



ニューラルネットワークの化学パターン認識への応用

メタデータ	言語: jpn 出版者: 公開日: 2013-12-03 キーワード (Ja): キーワード (En): 作成者: 大西, 章 メールアドレス: 所属:
URL	https://doi.org/10.24729/00007812

ニューラルネットワークの 化学パターン認識への応用

大西 章*

An Application of Neural Networks to the Pattern Recognition Problem in Chemistry

Akira OHNISHI*

要 旨

ニューラルネットワークの化学分野への応用について、身近な化学パターン認識問題を例にその有用性を検討した。本稿で取り上げた例題は考古学的出土品に含まれる10種類の微量金属元素量をもとに4ヶ所の産地を判定する問題であるが、簡単な階層ネットワークモデルによって良好にクラス分類ができることを示した。また、学習計算への非線形最適化手法の応用についても検討した。

キーワード：ニューラルネットワーク 化学パターン認識 最適化

1. はじめに

近年の情報処理理論・技術の発展と、あらゆる分野におけるそれらの応用にはめざましいものがある。化学分野の場合、電子計算機の実際の応用は1970年代ではマクロな立場からの解析や合成を主とするプロセス系部門が中心であった。1980年代にはいと分子動力学法、コンピュータグラフィックス、構造活性相関分析法等の応用による物質設計法が急速に発達し、最近では、化学の分野においても知識情報処理理論の多方面での応用について、その可能性が期待されている。

電子計算機の化学分野への高度応用は今後ますます進展すると考えられるが、このような分野で研究や技術開発に携わる者にとっては化学のみならず、数学や情報科学についての深い知識が要求される。ところで、従来、高等専門学校での化学系学科における情報処理教育の内容はプログラミング言語教育が中心であり、卒業研究においても計算機は便利な道具として認識、使用されてきた。しかし、上記のような技術進歩に対応し、これから計算機化学分野を目指す学生には、特定の情報処理手法についての深い理解を要求することは必要でないとしても、適切な具体例を通じて情報処理理論・技術の化学分野への応用についての可能性とそれに必要な数学、物理等の化学以外の基礎の

重要性を認識させる教育が重要であると考えられる。

しかしながら、最近のシミュレーション技法、知的情報処理理論、システム制御論等の情報処理分野の内容は化学系学生にはなじみにくく、また、入門者向けの解説やプログラム例もあまり見あたらないのが現状である。そこで、本稿では化学系学生対象の一教材資料として知識情報処理のうちでも生化学物質の構造活性相関解析、分析機器データの自動分析、蛋白質の構造予測、複雑なプラントの高度自動化や異常診断等への応用が期待されているニューラルネットワーク手法について身近な化学パターン認識問題例の検討を通してその初歩を解説し、あわせて著者の作成した FORTRAN言語によるニューラルネットシミュレータについて紹介する。

2. ニューラルネットワーク

2.1 ニューロンモデル

神経細胞の人工的なモデルをユニットと呼び、生理学的知見に基づいて現在一般的に用いられているユニットを図1に示す。ユニットには複数の入力 X_i と1本の出力 Y があり、入力は結合係数 W_i によって荷重加算され、応答関数 $f(\sum W_i X_i - \theta)$ によって出力に変換される。ここで θ はユニットのしきい値と呼ばれ、ニューロンが興奮するポテンシャル値を表す。応答関数としては各種のものが提案、研究されているが、実用的検討には次式のシグモイド関数モデルが広く採用されている。

1993年4月9日受理

* 工業化学科 (Department of Industrial Chemistry)

$$f(x) = \{1 + \tanh(x/u_0)\} / 2 = 1 / \{1 + \exp(-2x/u_0)\} \quad (1)$$

このモデルの応答は図2に示すような非線形の飽和特性を持ち、係数 u_0 を小さな値にすれば初期のニューロンモデルとしてMcCullochとPittsによって提案されたしきい値関数モデルに対応する。

2. 2 階層ネットワークモデルとその学習

ネットワークモデルには、階層ネットワークと相互結合ネットワークがあるが、工学的応用の検討が進められているのはほとんど前者であり、後者については今後の研究の発展に待つところが多い。

ユニットが図3のように層状に配置し、信号は入力層から出力層に向かって一方向に各ユニットで変換されながら伝わるとするモデルを階層ネットワークモデルという。すなわち、層数を L 、 m 層 j ユニット入力を $l_{m,j}$ 、出力を $o_{m,j}$ とすると、入力層については、(2)式、中間層および出力層($m=2, 3, \dots, L$)については、(3)、(4)式によって信号が変換される。

$$o_{1,i} = l_{1,i} \quad (2)$$

$$l_{m,j} = \sum_i W_{m-1,j,i} o_{m-1,i} \quad (3)$$

$$o_{m,j} = f(l_{m,j} - \theta_{m,j}) \quad (4)$$

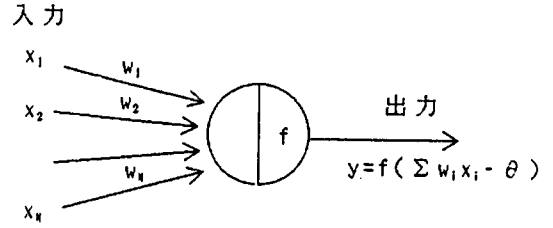


図1 ユニット

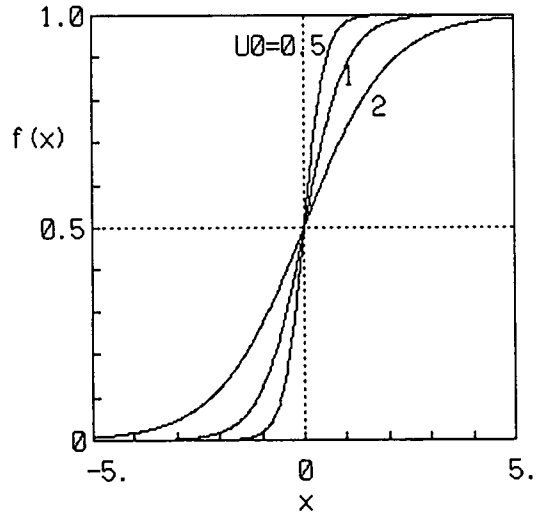


図2 シグモイド関数モデルの応答特性

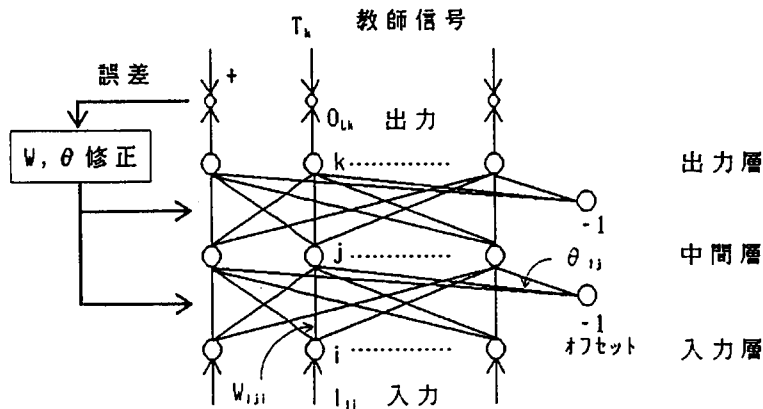


図3 階層ネットワークモデル

ネットワークの学習は、教師あり学習の場合、すべての学習データについて出力信号が教師信号にできるだけ一致するように結合荷重 W としきい値 θ を調整することによって行われる。この学習は各ユニットの出力がシグモイド関数のような連続関数の場合、出力と教師信号との誤差の自乗を最小とするように学習を行うので最小自乗平均誤差法と呼ばれている。すなわち学習は、ある特定の学習データ入力に対する教師信号を T 、出力層からの出力を O_L 、誤差を E_p とすると、すべての学習データについて(5)式を最小とするように W と θ を修正していく。ここで最小化法としてよく知られている勾配法を使って $\partial E_p / \partial W$ 、 $\partial E_p / \partial \theta$ (W と θ は数式上同等であって、 θ を W に、入力 I にオフセット要素-1を含めれば区別する必要はないので、以後 θ は省略する)の値にもとづいて E_p の減少する方向に結合係数を逐次更新していくのであるが、この勾配を出力層から入力層に向かって順次求めていく方法が、Rumelhartらによって提案された誤差逆伝搬(Back Propagation)アルゴリズムである。式(6)-(8)に計算法の考え方を示す。

$$E_p = \sum_k (T_k - O_{L,k})^2 / 2 \quad (5)$$

L-1層について、

$$\begin{aligned} \frac{\partial E_p}{\partial W_{L-1,k,j}} &= -(T_k - O_{L,k}) \frac{\partial O_{L,k}}{\partial W_{L-1,k,j}} \\ &= -(T_k - O_{L,k}) \frac{\partial f(I_{L,k})}{\partial W_{L-1,k,j}} \\ &= -(T_k - O_{L,k}) f'(I_{L,k}) O_{L-1,j} \quad (6) \end{aligned}$$

L-2層について、

$$\begin{aligned} \frac{\partial E_p}{\partial W_{L-2,j,i}} &= -\sum_k (T_k - O_{L,k}) \frac{\partial O_{L,k}}{\partial W_{L-2,j,i}} \\ &= -\sum_k (T_k - O_{L,k}) f'(I_{L,k}) \frac{\partial I_{L,k}}{\partial O_{L-1,j}} \\ &\quad \cdot \frac{\partial f(I_{L-1,j})}{\partial W_{L-2,j,i}} \\ &= -\sum_k (T_k - O_{L,k}) f'(I_{L,k}) W_{L-1,k,j} f'(I_{L-1,j}) O_{L-2,i} \quad (7) \end{aligned}$$

一般に $m(L-2, L-3, \dots, 1)$ 層については、

$$\begin{aligned} \frac{\partial E_p}{\partial W_{m,j,i}} &= \frac{\partial E_p(O_L(I_L(O_{L-1}(\dots, (I_{m+1})), \dots)))}{\partial W_{m,j,i}} \\ &= -\sum_{k_L} (T_k - O_{L,k_L}) f'(I_{L,k_L}) \\ &\quad \cdot \sum_{k_{L-1}} W_{L-1,k_L,k_{L-1}} f'(I_{L-1,k_{L-1}}) \\ &\quad \dots \dots \dots \\ &\quad \cdot \sum_{k_{m+2}} W_{m+2,k_{m+3},k_{m+2}} f'(I_{m+2,k_{m+2}}) \\ &\quad \cdot W_{m+1,k_{m+2},j} f'(I_{m+1,j}) O_{m,i} \quad (8) \end{aligned}$$

ここで、 k_L はL層ユニット番号、 W_{m,k_{m+1},k_m} はm層 k_m 番ユニットと $m+1$ 層 k_{m+1} 番ユニット間の結合係数である。以上の計算式をプログラミングに都合のよい表現に書き直すと次のようになる。

$$\partial E_p / \partial W_{m,j,i} = -\delta_{m+1,j} O_{m,i} \quad (9)$$

$$\begin{aligned} m &= L-1 \\ \delta_{m+1,j} &= (T_j - O_{m+1,j}) f'(I_{m+1,j}) \quad (10) \end{aligned}$$

$$\begin{aligned} m &= L-2, L-3, \dots, 1 \\ \delta_{m+1,j} &= \sum_k \delta_{m+2,k} W_{m+1,k,j} f'(I_{m+1,j}) \quad (11) \end{aligned}$$

学習、すなわち結合荷重の更新は、上記勾配を使ってパターンごとに誤差 E_p を序々に減少させるように行い、最終的に全パターンについての誤差 $E_t = \sum E_p$ の最小化をはかる逐次修正法が基本であるが、最急降下法は学習速度が遅いので(13)式のように先の修正量も考慮するモーメント法が有効であることが知られている。ここで、 t は学習回数、 α 、 β は小さな正の数である。

$$W_{m,j,i}(t+1) = W_{m,j,i}(t) + \Delta W_{m,j,i}(t) \quad (12)$$

$$\begin{aligned} \Delta W_{m,j,i}(t) &= -\alpha \partial E_p / \partial W_{m,j,i}(t) \\ &\quad + \beta \Delta W_{m,j,i}(t-1) \quad (13) \end{aligned}$$

結合係数の更新法としては全パターンについての勾配 $\partial E_t / \partial W_{m,j,i} = \sum \partial E_p / \partial W_{m,j,i}$ にもとづいて修正を行う一括修正法が考えられるが、複雑な問題には向かないとの報告もあり、あまり検討されていないようである。しかし、一括修正法では各種の優れた非線形

最適化法の適用が可能と考えられ、学習時間の短縮が期待される。そこで、本報では最適勾配(Optimum Gradient)法とBFGS(Broyden-Fletcher-Goldfarb-Shanno)法の応用による学習法についても検討した。

BFGS法は化学分野ではほとんど紹介されていないようであるが、著者の経験では微分方程式を含めて非線形形式のパラメータ決定にはきわめて一般的で強力な最適化計算法である。以下にアルゴリズムを示す。

いま、最小とすべき目的関数 $f(\mathbf{x})$ が(14)式の二次形式関数で近似できるとすれば、 i 番目の近似値を \mathbf{x}_i としてNewton-Raphson法を適用すると、次の更新値 \mathbf{x}_{i+1} は(15)式となる。次に一般的な非線形目的関数を考えると、修正方向を $\mathbf{p}_i = -\mathbf{G}_i^{-1} \mathbf{g}_i$ として、(16)式によって更新値を求めていくことが考えられる。しかし、これは二次微係数の逆行列の計算が必要なため、実用的ではない。

$$f(\mathbf{x}) = \mathbf{a} + \mathbf{c}^T \mathbf{x} + \mathbf{x}^T \mathbf{G} \mathbf{x} / 2 \quad (14)$$

$$\mathbf{x}_{i+1} = \mathbf{x}_i - \mathbf{G}_i^{-1} \mathbf{g}_i \quad (15)$$

$$\text{here, } \mathbf{g}_i = \text{grad}\{f(\mathbf{x}_i)\} = \mathbf{c} + \mathbf{G} \mathbf{x}_i$$

$$\mathbf{x}_{i+1} = \min_{\alpha} f(\mathbf{x}_i + \alpha \mathbf{p}_i) \quad (16)$$

これに対し、準Newton法は一次微係数の値のみを使って \mathbf{G}^{-1} に収束するようなマトリクス \mathbf{H} を逐次的に求めていく方法である。この逐次計算公式は数多く提案されているが、代表的なものが(20)式のDFP(Davidon-Fletcher-Powell)公式と(21)式のBFGS公式である。

$$\mathbf{p}_i = -\mathbf{H}_i \mathbf{g}_i \quad (17)$$

$$d\mathbf{x}_i = \mathbf{x}_{i+1} - \mathbf{x}_i \quad (18)$$

$$\mathbf{H}_{i+1} = \mathbf{H}_i + d\mathbf{H} \quad (19)$$

$$d\mathbf{H} = d\mathbf{H}_{DFP} = \frac{d\mathbf{x}_i d\mathbf{x}_i^T}{d\mathbf{x}_i^T d\mathbf{g}_i} - \frac{\mathbf{H}_i d\mathbf{g}_i^T \mathbf{H}_i}{d\mathbf{g}_i^T \mathbf{H}_i d\mathbf{g}_i} \quad (20)$$

$$d\mathbf{H} = d\mathbf{H}_{BFGS} + b\mathbf{v}_i \mathbf{v}_i^T \quad (21)$$

$$\text{here, } b = d\mathbf{g}_i^T \mathbf{H}_i d\mathbf{g}_i$$

$$\mathbf{v}_i = \frac{d\mathbf{x}_i}{d\mathbf{x}_i^T d\mathbf{g}_i} - \frac{\mathbf{H}_i d\mathbf{g}_i}{d\mathbf{g}_i^T \mathbf{H}_i d\mathbf{g}_i}$$

ここで、 \mathbf{H} を単位行列に固定すれば最適勾配法となる。

3. 化学パターン認識

複数の特徴からなる多数のデータについて、それぞれがどのようなカテゴリーに属するかを識別することがパターン認識である。一般にパターン認識は、文字、音声、画像認識について工学的応用と人間の知的情報処理メカニズムの解明の立場からの研究がよく知られている。

化学分野におけるパターン認識の応用は、1970年代にWoldやKowalskiらによって提唱された比較的新しい計量化学(Chemometrics)の一つの主要なテーマとして発展してきており、最近では、分子設計への応用の立場から、特徴を化学構造(構造パラメータ)として、生理活性や薬理効果を分類、予測する構造活性相関研究が注目されている。

化学パターン認識に使用される数学手法は基本的には多変量解析法である。多変量解析法についての教科書、研究書、ソフトウェアは数多いが、化学系学生や研究者にとっては、Kowalskiらによって開発された化学パターン認識プログラムARTHURと著書"Chemometrics"は大いに参考になるものである。本ニューラルネットワークによる化学パターン認識結果の評価にはこのプログラムと多変量解析パッケージ(日本能率協会総合研究所)を使用した。

検討に使用した化学パターン認識問題例題は文献6)より引用したもので、北米大陸の4ヶ所の産地における黒耀石サンプルに含まれる微量金属元素量(成分:Fe, Ti, Ba, Ca, K, Mn, Rb, Sr, Y, Zr, 含有量:10-1700ppm)をもとに、考古学的出土品(矢じり)に使用された黒耀石の採取地を判定するものである。実際に使用した学習データ(産地既知サンプル)とテストデータ(産地未知サンプル)はAppendixに示した。ARTHURによる解析結果例を以下に示す。図4はパターン認識手法の中でも図示法としてよく知られている線形写像法(K-L変換法)による結果である。この方法は主成分分析の応用であり、規格化されたデータの分散共分散行列の固有値と固有ベクトルを求め、固有値の大きな軸(分散の大きな軸、主成分)から二次元空間に写像して解析するものである。図4より、サンプル番号64, 68, 69, 70以外は産地の判定が可能である。図5は、多次元空間におけるサンプル間の距離をできるだけ保持して二次元空間に写像する非線形写像法による結果である。この図からはサンプル番号64, 66, 67, 70の判定についてはあいまいさが認められる。サンプル番号75については図に表示されていないが、産地3の下方に位置している。図6はクラスター分析結果例(重心法)

であるが、サンプル75を除けば4つのカテゴリーに分類でき、サンプル64は産地3、サンプル70は産地2と判定されている。以上を総合したテストデータの判定結果を表1に示す。

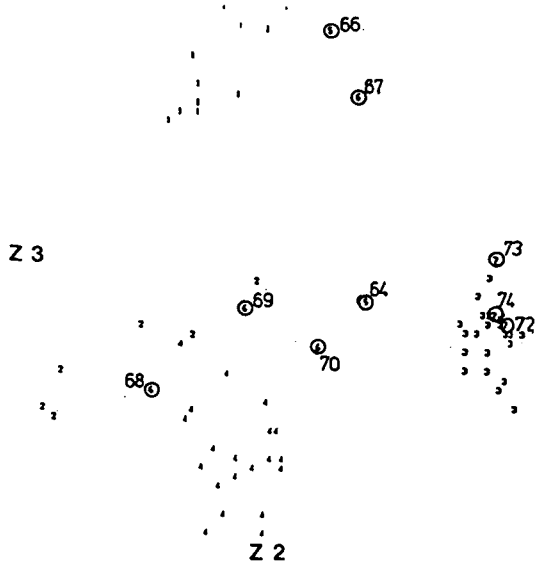


図4 線形写像法結果例

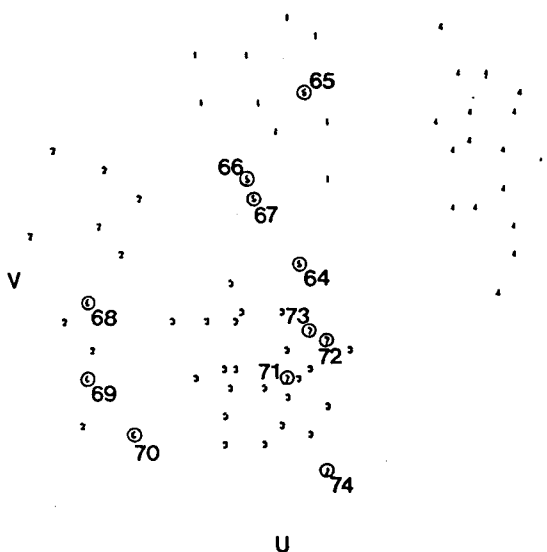


図5 非線形写像法結果例

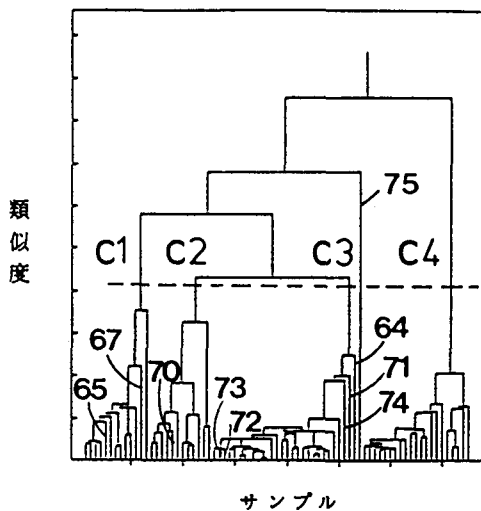


図6 クラスタ分析結果例

サンプル番号	産地	サンプル番号	産地
64	3	70	2
65	1	71	3
66	1	72	3
67	1	73	3
68	2	74	3
69	2	75	3

表1 既存法による判定結果

4. ニューラルネットワークによる検討結果

ニューラルネットワークの学習は非線形最適化問題の一つといえるが、通常のパラメータ決定問題と異なるのは「ゆらぎ」が重要な要素であることである。このために結合係数の初期値は乱数を用いて適切な値に設定する必要がある。ここで適切な値の意味は、大きすぎても小さすぎても学習はうまく進行しないし、また、ネットワークの自由度が大きい場合（ユニット数の多い場合）は複数の収束点が存在するからである。本検討では試行錯誤から±0.5の範囲とした。もう一つの大きな違いは、最初から最適勾配法や共役勾配法を適用してパラメータの最適化を目指す学習が全く進まないことである。したがって学習にはなんらかの

形でこの「ゆらぎ」すなわち確率的振動を与えることが必要であり、その最も簡単な方法が最急降下法（修正係数 α を一定値に固定する方法）の適用である。しかし、よく知られているように最急降下法は収束速度がきわめて遅いのが欠点である。本検討の結果、学習がある程度進行した後に非線形最適化手法を適用すればきわめて速く学習が収束することがわかった。

ニューラルネットワークによるパターン認識結果例を図7、表2-5に示す。これらは、原データの 카테고리3, 4のうちクラスター分析結果から特によく類似したサンプルの半分の削除した学習データについて、中間層ユニット数2の3層ネットワークによる結果であるが、学習時間は最も速いBFGS法で10分弱(PC-

9801DA)である。しかし、カテゴリーの境界にあるサンプル64や67が高すぎる率で判定されるため、本例においては学習を早めに打ち切ることが望ましいのではないかとと思われる。

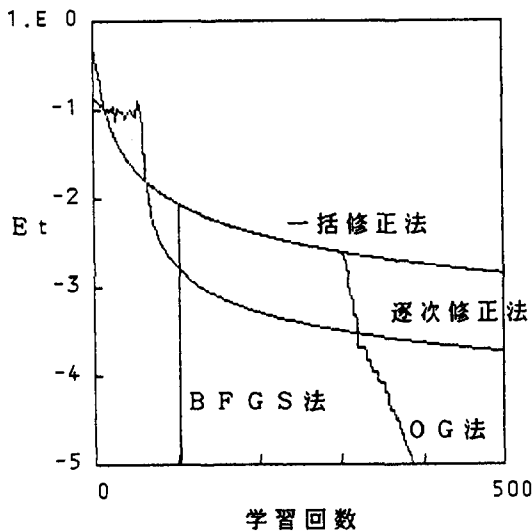


図7 学習法による収束速度の比較

サンプル 番号	カテゴリー			
	1	2	3	4
64	0.007	0.012	0.511	0.000
65	0.989	0.000	0.008	0.006
67	0.901	0.000	0.081	0.000
70	0.000	0.849	0.120	0.001
74	0.010	0.008	0.989	0.000

表2 逐次修正法（モーメント法）による判定結果 ($\alpha=0.15, \beta=0.675$)

サンプル 番号	カテゴリー			
	1	2	3	4
64	0.002	0.360	0.397	0.001
65	0.968	0.000	0.027	0.015
67	0.649	0.000	0.166	0.002
70	0.000	0.849	0.120	0.001
74	0.010	0.008	0.989	0.000

表3 一括修正法（最急降下法）による判定結果

サンプル 番号	カテゴリー			
	1	2	3	4
64	0.000	0.192	0.588	0.000
65	0.992	0.000	0.007	0.005
67	0.804	0.000	0.134	0.000
70	0.000	0.960	0.030	0.002
74	0.009	0.008	0.988	0.000

表4 一括修正法（OG法）による判定結果

サンプル 番号	カテゴリー			
	1	2	3	4
64	0.000	0.000	1.000	0.000
65	1.000	0.000	0.000	0.000
67	0.997	0.000	0.008	0.000
70	0.000	1.000	0.000	0.000
74	0.000	0.000	1.000	0.000

表5 一括修正法（BFGS法）による判定結果

原データではカテゴリー1, 2と3, 4とでは学習データ数にアンバランスがあるために誤判定されたり学習が停止してしまうことが多く, 学習計算においては工夫が必要であった. 一括修正法の場合, カテゴリーごとのデータ数に応じて勾配に重み付けをすることによって良好な結果が得られた. 一方, 逐次修正法では, 先の例のように一部のデータを削除するか, または, カテゴリー1, 2のデータを二倍学習させることによって同様の結果が得られた.

次にデータの規格化の問題であるが, ニューラルネットワークによるパターン認識では, データに特定の統計的性質を仮定しないので, 0-1の規格化を用いることが多く, 多変量解析法で一般に行われるオートスケーリングは必ずしも必要ではないといえる. 本検討では学習速度と判定の精度の面から min-maxスケーリングによって±2に規格化した. ただし, 逐次修正法ではオートスケーリングをした後に絶対値の最大値によって±2に規格化した方が結果は良好であった.

船橋¹⁸⁾よれば, ユニットがシグモイド関数モデルである4層ネットワークによって, 中間層ユニット数を必要なだけ増やせば, 任意の連続関数を任意の精度で近似することができることが報告されている. しかし, ネットワークの自由度が増すと過剰適応現象が起り, 誤判定や見かけ上の高い判定率を与えることが知られている. 本検討では, 原学習データについて層数4, 中間ユニット数6までの場合を調べてみたが, 層数については変化はなく, 中間ユニット数が4および6のときにサンプル64が誤判定または判定不能となる結果を得た.

実用性に関する検討として, 学習データの23個をカテゴリーを区別せずにランダムに削除し, 学習データを40個, テストデータを35個として産地判定を行ってみた. その結果, 削除した元の学習データを含めて認識率は約90%であった. 元のテストデータのうち誤判定されたのはサンプル番号64と67のみであった. 学習の比較的早い段階で高い率で判定されたテストデータを学習データに追加して再学習させる方法等をとればさらに認識率の向上が考えられ, ニューラルネットワークは化学パターン認識法として有用な方法であるといえる.

5. おわりに

一般に化学プロセスは強い非線形性と不確定性を有することが特徴であるが, ニューラルネットワークは学習によって非線形相関ができることが大きな特徴で

あるため, 解析的手法にもとづく従来のシステム解析合成手法では困難であったモデリング, 制御, 診断等の諸問題への応用が期待できる. 現在, 文字や画像認識問題の場合はユニット数が膨大となるため, ニューラルネットワーク手法は従来の多変量解析法に優る評価はなされていないし, また, 将来性についても未知な部分が指摘されている報告もあるようである. しかし, 化学パターン認識問題はそれらに比較すれば小規模であることから, ニューラルネットワーク手法を効果的に応用できる問題が数多くあるのではないかと思われる.

本稿の内容は, 1992年度の卒業研究課題として実施したものであるが, 内容についての学生の反応は良好であった. 本稿が, これから計算機化学分野を目指そうとする学生と指導の立場にある方々の参考になることを期待したい.

最後に, 本検討に際し, 貴重な資料と助言をいただきました豊橋技術科学大学工学部宮下芳勝助教授に感謝いたします.

参考文献

- 1) 中野 馨監修, 飯沼一元編, ニューロコンピュータ, 技術評論社(1989)
- 2) 麻生英樹, ニューラルネットワーク情報処理, 産業図書(1988)
- 3) S. D. Brown, R. S. Bear, Jr., T. B. Bink, Chemometrics, Analytical Chemistry(1992)22R
- 4) J. Zupan, J. Gasteiger, Neural networks, Analytica Chimica Acta, 248(1991)1-30
- 5) M. A. Sharaf, D. L. Illman, B. R. Kowalski, Chemometrics, John Wiley & Sons(1986)
- 6) B. R. Kowalski, T. F. Schatzki, F. H. Stross, Analytical Chemistry(1972)2176
- 7) L. C. W. Dixon著, 松原正一訳, 非線形最適化計算法, 培風館(1974)
- 8) 佐々木慎一, 阿部英次, 高橋由雅, 高山千代蔵, 宮下芳勝, 化学者のためのパターン認識序説, 東京化学同人(1984)
- 9) システム制御情報学会誌, 35, 1, (1991)
- 10) 計測自動制御学会誌, 30, 4, (1991)
- 11) システム制御情報学会誌, 36, 10, (1992)

< A p p e n d i x >

A 1. 使用データ

◇学習データ

No.	金属元素含有量[ppm]										産地		
	Fe	Ti	Ba	Ca	K	Mn	Rb	Sr	Y	Zr	4	3	2 1
1	1100.	390.	55.	920.	460.	45.	120.	57.	58.	142.	0	0	0 1
2	1173.	417.	54.	961.	441.	47.	135.	55.	60.	145.	0	0	0 1
3	1164.	404.	56.	916.	446.	42.	120.	58.	45.	148.	0	0	0 1
4	1030.	373.	59.	920.	487.	38.	128.	53.	58.	138.	0	0	0 1
5	1077.	373.	55.	888.	455.	38.	97.	51.	54.	145.	0	0	0 1
6	1080.	403.	53.	919.	442.	41.	133.	60.	45.	155.	0	0	0 1
7	1020.	360.	59.	883.	473.	43.	119.	40.	50.	134.	0	0	0 1
8	1050.	396.	56.	924.	482.	48.	140.	74.	71.	157.	0	0	0 1
9	1100.	373.	53.	910.	477.	51.	137.	61.	58.	152.	0	0	0 1
10	1069.	375.	51.	958.	429.	42.	100.	51.	47.	128.	0	0	0 1
11	863.	183.	8.	626.	452.	34.	121.	15.	58.	70.	0	0	1 0
12	1108.	289.	7.	783.	426.	41.	109.	15.	57.	67.	0	0	1 0
13	1210.	276.	10.	966.	430.	44.	117.	20.	44.	73.	0	0	1 0
14	1205.	291.	10.	975.	420.	43.	115.	25.	58.	73.	0	0	1 0
15	1100.	267.	10.	910.	500.	40.	145.	25.	65.	95.	0	0	1 0
16	1100.	280.	10.	872.	515.	49.	145.	38.	60.	65.	0	0	1 0
17	689.	114.	9.	534.	404.	26.	110.	25.	50.	55.	0	0	1 0
18	1186.	257.	10.	940.	431.	40.	121.	20.	53.	73.	0	0	1 0
19	860.	182.	7.	722.	418.	33.	115.	20.	55.	53.	0	0	1 0
20	952.	131.	37.	320.	430.	34.	118.	25.	65.	165.	0	1	0 0
21	933.	120.	36.	321.	414.	31.	125.	10.	59.	163.	0	1	0 0
22	999.	129.	38.	343.	446.	34.	121.	15.	52.	161.	0	1	0 0
23	975.	136.	38.	333.	418.	36.	111.	10.	53.	147.	0	1	0 0
24	885.	118.	32.	294.	382.	27.	95.	15.	55.	134.	0	1	0 0
25	928.	127.	37.	337.	455.	30.	109.	15.	47.	151.	0	1	0 0
26	890.	122.	33.	315.	380.	34.	106.	15.	60.	155.	0	1	0 0
27	840.	133.	42.	320.	405.	31.	100.	10.	55.	197.	0	1	0 0
28	890.	119.	37.	272.	384.	32.	117.	15.	50.	160.	0	1	0 0
29	1000.	135.	36.	335.	385.	32.	100.	15.	50.	135.	0	1	0 0
30	930.	131.	32.	342.	385.	28.	100.	10.	55.	140.	0	1	0 0
31	880.	156.	36.	279.	388.	37.	103.	15.	53.	143.	0	1	0 0
32	960.	137.	35.	315.	390.	34.	110.	10.	60.	167.	0	1	0 0
33	970.	133.	40.	310.	390.	36.	115.	15.	65.	160.	0	1	0 0
34	819.	131.	38.	435.	392.	35.	108.	15.	54.	150.	0	1	0 0
35	932.	147.	37.	318.	393.	32.	122.	28.	56.	174.	0	1	0 0
36	970.	140.	40.	338.	394.	36.	118.	15.	61.	160.	0	1	0 0
37	1000.	146.	41.	360.	394.	33.	125.	15.	48.	170.	0	1	0 0
38	970.	133.	40.	310.	390.	36.	115.	15.	65.	160.	0	1	0 0
39	930.	140.	36.	335.	395.	33.	100.	10.	65.	170.	0	1	0 0

40	960.	138.	35.	323.	395.	52.	103.	10.	55.	145.	0	1	0	0
41	990.	142.	38.	405.	400.	37.	122.	15.	60.	155.	0	1	0	0
42	918.	133.	35.	317.	408.	35.	120.	15.	58.	190.	0	1	0	0
43	1525.	270.	54.	860.	345.	65.	83.	47.	60.	203.	1	0	0	0
44	1565.	374.	54.	813.	295.	59.	88.	42.	49.	187.	1	0	0	0
45	1705.	441.	66.	835.	310.	66.	91.	44.	42.	187.	1	0	0	0
46	1650.	294.	60.	945.	367.	62.	91.	39.	58.	192.	1	0	0	0
47	1630.	314.	54.	934.	340.	67.	95.	33.	38.	220.	1	0	0	0
48	1635.	405.	58.	868.	311.	56.	83.	47.	60.	154.	1	0	0	0
49	1635.	385.	55.	845.	290.	62.	85.	47.	58.	196.	1	0	0	0
50	1480.	310.	54.	850.	320.	66.	71.	27.	47.	184.	1	0	0	0
51	1720.	425.	58.	893.	316.	65.	93.	33.	44.	192.	1	0	0	0
52	1560.	269.	56.	915.	350.	64.	96.	47.	52.	194.	1	0	0	0
53	1705.	333.	55.	1010.	345.	74.	94.	45.	61.	217.	1	0	0	0
54	1700.	360.	55.	945.	355.	60.	100.	47.	58.	183.	1	0	0	0
55	1675.	342.	56.	920.	360.	61.	91.	49.	59.	215.	1	0	0	0
56	1695.	358.	57.	920.	347.	67.	93.	42.	71.	192.	1	0	0	0
57	1645.	327.	65.	940.	322.	69.	88.	44.	60.	208.	1	0	0	0
58	1720.	392.	58.	890.	344.	64.	97.	48.	50.	181.	1	0	0	0
59	1620.	385.	55.	835.	295.	59.	87.	22.	60.	203.	1	0	0	0
60	1615.	337.	50.	850.	320.	66.	92.	48.	66.	212.	1	0	0	0
61	1675.	325.	62.	930.	350.	66.	89.	50.	63.	224.	1	0	0	0
62	1685.	353.	59.	875.	318.	60.	91.	41.	60.	203.	1	0	0	0
63	1600.	316.	54.	915.	347.	83.	82.	52.	66.	209.	1	0	0	0

◇テストデータ

No.	Fe	Ti	Ba	Ca	K	Mn	Rb	Sr	Y	Zr
64	1050.	195.	46.	865.	400.	36.	97.	10.	50.	160.
65	1010.	357.	65.	900.	455.	42.	112.	65.	55.	145.
66	920.	320.	60.	830.	440.	33.	110.	43.	60.	143.
67	1000.	194.	58.	965.	460.	42.	125.	30.	60.	145.
68	920.	215.	6.	650.	435.	37.	122.	15.	65.	75.
69	780.	140.	12.	605.	415.	35.	130.	15.	50.	70.
70	680.	105.	10.	460.	400.	31.	127.	15.	57.	87.
71	920.	127.	39.	475.	430.	34.	98.	5.	60.	156.
72	900.	115.	37.	310.	440.	34.	108.	30.	60.	175.
73	920.	138.	41.	350.	450.	34.	110.	20.	62.	157.
74	775.	110.	35.	327.	337.	34.	93.	15.	65.	132.
75	935.	131.	38.	360.	420.	37.	105.	10.	85.	150.

A 2. 出力計算, BP計算, 最適化計算サブルーチン

◇使用変数

NL	:層数
NU(L)	:L層ユニット数
IN(L, I)	:L層Iユニット入力

OUT(L, I) :L層Iユニット出力 (オフセットを含む)
 W(L, J, I) :L+1層Jユニット-L層Iユニット間結合係数 (オフセット係数を含む)
 TP(K) :Kユニット教師信号
 ET :全パターンについての自乗誤差の総和
 DELT(L, J) :(10), (11)式の $\delta_{L, J}$
 X(K) :W(L, J, I)の更新計算における作業変数 (相互変換にはKF(L, J, I), SKLJI(K, L, J, I)を使用する)

◇出力計算とBP計算

<L層各ユニットの入出力計算(2<=L<=NL)>

$$\begin{bmatrix} IN(L, 1) \\ IN(L, 2) \\ \text{---} \\ \text{---} \\ IN(L, NU(L)) \end{bmatrix} = \begin{bmatrix} W(L-1, 1, 1), W(L-1, 1, 2), \text{---}, W(L-1, 1, NU(L-1)), TH(L-1, 1) \\ \text{---} \\ \text{---} \\ W(L-1, NU(L), 1), \text{---}, W(L-1, NU(L), NU(L-1)), TH(L-1, NU(L)) \end{bmatrix} \times \begin{bmatrix} OUT(L-1, 1) \\ OUT(L-1, 2) \\ \text{---} \\ OUT(L-1, NU(L-1)) \\ 1.0 \end{bmatrix}$$

OUT(L, I)=f(IN(L, I)), I=1, NU(L)

TH:オフセット係数 1.0:オフセット

FUNCTION F(KK, X) →KK=0:誤差の計算 (目的関数), KK≠0:dF/dX_{KK}の計算

COMMON /BL1/NL, NU(5), U0, OUTLSF, MITT

COMMON /BL2/IN(5, 10), OUT(5, 11), W(4, 10, 11), TP(10), ET

COMMON /BLS/MNW, NW(4), DELT(5, 10)

DIMENSION X(40)

REAL IN

F=0

IF(KK.GT.0) GO TO 5

DO 100 I=1, NU(1)

OUT(1, I)=IN(1, I)

100 CONTINUE

DO 110 L=2, NL

DO 120 J=1, NU(L)

IN(L, J)=0

DO 130 I=1, NU(L-1)+1

130 IN(L, J)=IN(L, J)+X(KF(L-1, J, I))*OUT(L-1, I)

OUT(L, J)=SF(IN(L, J))

120 CONTINUE

110 CONTINUE

*

DO 140 K=1, NU(NL)

140 F=F+(TP(K)-OUT(NL, K))**2

F=F/2.

RETURN

*

5 CALL SKLJI(KK, L, J, I)

*

入力→出力計算と
誤差Epの計算

→ SF:シグモイド関数

```

IF(L.EQ.NL-1) THEN
  DELT(L+1, J)=(TP(J)-OUT(L+1, J))*SFD(IN(L+1, J))
  F=-DELT(L+1, J)*OUT(L, I)
ELSE
  DELT(L+1, J)=0
  DO 150 MM=1, NU(L+2)
150  DELT(L+1, J)=DELT(L+1, J)
  ++DELT(L+2, MM)*X(KF(L+1, J, MM))*SFD(IN(L+1, J))
  F=-DELT(L+1, J)*OUT(L, I)
ENDIF
END

```

→ SFD: シグモイド関数の微分値
B P アルゴリズムによる勾配計算

```

*
FUNCTION KF(L, J, I)          → W(L, J, I)-X(KK)間の添え字換算
COMMON /BL1/NL, NU(5), U0, OUTLSP, MITT
COMMON /BL5/MNW, NW(4), DELT(5, 10)
*
--- L, J, I -> K ---
KF=0
DO 10 LL=NL-1, L+1, -1
10  KF=KF+NW(LL)
KF=KF+J+(I-1)*NU(L+1)
END
*
SUBROUTINE SKLJI(K, L, J, I)  → X(K)-W(L, J, I)間の添え字換算
COMMON /BL1/NL, NU(5), U0, OUTLSP, MITT
COMMON /BL5/MNW, NW(4), DELT(5, 10)
*
--- K -> L, J, I ---
NWS=0
I=NL-1
10 IF(NWS+NW(I).GE.K) GOTO 20
NWS=NWS+NW(I)
I=I-1
GOTO 10
20 L=I
KK=K-NWS
I=(KK-1)/NU(L+1)
J=KK-I*NU(L+1)
I=I+1
END

```

◇最適化計算 (結合係数の更新計算)

```

SUBROUTINE SWOPTT(ITT, IOPT, IALS, ALSM, X)
*
ITT: iteration number, IOPT: 1. Steepest Decent 2. Optimum Gradient 3. DFP 4. BFS 5. BFGS
*
IALS: flag of  $\alpha$  search (1. auto-search), ALSM:  $\alpha$  max
*
Y=FET(X1, X2, ..., XM) ----> MIN.          →関数FET, Fを変更すれば任意の最適化計算に対応できる
COMMON /BL1/NL, NU(5), U0, OUTLSP, MITT
COMMON /BL2/IN(5, 10), OUT(5, 11), W(4, 10, 11), TP(10), ET

```

```

COMMON /BL5/MNW, NW(4), DELT(5, 10)
COMMON /BL6/GFT(40), GFDT(40), SST(40), H(40
, 40)
REAL IN
DIMENSION X(40), RS(2), SGM(40)
*
M=MNW
YT=ET
*
IF(IOPT.EQ. 1. OR. IOPT.EQ. 2) THEN
DO 10 I=1, M
10 SST(1)=-GFT(I)
ENDIF
*
IF(IOPT.EQ. 2) THEN
CALL SOKAI(M, IALS, ALSM, X, SST, YT, RS)
CALL FBN(16, M, X, SST, RS, ALS)
ELSE IF(IOPT.GE. 3) THEN
CALL SOKAI(M, IALS, ALSM, X, SST, YT, RS)
CALL FPH(IOPT, ITT, M, GFT, GFDT, SGM, H, SST)
CALL FBN(16, M, X, SST, RS, ALS)
ENDIF
DO 100 J=1, M
SGM(J)=ALS*SST(J)
X(J)=X(J)+SGM(J)
GFDT(J)=GFT(J)
100 CONTINUE
END
*
FUNCTION FET(X) → Etの計算
COMMON /BL1/NL, NU(5), UO, OUTLSP, MITT
COMMON /BL2/IN(5, 10), OUT(5, 11), W(4, 10, 11
), TP(10), ET
COMMON /BL3/ND, IPD(100, 10), TPD(100, 10)
REAL IN, IPD, X(40)
FET=0
DO 10 ITD=1, ND
DO 100 I=1, NU(1)
100 IN(1, I)=IPD(ITD, I)
DO 110 I=1, NU(NL)
110 TP(I)=TPD(ITD, I)
Y=F(0, X)
FET=FET+Y
10 CONTINUE
END
*
SUBROUTINE GRF(M, X, GF, Y)
DIMENSION X(40), GF(40)
Y=F(0, X)
DO 10 I=1, M
10 GF(I)=F(1, X)
RETURN
END
*
SUBROUTINE SOKAI(M, IALS, ALSM, X, SS, Y, RS)
αの近似値計算
DIMENSION X(40), SS(40), RS(2)
IF(IALS.EQ. 1) THEN
ALD=0.
DAL=1. E5
DO 10 I=1, M
IF(ABS(SS(I)). LT. 1. E-6) GOTO 10
DDAL=ABS(X(I)/SS(I))
IF(DDAL. LT. DAL) DAL=DDAL
10 CONTINUE
DAL=DAL/10.
ELSE
AL=0
ALD=AL
DAL=ALSM/100
DO 30 IT=1, 100
AL=AL+DAL
YN=G(M, X, SS, AL)
IF(YN. LT. Y) THEN
Y=YN
ALD=AL
ENDIF
30 CONTINUE
END IF
AL=ALD
RS(1)=AL-DAL
IF(RS(1). LT. 0.) RS(1)=0.
RS(2)=AL+DAL
RETURN
END
*
SUBROUTINE FPH(IOPT, IT, M, GF, GFD, SGM, H, SS)
DIMENSION GF(40), GFD(40), SGM(40), H(40, 40)
, SS(40), V(40), HY(40)
DIMENSION YY(40), A(40, 40), B(40, 40)
DO 5 I=1, M
DO 7 J=1, M

```

```

      A(I, J)=0.
      B(I, J)=0.
7  CONTINUE
5  CONTINUE
   IF(IT.EQ.1) THEN
   DO 10 I=1, M
   DO 20 J=1, M
   H(I, J)=0.
   IF(I.EQ.J) H(I, J)=1.
20 CONTINUE
10 CONTINUE
   ELSE
   DO 30 J=1, M
30 YY(J)=GF(J)-GFD(J)
   SY=0.
   DO 40 J=1, M
40 SY=SY+SGM(J)*YY(J)
   IF(ABS(SY).LT.1.E-7) THEN
   WRITE(*,*) ' [Warning] nearly over flow
               at SY'
   END IF
   DO 50 I=1, M
   DO 60 J=1, M
   IF(ABS(SY).LT.1.E-10) GOTO 60
   A(I, J)=SGM(I)*SGM(J)/SY
60 CONTINUE
50 CONTINUE
   DO 70 I=1, M
   SGM(I)=0.
   DO 80 J=1, M
80 SGM(I)=SGM(I)+H(I, J)*YY(J)
70 CONTINUE
   YHY=0.
   DO 90 J=1, M
90 YHY=YHY+YY(J)*SGM(J)
   IF(ABS(YHY).LT.1.E-7) THEN
   WRITE(*,*) ' [Warning] nearly over flow
               at YHY'
   END IF
   DO 100 I=1, M
   SGM(I)=0.
   GFD(I)=0.
   DO 110 J=1, M
   SGM(I)=SGM(I)+H(I, J)*YY(J)
   GFD(I)=GFD(I)+YY(J)*H(J, I)
110 CONTINUE
100 CONTINUE
   DO 120 I=1, M
   DO 130 J=1, M
   IF(ABS(YHY).LT.1.E-10) GOTO 135
   B(I, J)=-SGM(I)*GFD(J)/YHY
135 A(I, J)=A(I, J)+B(I, J)
130 CONTINUE
120 CONTINUE
   DO 300 I=1, M
   V(I)=0.
   DO 310 J=1, M
310 V(I)=H(I, J)*YY(J)
   V(I)=SGM(I)-V(I)
300 CONTINUE
   AC=0.
   DO 320 I=1, M
320 AC=AC+V(I)*YY(I)
   AC=AC/SY
   IF(IOPT.EQ.3.OR.AC.LE.0.) THEN
   DO 330 I=1, M
   DO 340 J=1, M
340 B(I, J)=0.
330 CONTINUE
   ENDIF
   IF(IOPT.EQ.3) GOTO 500
   IF(IOPT.EQ.5.AND.AC.LE.0.) GOTO 500
400 DO 200 I=1, M
   V(I)=0
   DO 210 J=1, M
210 V(I)=V(I)+H(I, J)*YY(J)/YHY
   V(I)=-V(I)+SGM(I)/SY
200 CONTINUE
   PHAI=YHY
   DO 220 I=1, M
   DO 230 J=1, M
230 B(I, J)=V(I)*V(J)*PHAI
220 CONTINUE
500 DO 240 I=1, M
   DO 250 J=1, M
250 H(I, J)=H(I, J)+A(I, J)+B(I, J)
240 CONTINUE
   END IF
   DO 140 I=1, M
   SS(I)=0.
   DO 150 J=1, M
150 SS(I)=SS(I)-H(I, J)*GF(J)

```

```

140 CONTINUE
    RETURN
    END
*
SUBROUTINE FBN(N, M, X, SS, RS, ALS)
*   フォック法による $\alpha$ の一次元探索
    DIMENSION X(40), SS(40)
    DIMENSION RS(2), AL(2), VG(2), FN(25)
    FN(1)=1.
    FN(2)=2.
    DO 10 I=3, N
10  FN(I)=FN(I-1)+FN(I-2)
        J=1
        D=FN(N-2)/FN(N)*(RS(2)-RS(1))
        DO 20 I=1, 2
            AL(I)=RS(I)+(-1.)**(I+1)*D
            VG(I)=G(M, X, SS, AL(I))
20  CONTINUE
30  IF(VG(1).GT.VG(2)) THEN
        RS(1)=AL(1)
        AL(1)=AL(2)
        VG(1)=VG(2)
        ID=2
    ELSE
        RS(2)=AL(2)
        AL(2)=AL(1)
        VG(2)=VG(1)
        ID=1
    END IF
        J=J+1
        IF(J.EQ.N-1) GO TO 40
        D=FN(N-1-J)/FN(N+1-J)*(RS(2)-RS(1))
        AL(ID)=RS(ID)+(-1.)**(ID+1)*D
        VG(ID)=G(M, X, SS, AL(ID))
        GO TO 30
40  ALS=(RS(1)+RS(2))/2.
        RETURN
    END
*
FUNCTION G(M, X, SS, AL)
    DIMENSION X(40), SS(40), XD(40)
    DO 10 I=1, M
10  XD(I)=X(I)+AL*SS(I)
        G=FET(XD)
    RETURN
    END

```