のRBFネットワークでは中間層のGreen関数の中心は学習データによって与えられ，既知として扱われる．

3.4 階層的ネットワーク（MLN）との比較

RBFネットワークも階層的ネットワーク（MLP）もいずれも非線形階層的フィードフォーワードネットワークであり，同様の関数近似能力をもつものである．しかしながらこれら2つのネットワークの本質的な相違点はつぎのところにある．

1．RBFネットワークにおいては中間層は常に1層であるが，階層的ネットワークは一般には多層をもつ．
2．中間層の素子の特性について，通常の階層的ネットワークでは各素子の活性化関数は共通であるが，RBFネットワークでは各素子の活性化関数が異なるGreen関数で表され，出力層に対する役割が異なる．
3．RBFネットワークの中間層は非線形であるが，出力層は線形である．これに対し，MLNは一般に中間層，出力層ともに非線形である．しかし非線形回帰の問題を扱う場合には出力層は線形で十分である．
4．中間層の素子の活性化関数において，RBFネットワークでは入力ベクトルと中心ベクトルのユークリッド距離を計算するのに対して，MLNでは入力ベクトルと結合重みとの内積を計算している．
5．RBFネットワークで用いられる関数は局所的な非線形性が指数的に減少する（Gaussian関数など）ため，入出力の非線形な関係を局所的に近似していることになる．また学習の時間（計算量）や学習データに対する敏感性はMLNに比べて少ないが，RBFを用いて滑らかな関数を得るためには中間層の素子数を十分多くとることが必要である．

RBFネットワークにおけるGreen関数として多変量正規分布の確率密度関数を含むことからこのネットワークは統計モデルとしては正規混合モデルあるいは正規密度関数をカーネルとして用いた場合の密度関数推定と密接な関連をもっている．RBFネットワークにおいては

$$F(x) = \sum_{i=1}^{N} w_i G(x; x_i)$$

の形をしていることから，いま S が n 個の母集団 $S_1, S_2, \ldots, S_n$ の混合であり，その混合比を

$\pi_1, \pi_2, \ldots, \pi_n$, すなわち

$$\sum_{i=1}^{n} \pi_i = 1, \quad \pi_i \geq 0$$

とし, S_i の確率密度関数を $f_i(x; \theta_i)$ とするとき S の確率密度関数は

$$f(x; \varphi) = \sum_{i=1}^{n} \pi_i f_i(x; \theta_i)$$

と表される. 一方, RBF の形は見方をかえるとデータ $x_1, x_2, \ldots, x_N$ が与えられたとき, それに基づく確率密度関数の推定としてカーネル関数法を用いた場合も同様な形

$$f(x) = \sum_{i=1}^{N} \alpha_i G(x; x_i)$$

で与えられる.

3.5 RBFネットワークの学習

RBF ネットワークの学習において本質的なことは, Green 関数の中心を決定することである. 中心の決め方としてはつぎの 3 通りのものが考えられている.

1. ランダムな選択
   N 個の学習データからランダムに M 個の中心を選択する. このとき Green 関数それ自体は同型の Gaussian 関数を仮定すると中心 t_i をもつ関数は

   $$G(\|x-t_i\|^2) = \exp\left\{\frac{M}{d^2}\|x-t_i\|^2\right\}, \quad i = 1,2,\ldots,M$$

   となる. ここに M は中心の個数であり, d は選択された中心間の最大距離である. このとき Gaussian 関数の標準偏差を

   $$\sigma = \frac{d}{\sqrt{2M}}$$

   と固定したことになる. この σ の値は近似関数がデータ点で尖りすぎず, また平坦すぎないような値であるといわれている.
2. k- 最近隣法による中心の選択
   k- 最近隣の情報に基づき, 最も密度の高い点から M 個を選択する. 出力層の線形重みは最小 2 乗法によって決定する. (hybrid learning process)
3. 最小 2 乗法による推定
   中心の座標 (ベクトル) も未知パラメータとする. したがって, この場合, 全体のパラメータは

   $$w_i, \quad t_i, \quad \Sigma_i^{-1}$$

   となる, ここに Σ_i^{-1} は Gaussian 関数の共分散行列に相当する. これらの未知パラメータを交互最小 2 乗法によって求める.

4. 結論

本論文ではFeed-Forward型ニューラルネットワークとして代表的な階層型ニューラルネットワークとして多層ニューラルネットワークおよびRBF（Radial Basis Function）ネットワークを取り上げ，それらと統計モデルとの関連について論じた．

ニューラルネットワークの根本的な考え方は，人間の脳における情報処理の方式を従来の計算機に取り入れ，それを何らかの形でシミュレートすることにより，計算機の処理能力を向上させようというものであろう．これは単にニューラルネットワークだけではなく，遺伝アルゴリズムや人工生命等の考え方にも見られる．しかし問題となるのは，これらの生物機能を従来の計算機上に実現するためには何らかのアルゴリズムで表現する必要がある．しかしながら，これらのアルゴリズム表現が，従来の方法論の枠内に留まるならば，結局意味のないことになる．勿論すべてではないが，ニューラルネットワークに関するおおくのアルゴリズムは，従来の方法論を越えているとは思えない．特に階層型ニューラルネットワークにおいては，人間の脳のようにニューロンの個数は無限に多く存在するという考え方に立ったアルゴリズムを実現するならば，単に未知パラメータの個数を増加させただけのモデルになりかねない．しかしExclusive-Or問題に見られるように，明らかに非線形な境界をもつ判別領域などの探索には，中間層への適当な変換をうまく利用することが考えられるが，この点はさらに検討する必要があろう．

謝　辞

本稿をまとめるにあたり，貴重なご指摘ならびにご助言をいただきました査読者の諸氏には心からお礼申し上げます．また編集委員の方には種々のご指導ご高配をいただきました．ここに記して謝意を表します．

参　考　文　献

Broomhead, D.S. and Lowe, D. (1988). Multivariable functional interpolation and adaptive networks, *Complex Systems*, **2**, 321-355.

Cheng, Bing and Titterington, D.M. (1994). A small selection of neural network methods and their statistical connections, *Advances in Applied Statistics*, Supplement to Journal of Applied Statistics, **21**, 9-37.

Cybenko, G. (1989). Approximation by superpositions of sigmoidal function, *Math. Control Signals Systems*, **2**, 303-314.

Funahashi, K. (1989). On the approximate realization of continuous mapping by neural networks, *Neural Networks*, **2**, 183-192.

Haykin, Simon (1994). *Neural Networks*, Macmillan College Publishing, New York.

Hornik, K., Stinchcombe, M. and White, H.(1989). Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks, *Neural Networks*, **3**, 551-560.

Lowe, D. (1991). On the iterative inversion of RBF networks: A statistical interpretation, *Second IEE International Conference on Artificial Neural Networks*, Conference Publication 349, 29-33, Institute of Electrical Engineering, Bournemouth, U.K.

Ripley, B.D. (1994). Neural networks and flexible regression and discrimination, *Advances in Applied Statistics*, Supplement to Journal of Applied Statistics, **21**, 39-57.

Smith, M. (1993). *Neural Networks for Statistical Modeling*, Van Nostrand Reinhold, New York.

Tikhonov, A.N. and Arsenin, V.Y. (1977). *Solutions of Ill-Posed Problems*, Winston and Sons, Washington DC.

桐谷　滋 (1989).『ニューロコンピュータ』(中野　馨 監修), 技術評論社, 東京.

On Artificial Neural Networks as a Statistical Model

Yoshiharu Sato
(Division of Systems and Information Engineering, Hokkaido University)

As a typical Model of artificial neural networks (commonly referred as a neural networks), we discuss multilayer networks and recurrent networks. The first object of this paper is to investigate the learning algorithms and the second is a consideration of these networks from a statistical model.

Researches on the neural networks have been originally motivated by the recognition process of the brain. It seems to have highly capability than the conventional digital computers. So if we simulate the computing process of the brain on the conventional computer, then we will gain more efficient computational ability. But the problem is how to implement it on the digital computer as a concrete algorithm. Most of the algorithms using neural networks are reduced to the conventional algorithms in statistical data analysis. Therefore, it seems that these algorithms of neural networks do not offer a completely new idea for the statistical data anlaysis.

Key words: Neural networks, statistical data analysis, nonlinear regression, nonlinear discriminant function, radial basis function.