

修士学位論文

非線形変換を用いた
モジュール型ニューラルネットワークによる
パターン認識に関する研究

東北大学大学院工学研究科
電気・通信工学専攻 阿曾研究室
森島 公一

目次

第1章 序論	1
1.1 本研究の背景	1
1.1.1 パターン認識とは	1
1.1.2 パターン認識の問題点	3
1.2 階層型ニューラルネットワーク	4
1.3 研究の目的	4
1.4 本論文の構成	5
第2章 モジュール型ニューラルネットワーク	6
2.1 はじめに	6
2.2 3層パーセプトロンについて	6
2.2.1 構造	7
2.2.2 学習法	8
2.2.3 学習の問題点	9
2.3 モジュール型ニューラルネットワークについて	10
2.3.1 構造	10
2.3.2 学習法	11
2.3.3 モジュール型ニューラルネットワークを用いる利点	12
2.4 シグモイド型出力関数の問題点	13
第3章 RBF 出力素子を用いたモジュール型ニューラルネットワーク	15
3.1 はじめに	15
3.2 RBF 出力素子を用いたモジュールの分離境界面	15
3.3 RBF 出力素子を用いたモジュールの入出力関係	16
3.4 RBF 出力素子を用いたモジュールの学習	18
3.5 数値実験	18

3.6	RBF 出力素子を用いたモジュールの問題点	19
3.6.1	RBF 出力素子を用いたモジュールで対応できる場合	19
3.6.2	RBF 出力素子を用いたモジュールで対応できない場合	21
3.6.3	シグモイド関数に代わる理想的な非線形変換について	22
第4章	非線形変換を用いたモジュール型ニューラルネットワーク	24
4.1	はじめに	24
4.2	非線形変換を用いたモジュールの構成	24
4.3	非線形変換を用いたモジュールの動作	26
4.4	計算機実験	28
4.4.1	実験1 非線形変換を用いたモジュールの動作確認	28
4.4.2	実験2 特徴量が複雑に分布するパターンを含む3クラス問題	44
4.4.3	実験3	47
第5章	結論	50
5.1	結論	50
5.2	今後の課題	50
	謝辞	52
	発表予定学会	53
	参考文献	54

目 次

1.1	特徴量の分布の様子 (2次元)	2
1.2	ガウス分布で特徴量分布を近似することの問題点	3
2.1	3層パーセプトロン	7
2.2	モジュール型ニューラルネットワーク	10
2.3	1つのモジュール	11
2.4	入力特徴量2次元、中間層の素子2つのモジュール	14
2.5	中間層の様子 (横軸 y_1 、縦軸 y_2)	14
3.1	RBF 出力素子を用いたモジュールの分離境界面 (楕円状)	16
3.2	RBF 出力素子をもつモジュールと得られる境界の様子	17
3.3	RBF 出力素子を用いたモジュールによる認識 (成功例)	20
3.4	RBF 出力素子を用いたモジュールによる認識 (失敗例)	21
3.5	理想的な非線形変換を行うモジュールによる認識	23
4.1	非線形変換を用いたモジュール	25
4.2	非線形変換を用いたモジュールの動作	27
4.3	実験 1-1 使用したモジュールと特徴量	28
4.4	実験 1-1 認識の様子	29
4.5	実験 1-2 使用したモジュールと特徴量	30
4.6	実験 1-2 認識の様子	31
4.7	実験 1-3 使用したモジュールと特徴量	32
4.8	実験 1-3 認識の様子	33
4.9	実験 1-4 使用したモジュールと特徴量	34
4.10	実験 1-4 認識の様子	35
4.11	実験 1-5 使用したモジュールと特徴量	36
4.12	実験 1-5 認識の様子	37

4.13	x_1^2 、 x_2^2 、 x_1x_2 をすべて用いたモジュール	38
4.14	実験 1-6 認識の様子 (1)	39
4.15	実験 1-6 認識の様子 (2)	40
4.16	実験 1-6 認識の様子 (3)	41
4.17	実験 1-6 認識の様子 (4)	42
4.18	特徴量が複雑に分布するパターンを含む 3 クラス問題	44
4.19	モジュール 1 の中間層における識別境界の生成	45
4.20	モジュール 2 の中間層における識別境界の生成	45
4.21	非線形項を使用したネットワークによる認識	46
4.22	実験結果 1 各手法における学習に要した更新回数	48

第1章

序論

1.1 本研究の背景

膨大なデータの中から所望の情報を即座に獲得する検索技術、文字情報から音声情報へといったメディア変換技術、あるいは人間に代わってさまざまな情報処理を行う知的エージェントなど、本格的なマルチメディア時代を迎えるにあたって種々のメディアを効率的に処理しなければならない機会が急速に増えつつある。そのような要求に応えるための基本技術がパターン認識である

1.1.1 パターン認識とは

パターン認識とは、観測されたパターンをあらかじめ定められた複数の概念のうちの一つに対応させる処理である。この概念をクラスと呼んでいる。例えばアルファベットの認識であれば、入力パターンを26個のクラスのいずれかに対応させる処理ということになる。パターンというと人間の視覚に入ってくる2次元のパターンを思い浮かべるかもしれないがパターン認識で扱う対象はもっと広い。例えば音声のような時系列信号を処理して五十音や単語に対応させる音声認識もパターン認識の一分野であるし、心電図の波形を分析して異常のあるなしを判定するのも同様である。また視覚や聴覚だけでなく、嗅覚や触覚など様々なセンサを用いて状況を判断するのもパターン認識とすることができる。

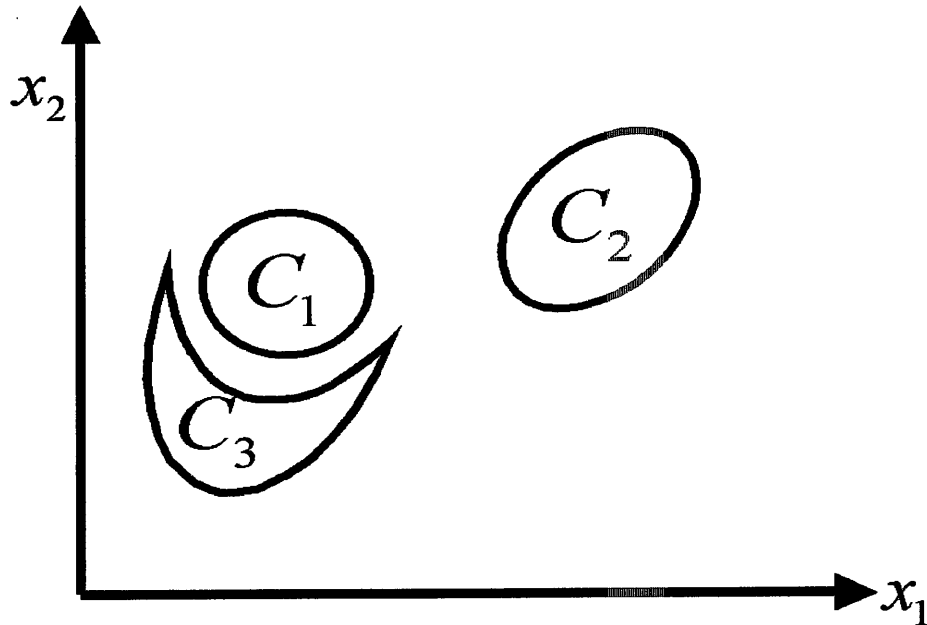


図 1.1: 特徴量の分布の様子 (2次元)

認識のためにはまず原パターンから本質的な特徴を抽出する必要がある。特徴としてはさまざまなものが考えられる。例えば文字認識の場合、文字線の傾き、幅、曲率、面積、ループの数などはしばしば用いられる特徴である。各々の特徴は数値で表され、通常はそれらを組にしたベクトルが用いられる。いま I 個の特徴を用いることにすると、パターンは次式のような I 次元ベクトル \mathbf{x} として表される。

$$\mathbf{x} = (x_1, x_2, \dots, x_I)^T$$

ここで T は転置を表す。上のベクトルを特徴量、特徴量が定義されている I 次元空間を特徴空間と呼ぶ。したがって1つのパターンは特徴空間上での1点として表される。また対象としているパターンのクラスの総数を K とし、各クラスを c_1, c_2, \dots, c_K で表す。同一クラスに属するパターンは互いに類似しているのが一般的であり、特徴空間上でパターンは図 1.1 のようにクラスごとに空間的に局在して分布し観測されることが多い。また他にもパターン認識で用いられる特徴としては数種類考えられていて、例えばクラスごとに特徴量が特徴空間の中の非常に次元の小さい部分空間に偏って分布するものもある [1]。

本論文では図 1.1 のようにクラスごとに空間的に局在して分布し観測される特徴

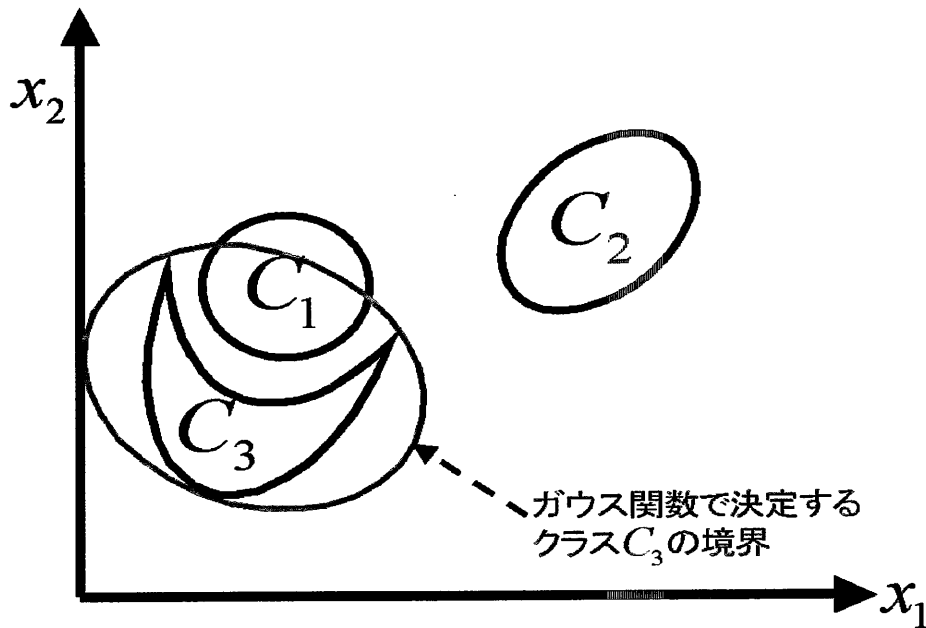


図 1.2: ガウス分布で特徴量分布を近似することの問題点

量を用いることを前提とする。そして本論文で述べるパターン認識とはこの特徴空間における距離でクラス分けを決定するというものである。

1.1.2 パターン認識の問題点

一般的なパターン認識では、特徴量の分布をガウス分布で表現し、識別に必要な特徴空間内での距離及び境界を定める。ガウス関数で記述される距離及び境界面は、多次元特徴空間内で構成すると超楕円体を形成する。よって図 1.1 のクラス c_1 やクラス c_2 のように特徴量の分布形状が楕円 (2次元超楕円体) に近い形状をしていると、比較的識別しやすいといえる。

しかしクラス c_3 のように凹領域を持つような複雑な分布形状をしていてクラス c_2 と接近している場合は、クラス c_3 の境界面を楕円で定義すると、このクラス c_3 の境界面がクラス c_1 を内包してしまう (図 1.2)。これはクラス c_1 をクラス c_3 と誤分類してしまう可能性があることを意味する。

このような事例がパターン認識における問題点の一つとして挙げられている。

1.2 階層型ニューラルネットワーク

多層パーセプトロンなどに代表される階層型ニューラルネットワークは、脳(特に視覚系)の情報処理のモデルであり、学習機能を併せもっている。そして代表的なパターン認識システムの一つであるといえる。しかしながら、前節で述べたような、特徴量が複雑な分布形状をもつようなパターン認識問題に対しこの階層型ニューラルネットワークで適応すると以下のような問題点がある。

- (a) 学習が収束しにくい
- (b) パターンの特徴量分布をガウス分布で表現できないため誤認識してしまう
- (c) 複雑な分布を持つ特徴を認識するためには多くの素子を必要とする

これらについての詳しい解説は次章以降で述べる。

1.3 研究の目的

前節までに述べた点をふまえ、本研究の目的は、特徴量が複雑な分布形状をもつパターンを認識を実現するシステムを構築することである。その構築方針としては階層型ニューラルネットワークに関する下記の三つの手法を組合せてシステムを構成していくことを考える。

- (a) モジュール型ニューラルネットワーク
- (b) RBF 出力関数
- (c) 非線形変換

この (a)-(c) の手法は前節で述べた (a)-(c) の問題点にそれぞれ個別に対応している。具体的には (a) ネットワークをモジュール型と呼ばれるものにする事でその学習の収束性を向上させ、(b) ネットワークの出力関数を RBF とすることで特徴量の分布をガウス分布で表現でき、(c) 非線形変換を用いることで特徴量の複雑な分布に対応できる。

1.4 本論文の構成

本論文の構成は以下のようにになっている。

第1章 序論

研究の背景や目的、及び本論文の構成を述べる。

第2章 モジュール型ニューラルネットワーク

パーセプトロンの学習の収束性の問題点を述べ、ネットワークをモジュール型と呼ばれるものにする事で学習の収束性を向上できることを述べる。またこのネットワークの問題点を指摘する。

第3章 RBF 出力素子を用いたモジュール型ニューラルネットワーク

第2章で述べたネットワークの問題点を、RBF 出力素子を用いて特徴量の分布をガウス分布で表現することで改善することを述べる。またこのネットワークの問題点を指摘する。

第4章 非線形変換を用いたモジュール型ニューラルネットワーク

この章では非線形変換をネットワークに導入することで、第3章で述べた問題点を解消する手法を提案する。また実際に実験を通してその有効性を確認する。

第5章 結論

本研究の成果、及び今後の課題について述べる。

第2章

モジュール型 ニューラルネットワーク

2.1 はじめに

前章でも述べた通り、従来の階層型ニューラルネットワークでは複雑に分布している特徴量に対し、学習の収束性に問題があるといえる。本章ではまず代表的な階層型ニューラルネットワークである3層パーセプトロンに対する収束性の問題を紹介する。その後、3層パーセプトロンの収束性の問題点を解消するモジュール型ニューラルネットワークについて紹介する。

2.2 3層パーセプトロンについて

パーセプトロンは、コーネル大学の心理学者 Rosenblatt[4] によってその原型が提案された階層型ニューラルネットワークである。Rosenblattが提案したパーセプトロンは、初めて脳の神経回路網理論を具体化したモデルとしてだけでなく、世界初の学習する機械としてパターン認識の分野等で大きな注目を集めたが、Minsky[5]らによりその学習能力の限界が指摘され、実際の工学的な応用までは至らなかった。しかし、後に Rumelhart[6]らが、Rosenblattのモデルを拡張し新たな学習則を適用することで、Minskyらが指摘した問題に対しても良い近似解を与

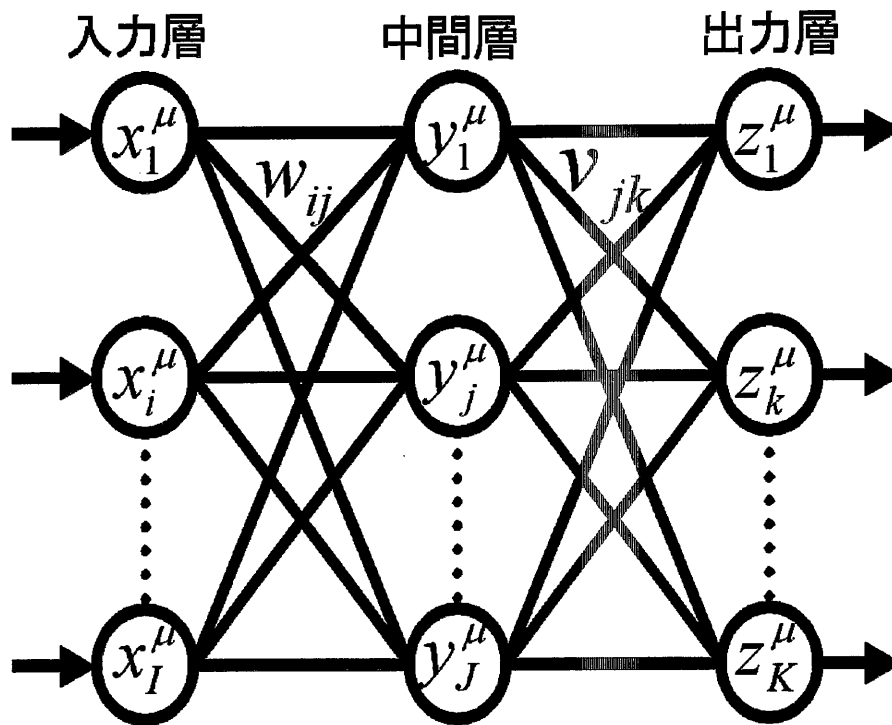


図 2.1: 3層パーセプトロン

えることを示し、パーセプトロンは再び注目を集めることとなった。現在では、パターン認識や時系列予測といった多くの分野への適用が行われ、神経回路網理論のみならず、工学的意味においても非常に重要な位置を占める階層型数理モデルの一つとして認知されている。しかしながら、実際の工学的な応用の観点から捉えたと、Rumelhartらが提案した学習則による学習過程、及び学習後に示す汎化性には未だ幾つかの問題点が残されている。本論文で問題視する、学習の収束性もそのうちの一つである。

ここではそのパーセプトロンの中で最も基本的な3層パーセプトロンについて解説し、学習における収束性の問題点について述べる。

2.2.1 構造

このニューラルネットワークは3層から構成されている。(図2.1)

第1層は入力層である。ここには入力として、パターンの特徴量 x_i^μ を与える。ここで μ は学習、あるいは識別に用いるパターンの番号である。例えば P 個の学習パターンの特徴量を $x_i^\mu, \mu = 1, 2, \dots, P$ と記す。 i は入力層の素子の番号で、特徴

量の各次元に対応している。第1層と第2層の素子間は、全結合となっていて、各結合にはシナプスと呼ばれる重み w_{ij} が与えられている。

第2層は中間層である。パターン μ に対する中間層の各素子の出力を y_j^μ と定義する。 j は中間層の素子の番号である。素子数は J とする。第2層と第3層の素子間には重み v_{jk} が与えられている。

第3層は出力層で、パターン μ に対する各素子の出力は z_k^μ である。ここでの各素子は識別する各クラスに対応していて、 K クラス識別問題を扱う場合、素子は K 個用意する。特徴量 x_i^μ に対する認識結果 c は $c = \arg \max [z_k]$ で与えられる。

各素子の入出力関係は以下の通りである。

$$y_j^\mu = s \left(\sum_{i=0}^I x_i^\mu w_{ij} \right) \quad j = 1, 2, \dots, J \quad (2.1)$$

$$z_k^\mu = s \left(\sum_{j=0}^J y_j^\mu v_{jk} \right) \quad k = 1, 2, \dots, K \quad (2.2)$$

$s()$ はシグモイド関数である。

$$s(a) = \frac{1}{1 + e^{-a}} \quad (2.3)$$

2.2.2 学習法

学習とはネットワークが適切な値を出力するように、重みを更新させていくことである。通常、パーセプトロンでは Rumelhart[6] らによって提案された誤差逆伝搬法と呼ばれる学習法が採用されている。本論文でも学習は全てこの誤差逆伝搬法で行う。この学習法ではまず、得られた出力値から式(2.4)に示す平均二乗誤差を考える。

$$E = \frac{1}{2P} \sum_{k,\mu} (z_k^\mu - \hat{z}_k^\mu)^2 \quad (2.4)$$

ここで \hat{z}_k^μ は教師信号であり、本論文では

$$\hat{z}_k^\mu = \begin{cases} 0.95 & (x^\mu \text{が対応するクラスの時}) \\ 0.05 & (x^\mu \text{が対応するクラスではない時}) \end{cases} \quad (2.5)$$

とした。

誤差逆伝搬法とは E を最小にする w_{ij} 、 v_{jk} を求めるために w_{ij} 、 v_{jk} を微量ずつ変化させる逐次近似法である。その更新量を Δw_{ij} 、 Δv_{jk} とすると

$$\Delta w_{ij} = -\lambda_w \frac{\partial E}{\partial w_{ij}} \quad (2.6)$$

$$\Delta v_{jk} = -\lambda_v \frac{\partial E}{\partial v_{jk}} \quad (2.7)$$

のように求める。ここで λ_w, λ_v は w_{ij}, v_{jk} それぞれの学習係数である。誤差逆伝搬法で正しく学習が行われることはすでに証明されている [6]。

例として、いま仮に正しく学習が収束した場合に、クラス 1 に属する特徴量をその 3 層パーセプトロンに入力したとする。するとそのパーセプトロンにおける出力層のクラス 1 に対応する素子のみが 0.95 かそれに近い値を出力し、それ以外の出力素子は 0.05 かそれに近い値を出力することになる。そこで入力データはクラス 1 に属すると認識される。

2.2.3 学習の問題点

上記学習法の問題点は式 (2.4) で表される平均二乗誤差 E の定義にある。この場合、素子番号 k について和をとっているが、これはすべての出力素子の誤差を合計していることになる。従ってすべてのクラスの誤差をもとに一つの重みの更新量を式 (2.6)、式 (2.7) により計算している。このようにネットワーク全体の平均二乗誤差を全ての重みの更新で共用するので、各クラスの誤差情報が学習の際に混信することになり、そのために収束が遅くなるという問題が考えられる。次節から紹介するモジュール型ニューラルネットではこの問題を解消している。

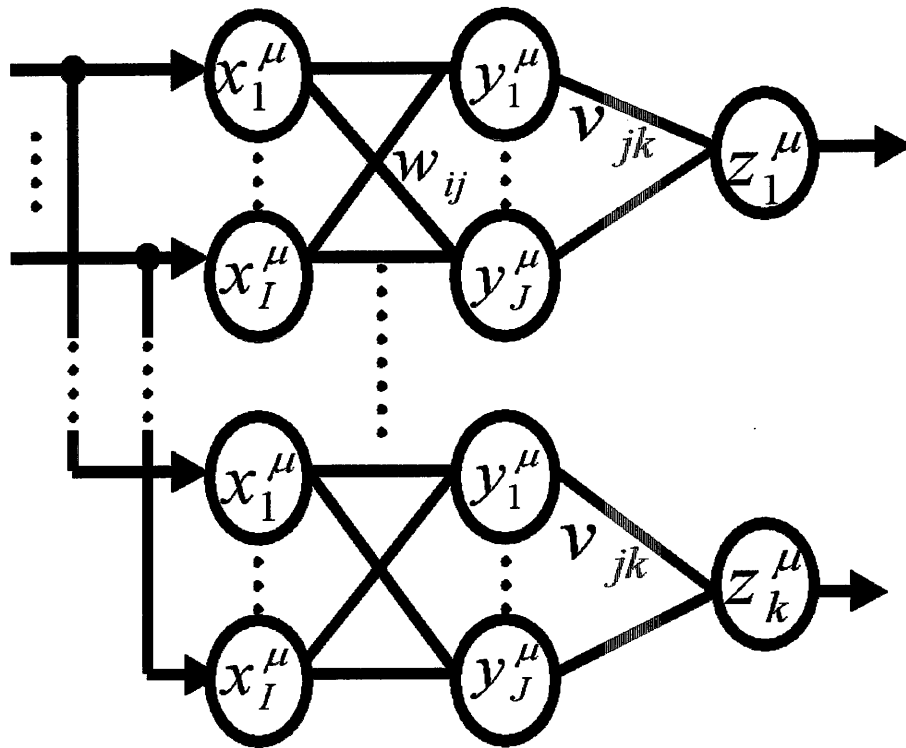


図 2.2: モジュール型ニューラルネットワーク

2.3 モジュール型ニューラルネットワークについて

モジュール型ニューラルネットワーク [8] はパーセプトロンのような階層型ニューラルネットワークの構造を改良したもので K クラス識別問題を K 個の 2 クラス識別問題に置き換える目的で提案された。このような構造を採用することにより、より速く学習が収束するということを説明する。

2.3.1 構造

図 2.2 のようにモジュール型ニューラルネットワークはモジュールと呼ばれる小型の 3 層パーセプトロンを、識別するクラス数だけ用意し、共通の入力をもつ構造となっている。

図 2.3 は 1 つのモジュールである。各モジュールにおいてそれぞれ入力共通であるが出力素子、中間層、それぞれの重みは各モジュールで独立である。1 つのモジュールに担当する 1 つのクラスが割り当てられる。そして入力された特徴量

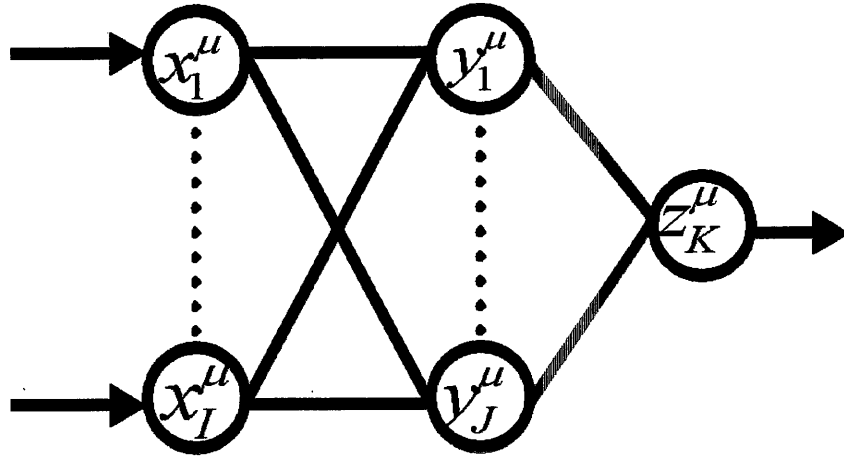


図 2.3: 1つのモジュール

がその担当しているクラスかそれ以外のクラスかを各モジュールで独立に、そして並列に学習する。つまり1つのモジュールは担当するクラスかそれ以外のクラスであるかという2クラス問題の学習を行うことになる。よって出力素子は1つだけ用意される。

入出力関係は3層パーセプトロンの場合と変わらない。つまり式(2.1)から式(2.3)までが各モジュールにおいて独立に計算されることになる。

2.3.2 学習法

モジュール型ニューラルネットワークの場合も誤差逆伝搬法で学習を行う。しかしこの場合、識別するクラスの数だけ独立した中間層や重みが用意されるので平均二乗誤差は各クラスで独立に考える。図2.2の例でクラス1に対するパターンの学習についての式を求める。まず

$$E_1 = \frac{1}{2P} \sum_{\mu} (z_1^{\mu} - \hat{z}_1^{\mu})^2 \quad (2.8)$$

のようにクラス1についての平均二乗誤差 E_1 を考える。次に式 (2.9)、式 (2.10) を適応して各重みの更新量を計算する。

$$\Delta w_{ij} = -\lambda_w \frac{\partial E_1}{\partial w_{ij}} \quad (2.9)$$

$$\Delta v_{jk} = -\lambda_v \frac{\partial E_1}{\partial v_{jk}} \quad (2.10)$$

この学習法の場合、1つの重みは担当する1つのクラスの誤差をもとに更新量が決定されるので、前述のパーセプトロンと比べて速く収束するといえる。

表1 “Fisher’s Iris Data” の認識に要した重みの更新回数

	重みの平均更新回数
(i) モジュール型	5 1 3. 3回
(ii) ノンモジュール型	8 3 1. 2回

また表1は Fisher’s Iris Data という識別問題について (i) モジュール型ニューラルネットワークと、(ii) 通常3層パーセプトロンでの学習に要した更新回数の比較である。Fisher’s Iris Data は3種類の花を識別する問題であり、各学習データは4次元の特徴量で、各クラス50の学習データをもつ3クラス識別問題である。2つのネットワークで平均二乗誤差が0.01以下になるまで重みの更新を繰り返したものである。10回実験を行い、重みの平均更新回数を求めた。

結果はモジュール型ニューラルネットワークの方が少ない更新回数で収束している。このことからモジュール型ニューラルネットワークの方が学習の収束性に関して有効であるといえる。

2.3.3 モジュール型ニューラルネットワークを用いる利点

ここでモジュール型ニューラルネットワークを導入する利点についてまとめておく。前節まで述べた通り、クラス毎に平均二乗誤差を考え、独立に学習が行われるので、重みが収束しやすくなるということが1つである。この研究の目的である複雑な特徴量の分布形状をもつパターンを認識するために、次章以降、さらにネットワークにいろいろな手法を組み合わせるが、そのようにネットワー

クを複雑化すると、通常の階層型ニューラルネットワーク構造のままでは学習の収束性に不安が残る。そこでこのモジュール型ニューラルネットワークの構造を採用することにより、ネットワークを複雑化しても良好に収束するという期待が持てる。

またモジュール型ニューラルネットワークを用いると、全ての問題を2クラス問題に置き換える事ができる。そしてその2クラス問題とはある1つの担当するクラスであるか否かの2クラス問題であるので、今後はいかにその1つのクラスの特徴量の空間分布を適切に表現し認識できるかという議論となる。このこともモジュール型ニューラルネットワークを用いる利点の1つといえる。

2.4 シグモイド型出力関数の問題点

パターン認識問題に対して、前節までで述べたようなシグモイド関数を出力関数として用いるモジュール型ニューラルネットワークの各モジュールでは、対応する1つのクラスとその他のクラスとが中間層の出力空間において線形分離可能な状態となるように学習が行われる。この場合、中間層の出力空間における分離境界面から得られる情報は、その面を境に対応するクラスとその他のクラスが分離できるということだけである。つまり、そのモジュールが担当するクラスについて、中間層の出力空間における位置や分布形状をそのモジュールの重みから直接知ることは困難である。

図2.5は図2.4のモジュールにおける中間層の出力空間を表している。黒点の1つ1つは各クラスの特徴量を表していて $c_1 \sim c_4$ の各クラス毎に塊となっている。図2.4のモジュールは c_1 とその他のクラスを分離する役割が割り当てられているとする。そうするとこのモジュールは学習により図2.5の破線で示すような分離境界を中間層の出力空間において実現する。モジュールの重み v_1, v_2 が破線の傾きに対応している。

このモジュールでは、中間層の出力空間において図の境界線より、 c_1 側かそうでないかの線形分離を行うだけである。境界線より、 c_1 側であればすべて c_1 と認識してしまうので、このネットワークでは c_1 の本当の分布形状を表現することは不可能であり、より正確な認識ができないといえる。

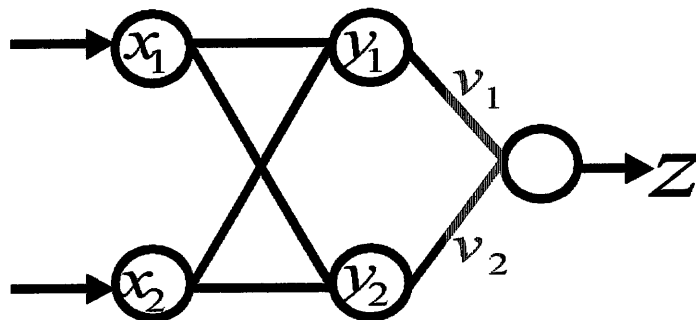


図 2.4: 入力特徴量 2 次元、中間層の素子 2 つのモジュール

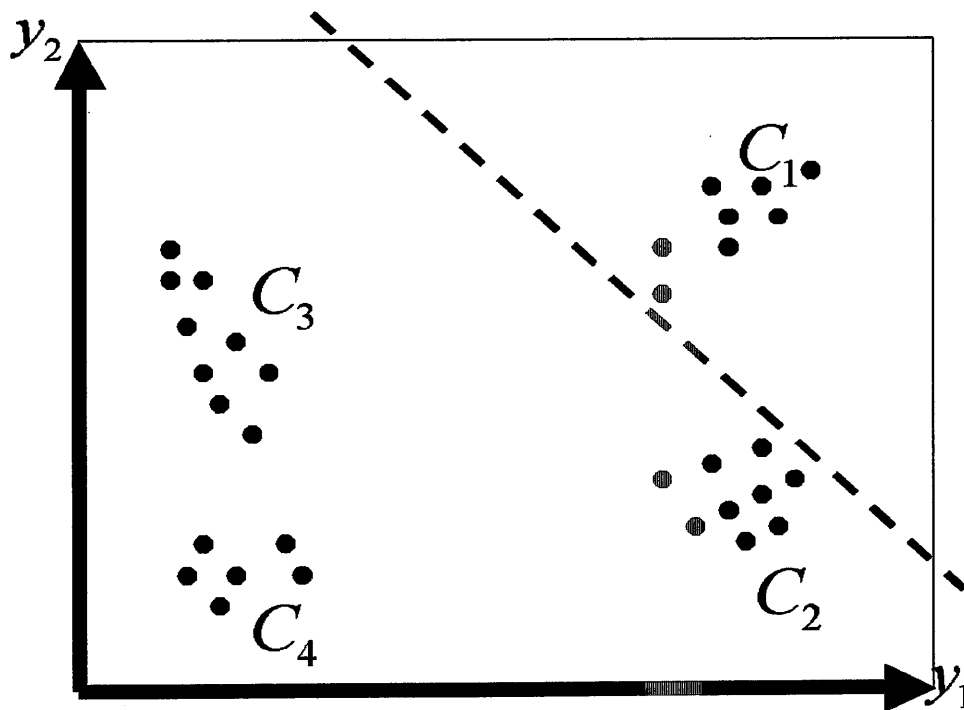


図 2.5: 中間層の様子 (横軸 y_1 、縦軸 y_2)

第3章

RBF 出力素子を用いた モジュール型 ニューラルネットワーク

3.1 はじめに

本章では、まず第2章で述べたモジュール型ニューラルネットワークの問題点に対し、放射基底関数 (Radial Basis Function:RBF) を出力関数とする出力素子 (RBF 出力素子)[11] を用いた問題の解決方法を述べる。そして具体的に RBF 出力素子を用いたモジュール型ニューラルネットワークの入出力関係や学習方法を説明し、また最後にこのネットワークで考えられる問題点を指摘する。

3.2 RBF 出力素子を用いたモジュールの分離境界面

第2章で述べたモジュール型ニューラルネットワークの問題点に対する改善策として各モジュールの出力層に RBF 出力素子を使用する方法が提案されている [11]。RBF 出力素子を用いたモジュール型ニューラルネットワークの各モジュールでは、中間層の出力空間において、担当する一つのクラスとその他すべてのクラスに対し、超楕円体状の分離境界面による分離が可能な状態となるように学習が行われ

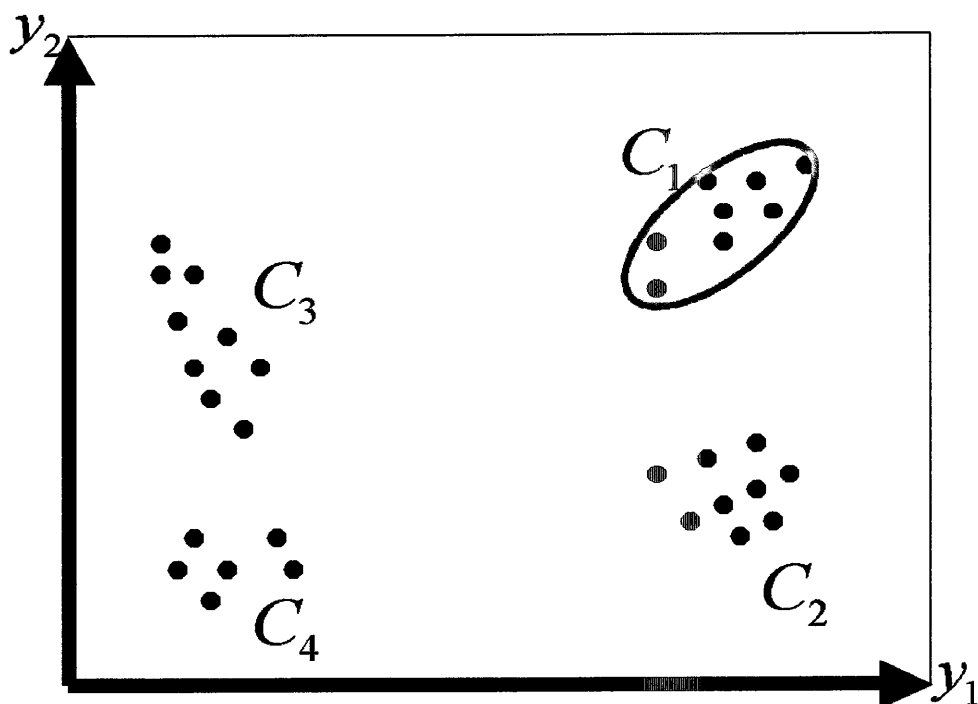
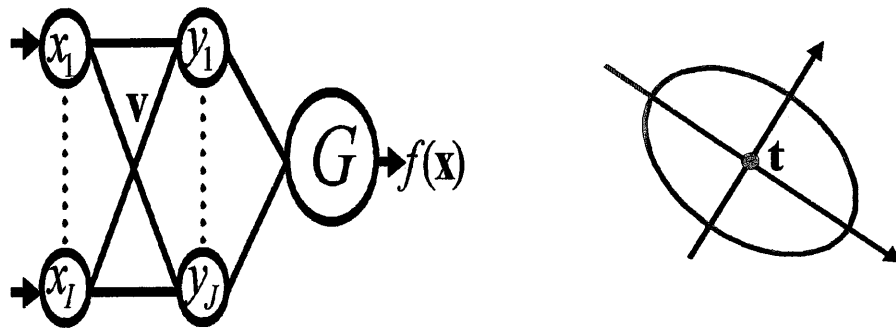


図 3.1: RBF 出力素子を用いたモジュールの分離境界面 (楕円状)

る (図 3.1)。この場合、中間層の出力空間における分離境界面から得られる情報は、その面を境に担当するクラスとその他のクラスが分離できるということだけではない。この分離境界面には、中間層の出力空間における担当するクラスの中心位置や分布形状についての情報が含まれている。またシグモイド関数のみを出力関数として用いるモジュールは学習により分離を行う境界線の傾きを求めていたことに対し、RBF 出力素子を用いたモジュールは分離境界面を形成する超楕円体の中心位置、傾き、広がりをも学習により得ている。よってシグモイド関数のみを出力関数として用いる場合よりも適切に分離境界面を形成することが可能となる。

3.3 RBF 出力素子を用いたモジュールの入出力関係

RBF 出力素子を用いたモジュールの構造を図 3.2(a) に示す。入力層から中間層までの変換は第 2 章までのものと変わらない。ここでは入力層から中間層までの



(a)RBF 出力素子をもつモジュール (b) $f(x)$ が構成する楕円 ($x \in \mathbf{R}^2$ の場合)

図 3.2: RBF 出力素子をもつモジュールと得られる境界の様子

重みを行列 \mathbf{V} とし、各層の素子出力をベクトル \mathbf{x} 、 \mathbf{y} で表記する。

$$\mathbf{x} = (x_1, x_2, \dots, x_I) \tag{3.1}$$

$$\mathbf{y} = (y_1, y_2, \dots, y_J) \tag{3.2}$$

$$\mathbf{v} = \begin{pmatrix} v_{1,1} & \cdots & v_{1,J} \\ \vdots & \ddots & \vdots \\ v_{I,1} & \cdots & v_{I,J} \end{pmatrix} \tag{3.3}$$

$$\mathbf{y} = s(\mathbf{xv}) \tag{3.4}$$

次に中間層から出力層にかけてであるが、RBF 出力素子の出力 $f(x)$ は

$$G(\|\mathbf{y} - \mathbf{t}\|_{\mathbf{w}}^2) = \exp(-(\mathbf{y} - \mathbf{t})\mathbf{w}^T\mathbf{w}(\mathbf{y} - \mathbf{t})^T) \tag{3.5}$$

を用いて

$$f(\mathbf{x}) = G(\|s(\mathbf{xv}) - \mathbf{t}\|_{\mathbf{w}}^2) \tag{3.6}$$

となる。ここで G はガウス関数、 \mathbf{t} はその中心位置ベクトル \mathbf{w} はその広がりや傾きを決める重み行列で、それぞれ次のように表せる。

$$\mathbf{t} = (t_1, t_2, \dots, t_J) \tag{3.7}$$

$$\mathbf{w} = \begin{pmatrix} w_{1,1} & \cdots & w_{1,J} \\ \vdots & \ddots & \vdots \\ w_{J,1} & \cdots & w_{J,J} \end{pmatrix} \quad (3.8)$$

3.4 RBF 出力素子を用いたモジュールの学習

学習によって更新されるパラメータは重み \mathbf{v} 、ガウス関数の中心位置 \mathbf{t} 、傾きと広がり \mathbf{w} の3種類である。そして第2章のときと同様に誤差逆伝搬法で学習を行う。平均二乗誤差 E を

$$E = \frac{1}{2P} \sum_{\mu} (f^{\mu}(\mathbf{x}) - \hat{f}^{\mu}(\mathbf{x}))^2 \quad (3.9)$$

とする。ここで $\hat{f}^{\mu}(\mathbf{x})$ は μ 番目の特徴量に対する教師信号を表す。更新則は次式で与えられる。

$$\Delta \mathbf{v} = -\lambda_{\mathbf{v}} \frac{\partial E}{\partial \mathbf{v}} \quad (3.10)$$

$$\Delta \mathbf{t} = -\lambda_{\mathbf{t}} \frac{\partial E}{\partial \mathbf{t}} \quad (3.11)$$

$$\Delta \mathbf{w} = -\lambda_{\mathbf{w}} \frac{\partial E}{\partial \mathbf{w}} \quad (3.12)$$

3.5 数値実験

前節までで、RBF 出力素子を用いたモジュールが楕円状の分離境界面を形成することについて述べてきたが、実際に数値実験を行って認識率が向上することを確認する。

表2 “Glass Data” の認識

	学習データ認識率	テストデータ認識率
(i) シグモイド関数	96.2%	81.5%
(ii) RBF 出力関数	97.4%	82.9%

Glass Data 識別問題について (i)RBF 出力素子を用いる場合と、(ii)シグモイド関数のみを出力関数とした場合で認識実験を行った。Glass Data はガラスを識別する問題でクラス数は6、特徴量は10次元で、それぞれ107の学習データと107のテストデータがある。結果は表2の通りでRBF 出力素子を用いたモジュール型ニューラルネットワークにより認識率の向上が確認できた。

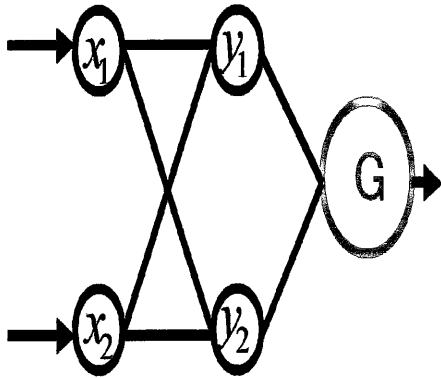
よってこの結果からもRBF 出力素子を用いたモジュールはシグモイド関数のみを用いるものより、優れた分離境界面を形成することができるといえる。また各クラスの特徴量が空間的に局在化しているという本稿の仮定が妥当であることを裏付けるものでもある。

3.6 RBF 出力素子を用いたモジュールの問題点

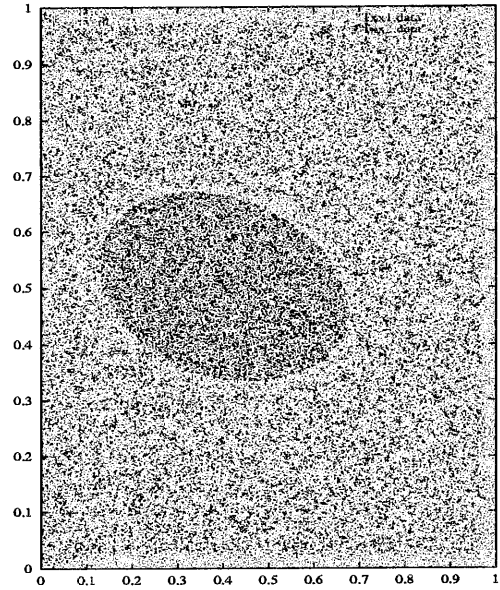
第4章に入る前に、このRBF 出力素子を用いたモジュールに関する問題点を一つ説明する。実際に人工の学習データをこのモジュールに入力し、分離境界を得る課程を示し、このモジュールで認識可能なデータとそうでないデータを説明して、その問題点を指摘する。

3.6.1 RBF 出力素子を用いたモジュールで対応できる場合

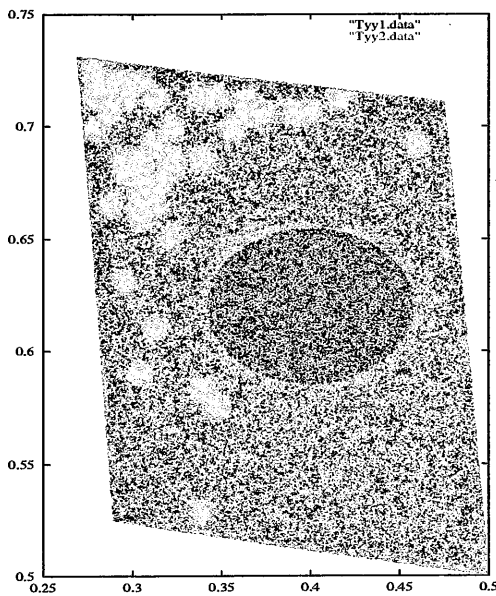
例えば図3.3(a)のモジュールが図3.3(b)のような楕円の形に分布しているような特徴量をもつクラス c_1 とそれ以外 c_2 を分離するような学習を行って得られた場合を考える(特徴量2次元、中間層の素子2つとする)。入力データとして図3.3(b)に示す各点を考え、モジュールに入力すると中間層では線形変換とシグモイド関数による変換で図3.3(c)のように変換される。このときシグモイド関数はある意味で2値化するだけなので、重みをかけるという線形変換がきいていて楕円の形状が大きく変化することはない。そして中間層に写像された c_1 に対しRBF 出力素子で境界線を近似する(図3.3(d))。すなわち特徴量が楕円の形に分布しているので、認識が容易であるといえる。



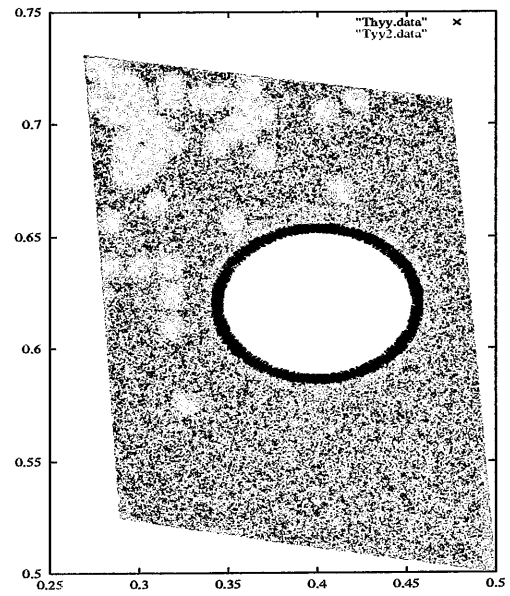
(a) 使用するモジュール



(b) 人工データ 1 c_1 (青)、 c_2 (赤)



(c) 中間層の様子 (横軸 y_1 、縦軸 y_2)



(d) 形成した識別境界

図 3.3: RBF 出力素子を用いたモジュールによる認識 (成功例)

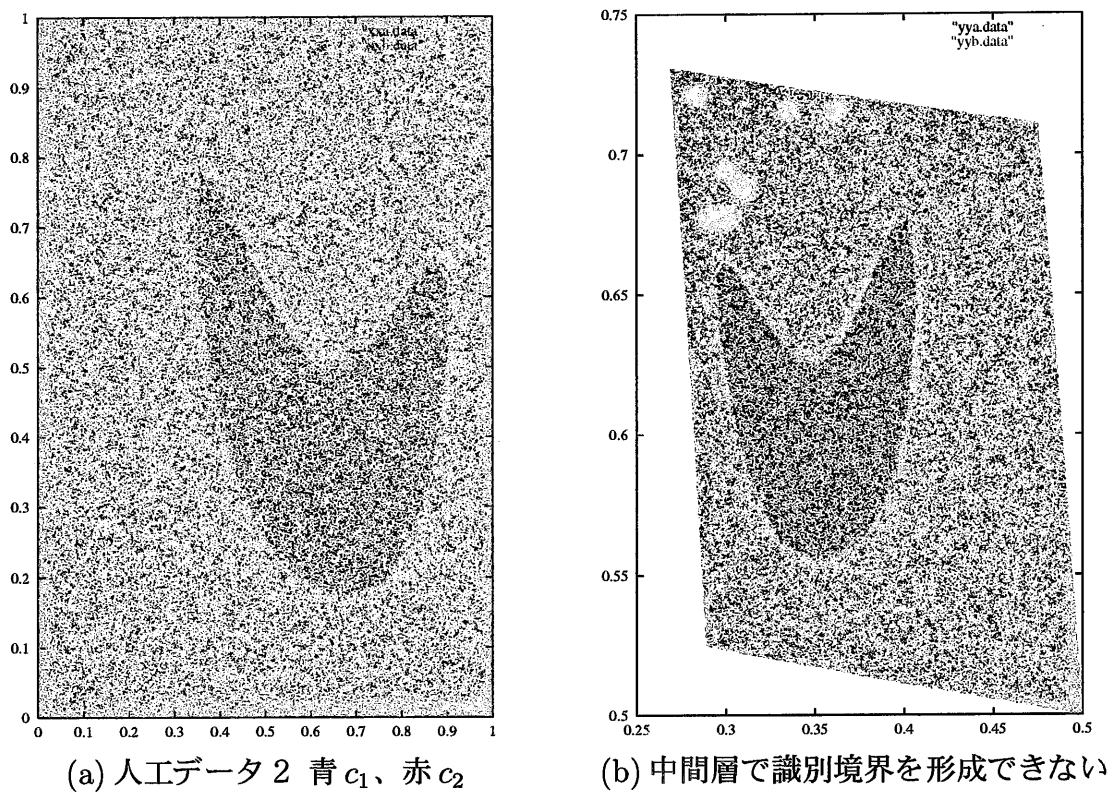


図 3.4: RBF 出力素子を用いたモジュールによる認識 (失敗例)

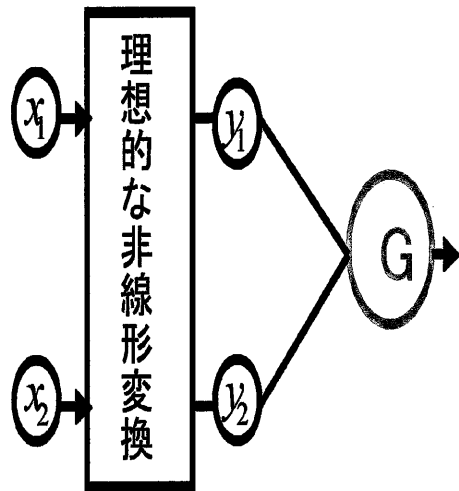
3.6.2 RBF 出力素子を用いたモジュールで対応できない場合

次に前節と同じ図 3.3(a) のモジュールに対し図 3.4(a) のような凹領域をもつような複雑な分布をする特徴量を学習し認識させようとする場合を考える。入力層、中間層の入出力関係から、例えば、入力データとして図 3.4 (a) の各点をモジュールに入力すると中間層では図 3.4 (b) のような形の点集合に変換される。このとき c_1 の分布形状は凹性を保ったままであるので、中間層において、楕円状の識別境界を形成しようとしても、学習が収束しない。すなわち適切な境界を形成することができない。この章で説明した RBF 出力素子を用いただけでは図 3.4 (a) のような複雑な分布形状をもつパターンを認識できないといえる。

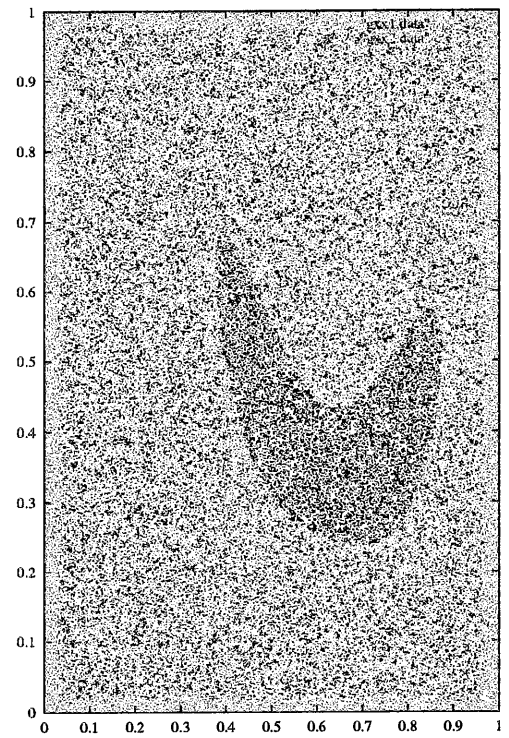
3.6.3 シグモイド関数に代わる理想的な非線形変換について

入力層から中間層にかけての線形変換とシグモイド関数による変換の特性から、図3.4 (a)のような分布形状を認識することはできないと考えられる。そこで入力層から中間層の変換に非線形変換を考える。例えば図3.5 (a)が理想的な非線形変換を行うモジュールであるとする。いま図3.5 (b)のような複雑な分布形状をもつパターン(各点)を識別させようとしたとき、モジュールの入力層から中間層にかけて、理想的な非線形変換が行われ c_1 の分布形状が中間層において楕円の形に変換されていると仮定する(図3.5 (c))。その場合 RBF 出力素子で、図3.5 (d)のように識別境界を形成することができる。

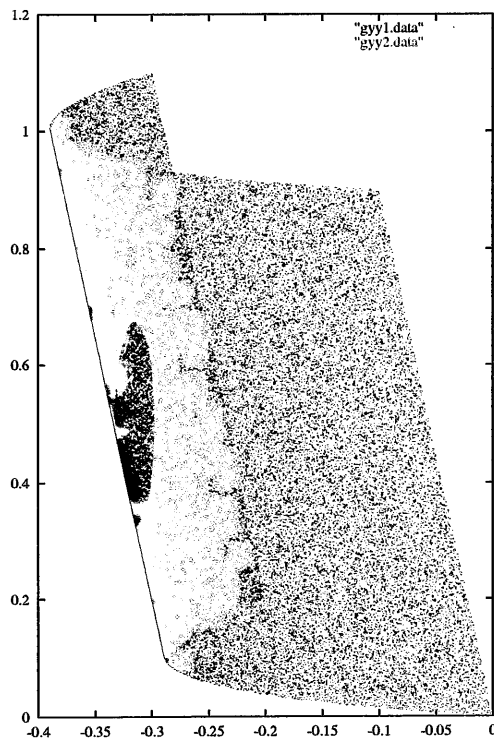
このように入力層から中間層にかけて、従来から適用されている線形変換とシグモイド関数を用いるのではなく、あらゆる複雑な分布形状をもつ特徴量に対し、楕円の形状に変換するような、理想的な非線形変換を考えることで複雑な分布形状をもつパターンを認識できるのではないかと考え、第4章では入力層から中間層にかけての非線形変換について述べる。



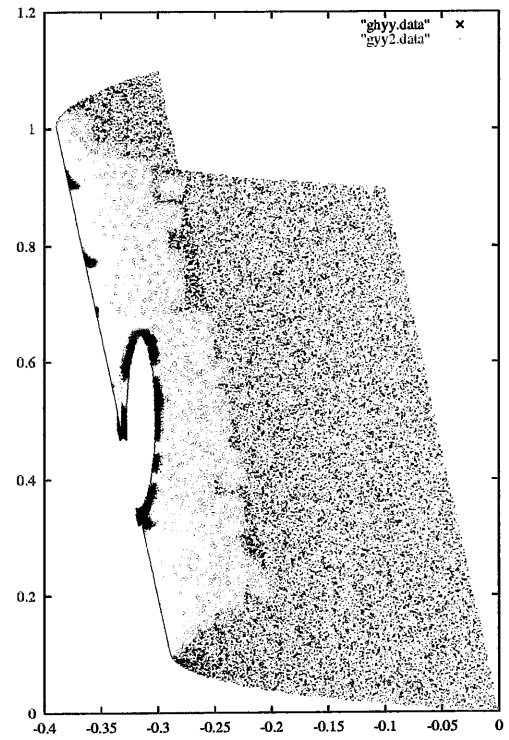
(a) 理想的な非線形変換 (例) を行うモジュール



(b) 人工データ 2 青 c_1 、赤 c_2



(c) モジュールの中間層



(d) 形成される識別境界

図 3.5: 理想的な非線形変換を行うモジュールによる認識

第4章

非線形変換を用いたモジュール型 ニューラルネットワーク

4.1 はじめに

特徴量が複雑な分布形状をもつパターンの認識を実現させるために、入力層から中間層への変換において、これまでと異なる非線形変換を取り入れたモジュール型ニューラルネットワークを提案する。これまでは特徴量に線形変換したものを中間層出力として考えてきたが、今回用いるモジュールでは、中間層出力が入力特徴量の各要素に関する高次多項式で与えられるものを考える。この提案手法において、第3章で説明したRBF出力素子だけに拡張したモジュールでは認識できないような複雑な分布形状の特徴量に対しても、認識が成功することを実験を通して説明する。

4.2 非線形変換を用いたモジュールの構成

図4.1に入力する特徴量が2次元の、非線形変換を用いたモジュールを示す。第3章までのモジュールの場合、 y_1 には、 x_1 、 x_2 に重みを付け加えた線形和を入力しシグモイド関数で出力するが、その方法では第3章でも述べたように、あらゆる分布形状に対して、中間層において分布を楕円で近似するには限界がある。線

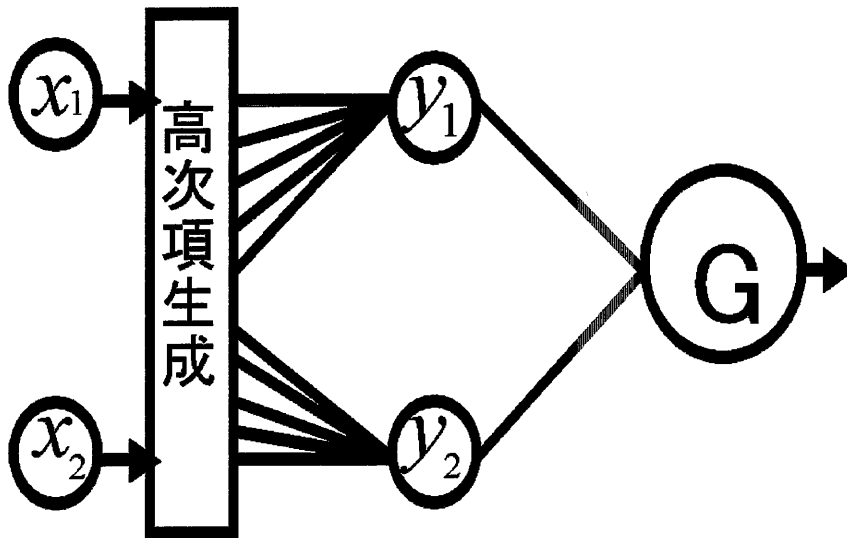


図 4.1: 非線形変換を用いたモジュール

形変換とシグモイド関数による変換では、入力と中間層出力とは1対1に対応している。入力空間が平面状であれば、中間層出力空間も平面状になる。しかし入力部に非線形性をもたせることによって、入力空間から中間層の出力空間への対応は多対1になり、入力空間が平面の場合、中間層出力空間は曲面を射影したようになる(図3.5(c))。この非線形変換によりネットワークで表現できる分布形状の自由度を上げることが可能であると考えられる。またどのくらい非線形性をもたせるか、調節できることも重要である。そこで、この図4.1のモジュールでは、中間層出力の計算のための入力に従来の x_1 、 x_2 に加え、高次項 x_1^2 、 x_2^2 、 x_1x_2 を考慮することで入力部に非線形性をもたせる(今回は高次項は2次まで)。そしてその非線形性を調節するために、各高次項に重みをつけて、学習でこの重みを調節することにする。学習方法はこれまでと同様に誤差逆伝搬法とする。

中間層出力は、例えば図4.1のモジュールにおいて y_1 は

$$y_1 = v_1x_1 + v_2x_2 + v_3x_1^2 + v_4x_2^2 + v_5x_1x_2 \quad (4.1)$$

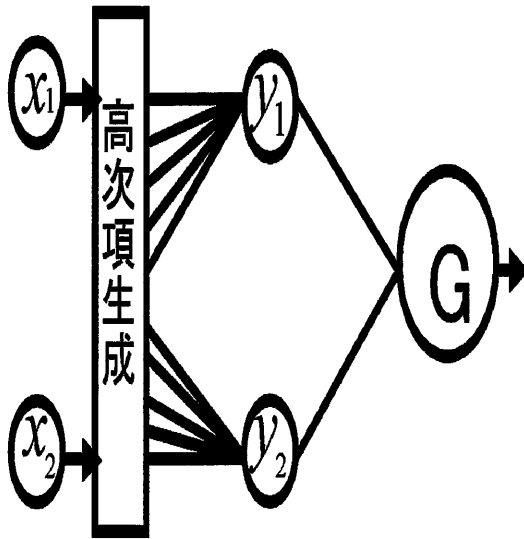
と表される。ここで v_i ($i = 1, \dots, 5$) は各入力に対する重みである。このモジュールの学習はモジュール出力から計算される平均二乗誤差を E として、

$$\Delta v_i = -\lambda_v \frac{\partial E}{\partial v_i} \quad (4.2)$$

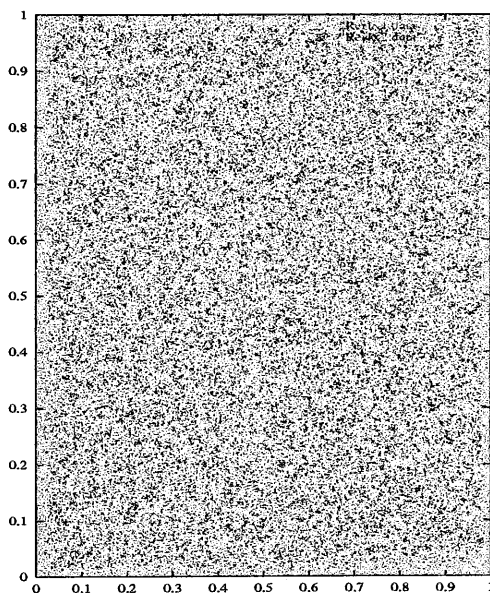
を更新量として学習する。RBF 出力素子の入出力関係や平均二乗誤差の定義は、前章と同様である。

4.3 非線形変換を用いたモジュールの動作

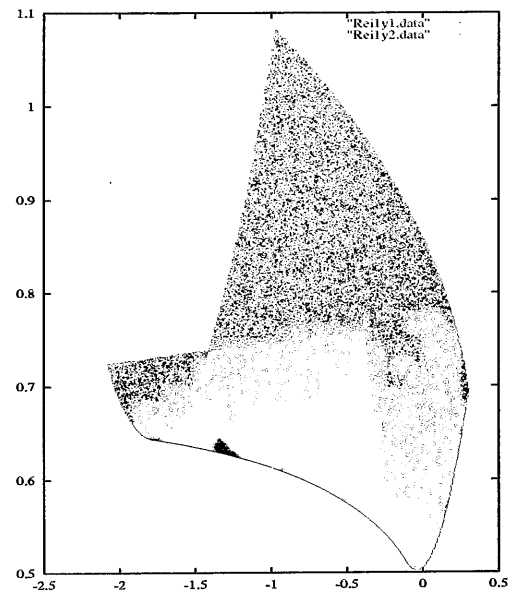
実際に図 4.1 のモジュールを構成し (重みの値は仮に乱数を用いる) 図 4.2(b) のような長方形に一様分布する点を入力すると、中間層における点の分布の様子は図 4.2(c) のようになる。すなわち、入力の各点は中間層には 2 対 1 で写像され、ある曲面が射影された形になることがわかる。ここでやみくもに高次項を増やしていったって、認識実験を行う前に、このような曲面を形成するには、追加した高次項のうちどの要素がどのように作用しているのか、また曲面を形成することにより、じっさいにどのように認識に効果をもたらすのか把握する必要があると考え、実験的に調査することにした。次節に述べる。



(a) 高次項を生成するモジュール

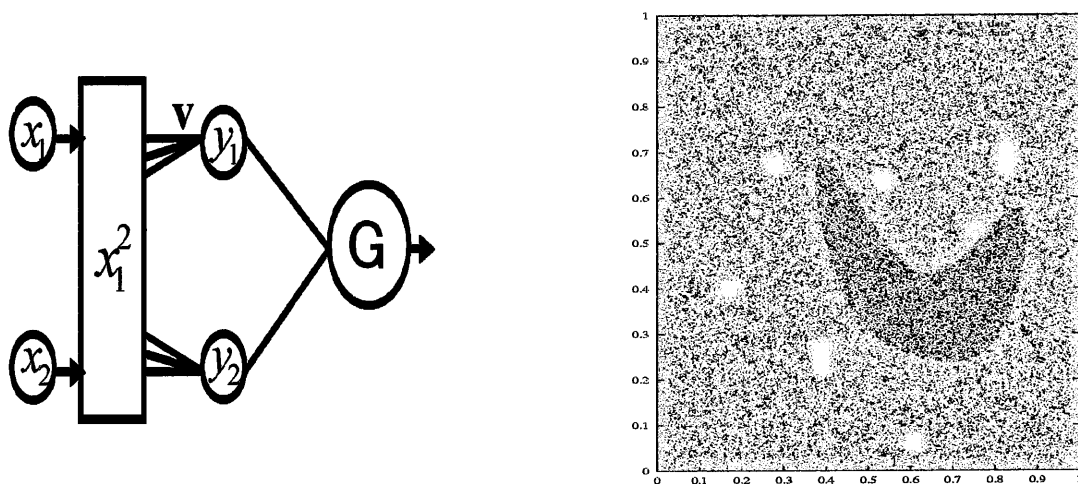


(b) 入力 (一様な乱数 横軸 x_1 縦軸 x_2)



(c) 中間層の様子 (横軸 y_1 縦軸 y_2)

図 4.2: 非線形変換を用いたモジュールの動作



(a) 実験 1-1 に使用したモジュール (b) 認識したデータ (人工データ) 青 c_1 、赤 c_2

図 4.3: 実験 1-1 使用したモジュールと特徴量

4.4 計算機実験

4.4.1 実験 1 非線形変換を用いたモジュールの動作確認

図 4.1 のような高次項を生成するモジュールにおいて、高次項 x_1^2 、 x_2^2 、 x_1x_2 をそれぞれ 1 つずつ使用したとき、中間層がどのように変化するか確認する。またそれによってどのような分布を識別することが可能か調査する。

実験 1-1 x_1^2 を使用 (1)

まず x_1^2 の項のみを使用した。調査結果から述べると 2 次元特徴量 (x_1, x_2) が図 4.3 (b) のような、凹領域の分布形状をもつようなクラス c_1 のパターン (人工データ) を認識できることがわかった。

x_1^2 の項のみを使用すると、図 4.4 (b) にも見られるように、形成された面が面の縦軸を中心として放物線を描くように折れ曲がる傾向があることが調査の結果、確認できた。

この性質を利用して図 4.3 (a) のような分布形状を認識できる。このモジュールでは中間層において、入力された図 4.3 (a) の図面を縦に折曲げるという変換を行い、 c_1 を 1 つの楕円の形に近似している。(図 4.4 (b))。そうなったところで RBF 出力素子により楕円状の識別境界を生成する (図 4.4 (c))。よって入力特徴量に対して図 4.4 (d) のような識別境界が得られ、認識に成功したといえる。

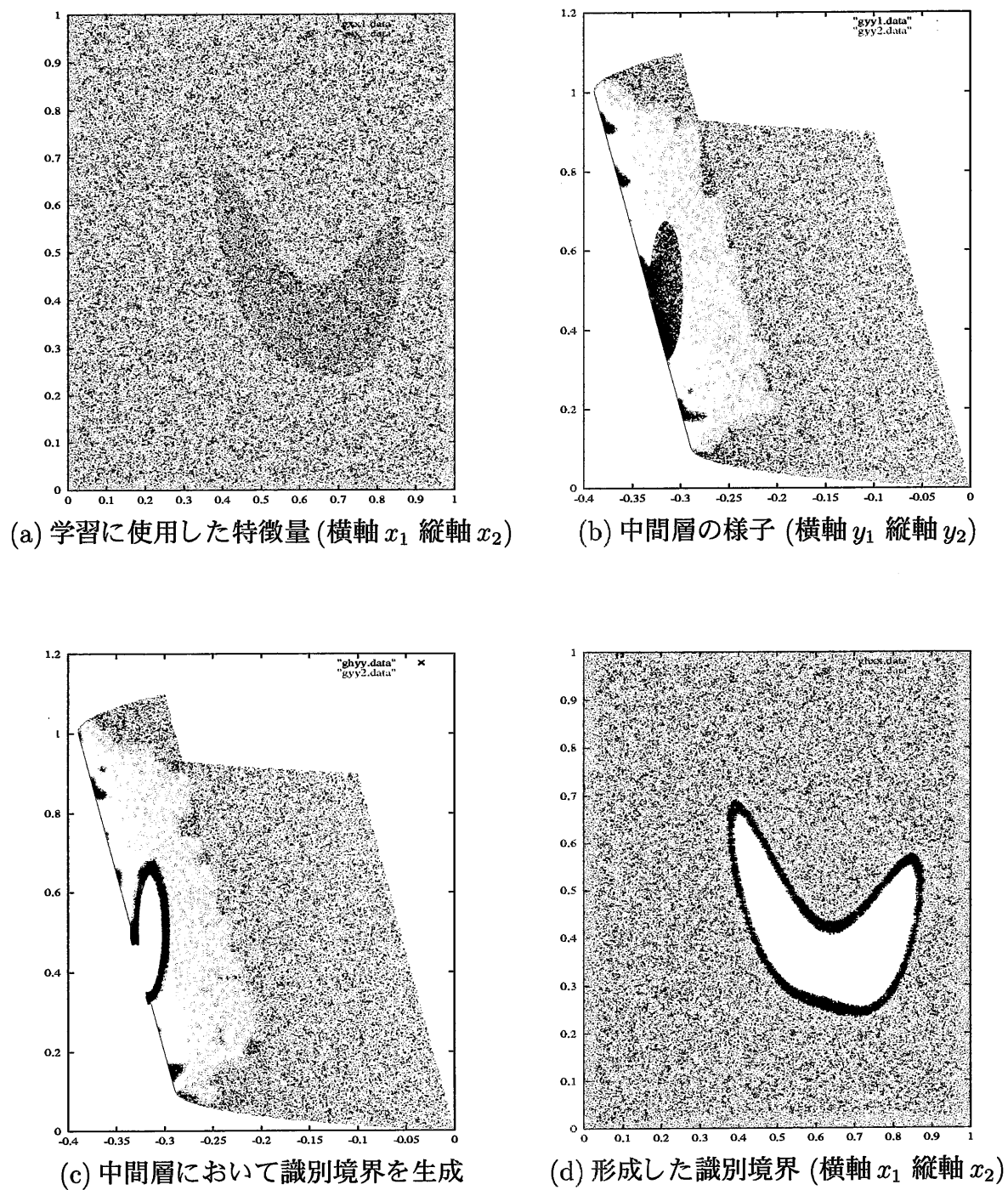


図 4.4: 実験 1-1 認識の様子

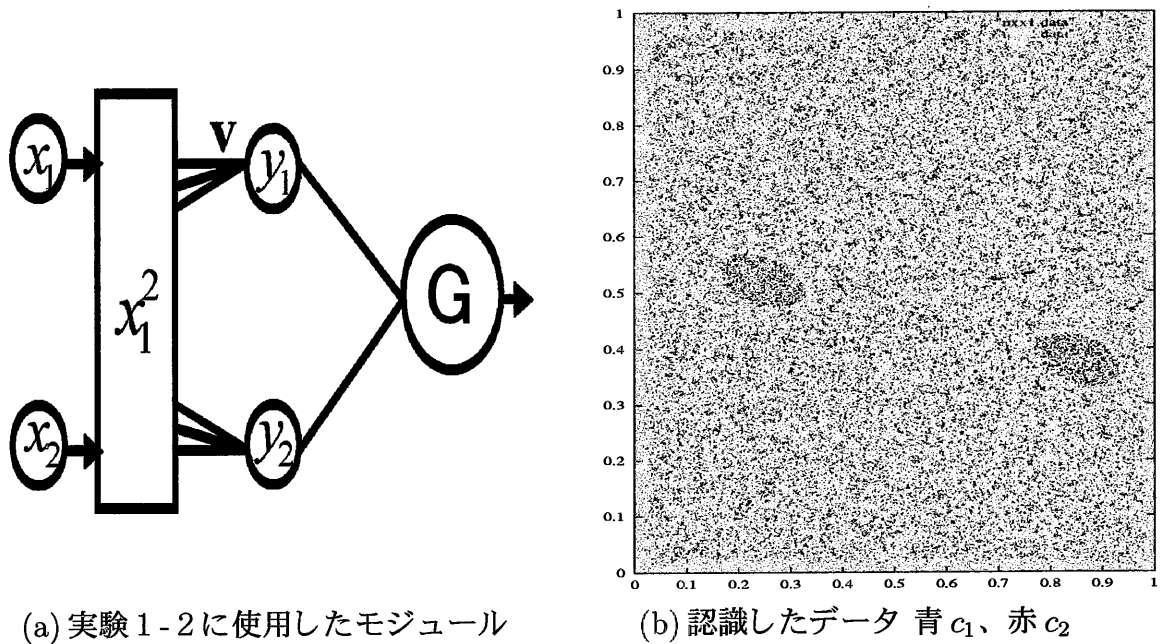


図 4.5: 実験 1-2 使用したモジュールと特徴量

実験 1-2 x_1^2 を使用 (2)

また x_1^2 の項のみを使用したとき図 4.5 (b) のようなひとつのクラスでありながら特徴量が離れた箇所に分布しているようなクラスのパターンを認識できることが確認できた。

x_1^2 の項を使用すると、実験 1-1 でも述べたように形成された面が面の縦軸を中心として放物線を描くように折れ曲がるのでこのモジュールでは入力された図 4.6 (a) の図面を中間層において、縦に折曲げて c_1 を 1 つの楕円の形に近似する (図 4.6 (b))。そうなったところで RBF 出力素子により楕円状の識別境界を生成する (図 4.6 (c))。よって入力特徴量に対して図 4.6 (d) のような識別境界が得られ、認識に成功したといえる。

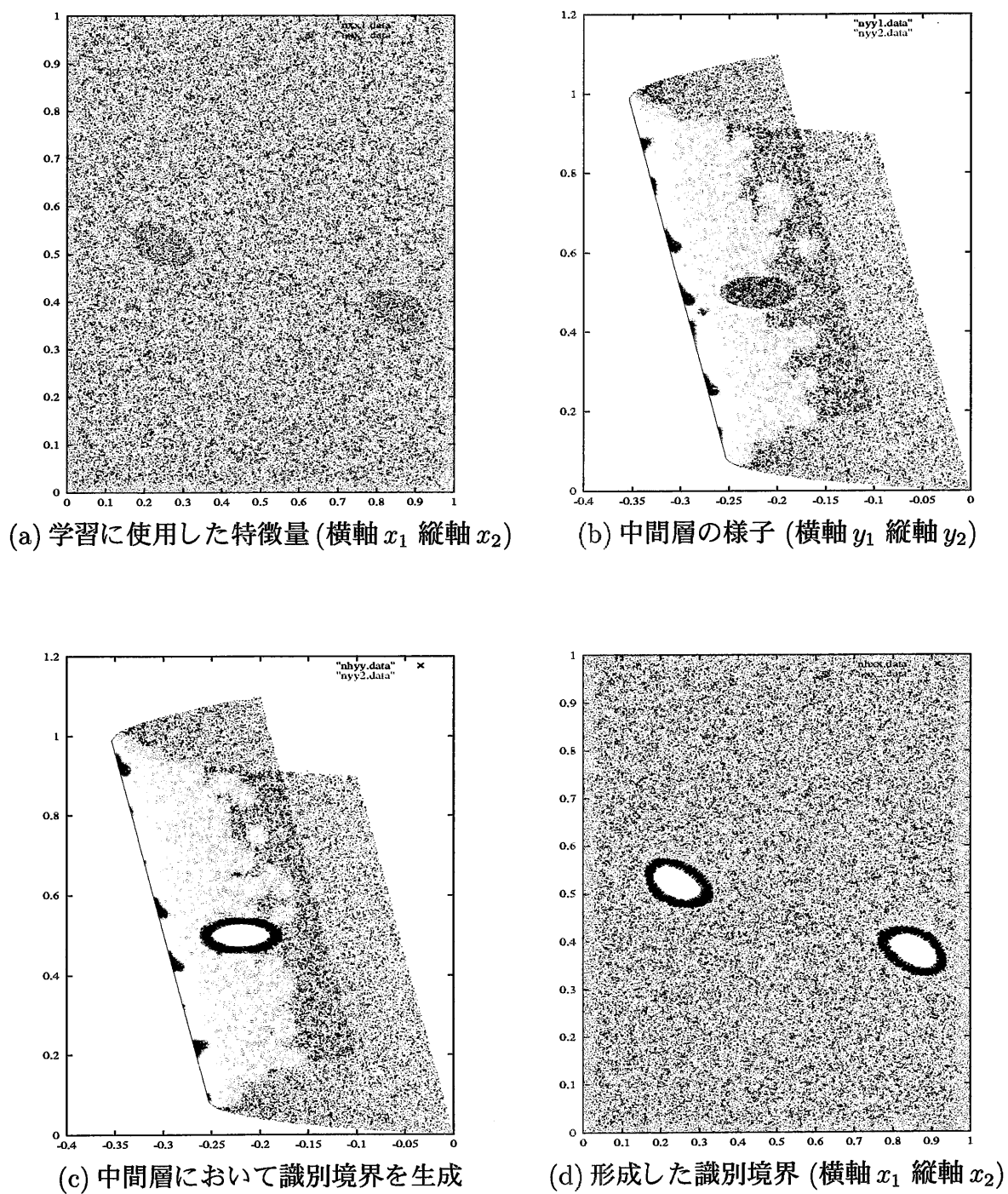


図 4.6: 実験 1-2 認識の様子

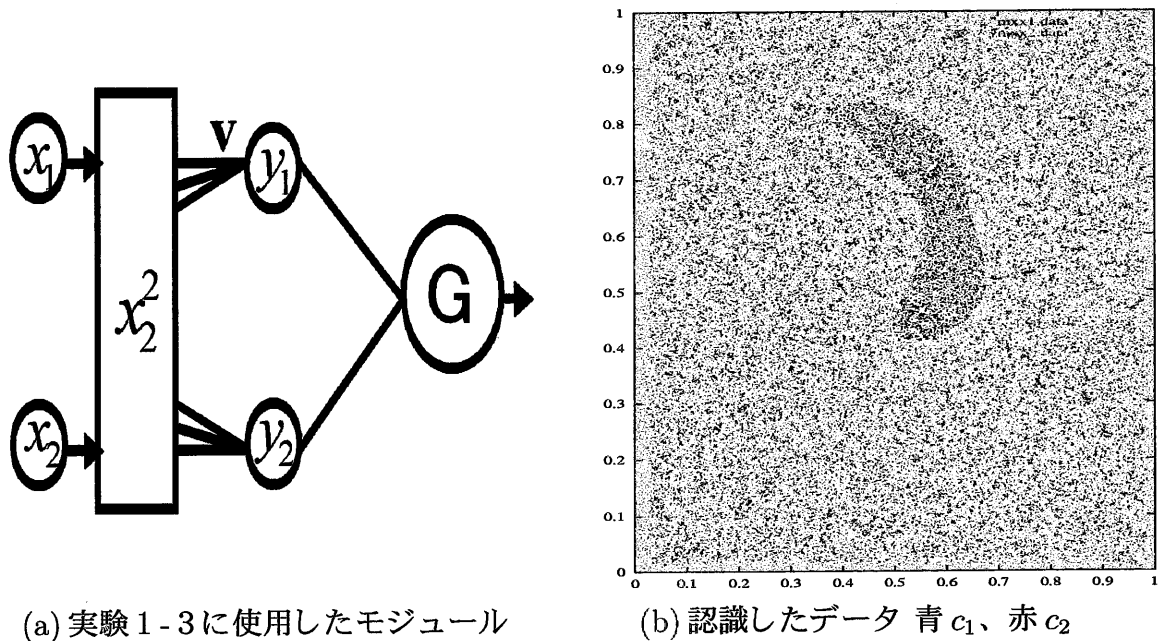
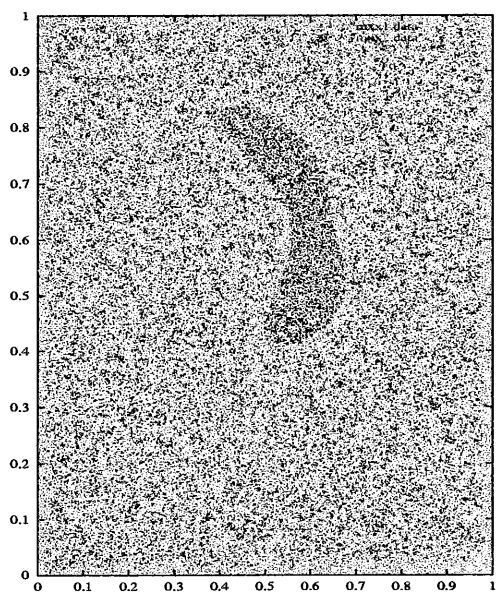


図 4.7: 実験 1-3 使用したモジュールと特徴量

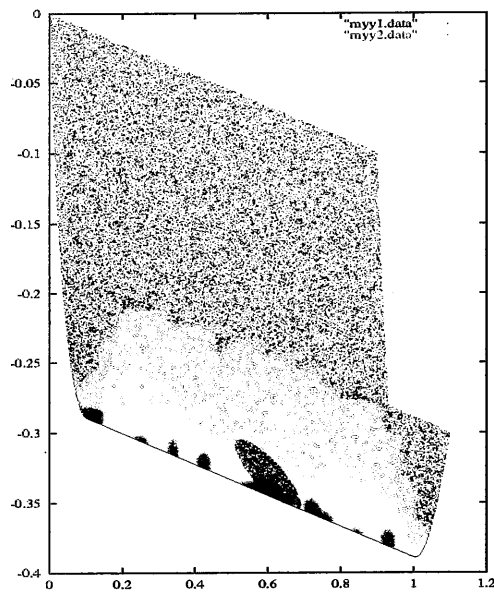
実験 1-3 x_2^2 を使用

また x_2^2 の項を使用したとき図 4.7 (b) のような、実験 1-1 のときと向きを変えたような、特徴量分布が凹領域をもつクラスのパターンを認識できることが確認できた。

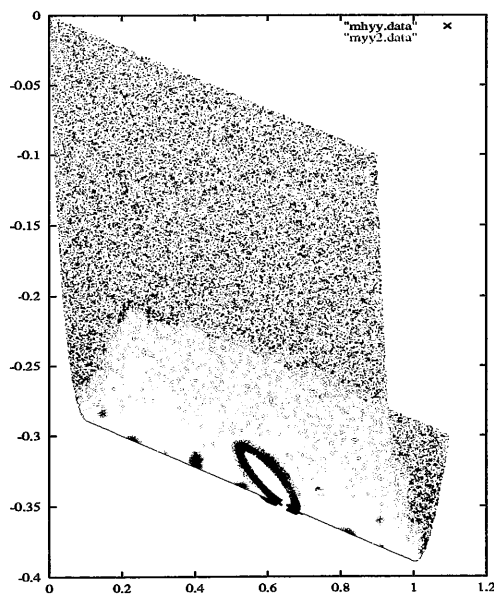
x_2^2 の項を使用すると、図 4.8 (b) にも見られるように形成された面が面の横軸を中心として放物線を描くように折れ曲がる傾向にあることが調査の結果、確認された。このモジュールでは入力された図 4.8 (a) の図面を中間層において、横に折曲げて c_1 を一つの楕円の形に近似する (図 4.8 (b))。そうなったところで RBF 出力素子により楕円状の識別境界を生成する (図 4.8 (c))。よって入力特徴量に対して図 4.8 (d) のような識別境界が得られ、認識に成功したといえる。



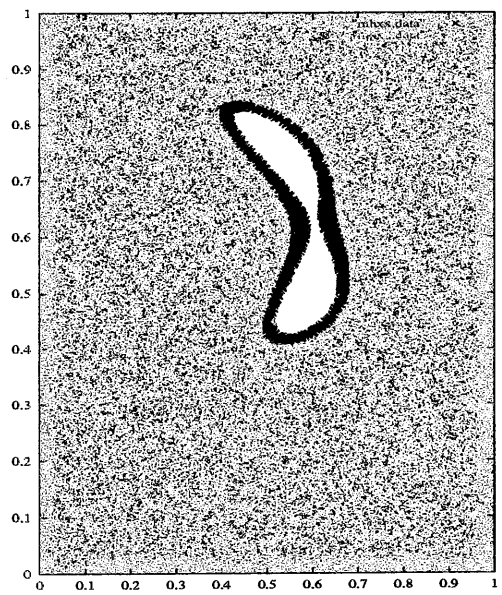
(a) 学習に使用した特徴量 (横軸 x_1 縦軸 x_2)



(b) 中間層の様子 (横軸 y_1 縦軸 y_2)



(c) 中間層において識別境界を生成



(d) 形成した識別境界 (横軸 x_1 縦軸 x_2)

図 4.8: 実験 1-3 認識の様子

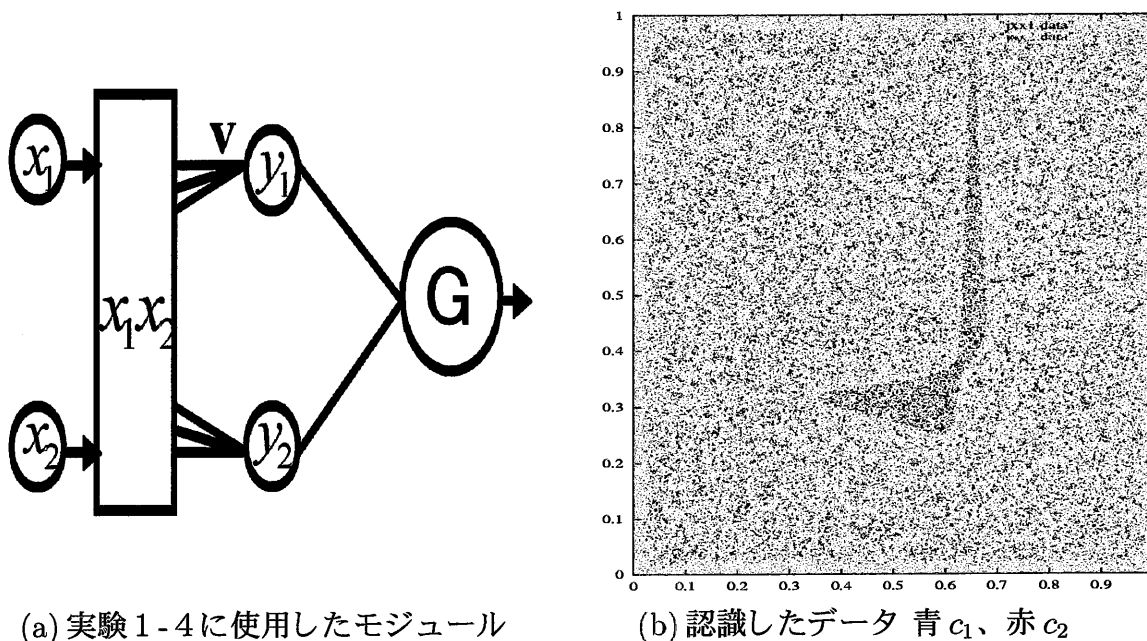


図 4.9: 実験1-4 使用したモジュールと特徴量

実験1-4 x_1x_2 を使用 (1)

また x_1x_2 の項を使用したとき図 4.9 (b) のような、やはり特徴量分布が凹領域をもつ複雑なクラスのパターンを認識できることがわかった。

x_1x_2 の項を使用すると、図 4.10 (b) にも見られるように、形成された面が面の対角線を中心として折れ曲がる傾向にあることが調査の結果、確認できた。

このモジュールでは入力された図 4.10 (a) の図面を中間層において、対角線で折曲げて c_1 を一つの楕円の形に近似する (図 4.10 (b))。そうなったところで RBF 出力素子により楕円状の識別境界を生成する (図 4.10 (c))。よって入力特徴量に対して図 4.10 (d) のような識別境界が得られ、認識に成功したといえる。

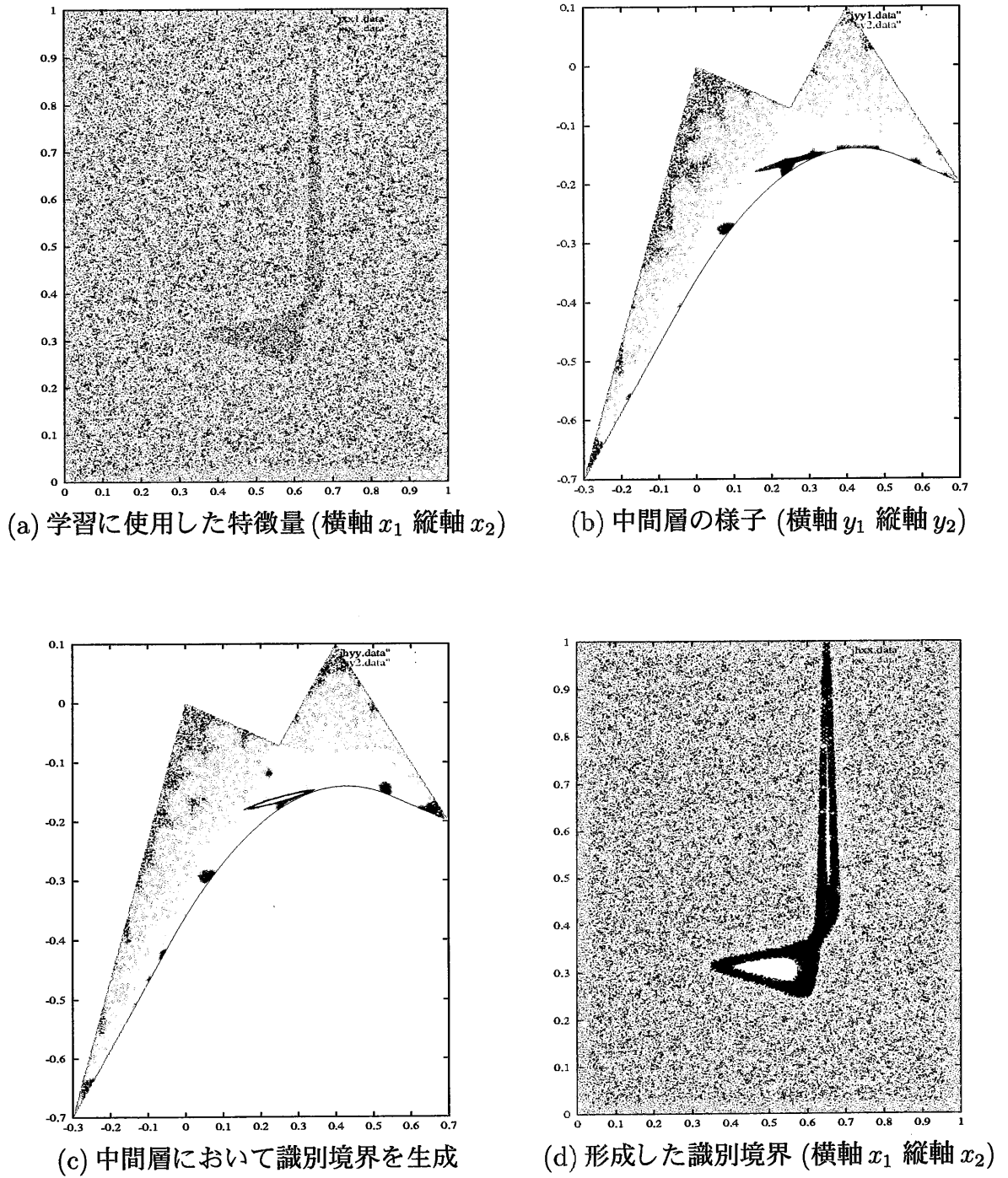
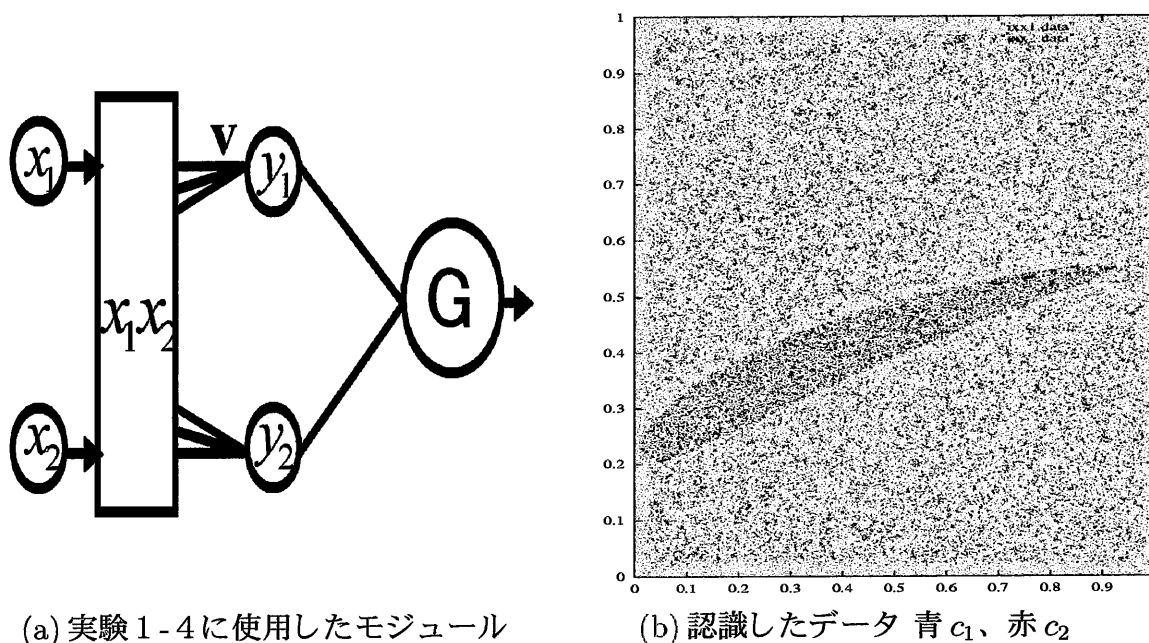


図 4.10: 実験 1-4 認識の様子



(a) 実験 1-4 に使用したモジュール

(b) 認識したデータ 青 c_1 、赤 c_2

図 4.11: 実験 1-5 使用したモジュールと特徴量

実験 1-5 x_1x_2 を使用 (2)

また x_1x_2 の項を使用したとき図 4.11 (b) のような、横長で、左右において異なる尖度をもつような複雑な分布をするクラスの特徴量を認識できることがわかった。

x_1x_2 の項を使用すると、図 4.12 (b) にも見られるように、入力空間のねじ曲げにより、形成された面が極端に拡張したり圧縮したりする例が調査の結果、確認できた。また面が拡張したり圧縮したりする例は x_1x_2 の項のみを使用したときに限らず他の高次項を追加することでもしばしば確認された。

このモジュールでは入力された図 4.12 (a) のクラス c_1 の各特徴量に対し、 c_1 の左右両側において、それぞれ適度な圧縮をかけて、図 4.12 (b) のように c_1 を 1 つの楕円の形に近似している。そうなったところで RBF 出力素子により楕円状の識別境界を生成する (図 4.12 (c))。よって入力特徴量に対して図 4.12 (d) のような識別境界が得られ、認識に成功したといえる。

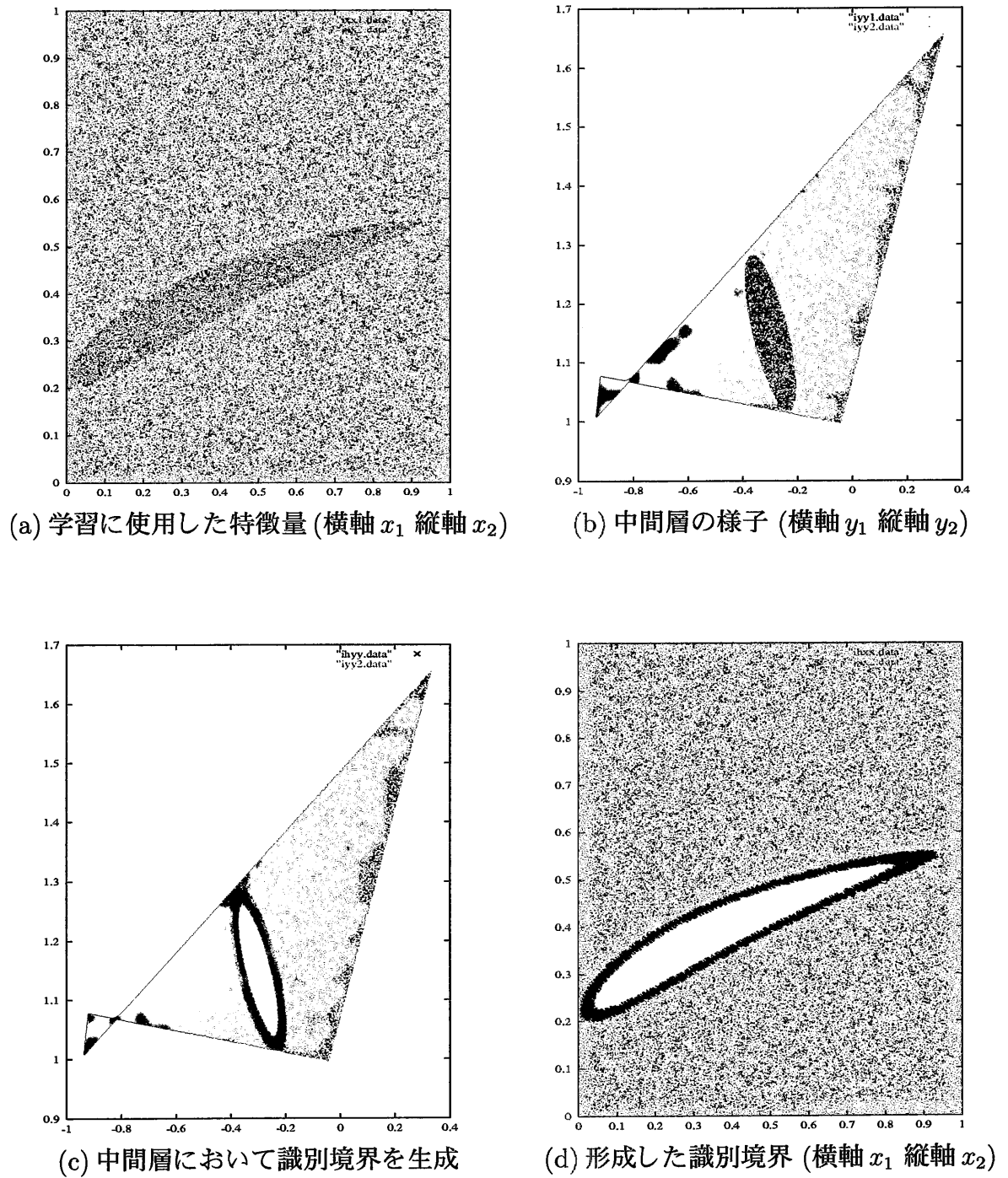
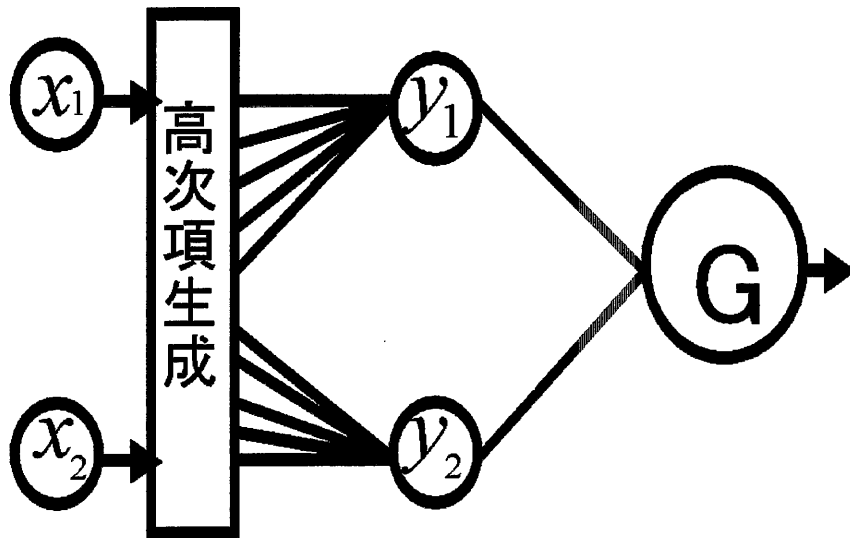
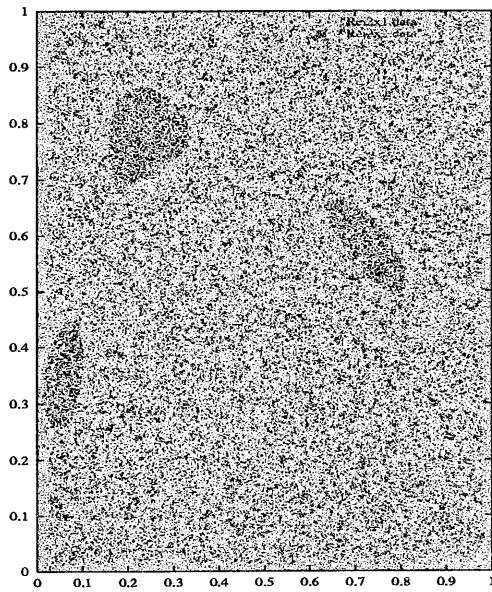


図 4.12: 実験 1-5 認識の様子

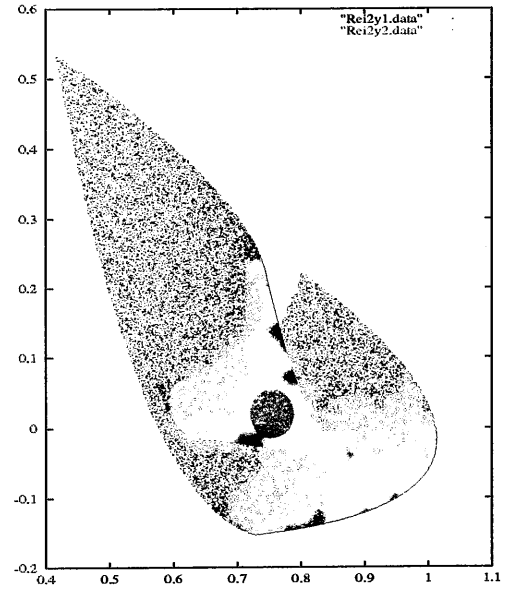
図 4.13: x_1^2 、 x_2^2 、 x_1x_2 をすべて用いたモジュール

実験 1-6 すべての高次項を使用

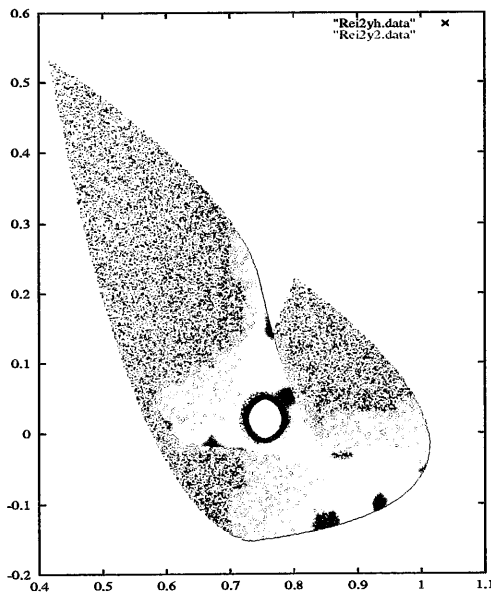
これまでで x_1^2 、 x_2^2 、 x_1x_2 それぞれの項の作用について述べてきたが、すべての項を使用するとそれぞれの項の作用を組み合わせるとさまざまな分布形状の特徴量を持つパターンに対し対応できると考えられる。そこで、すべての項を使用したモジュールで学習・認識実験を行った。その結果を図 4.14～図 4.17 に示す。



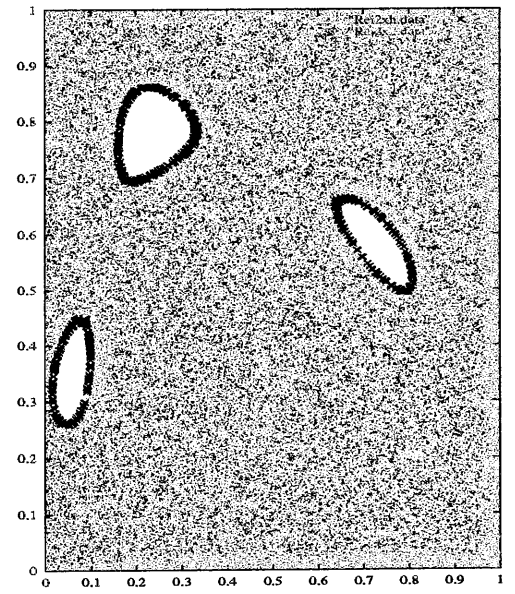
(a) 学習に使用した特徴量 (横軸 x_1 縦軸 x_2)



(b) 中間層の様子 (横軸 y_1 縦軸 y_2)

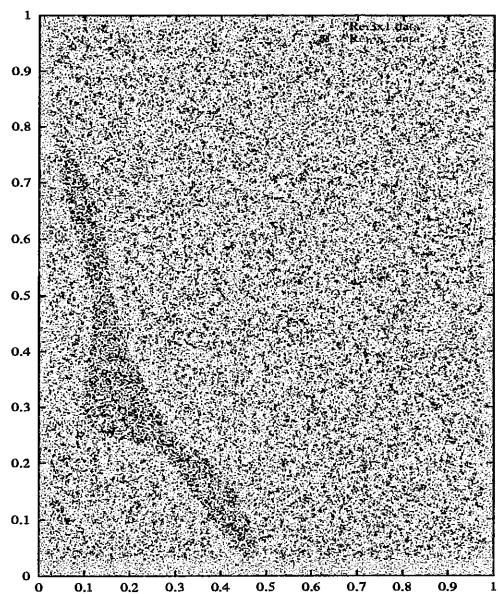


(c) 中間層において識別境界を生成

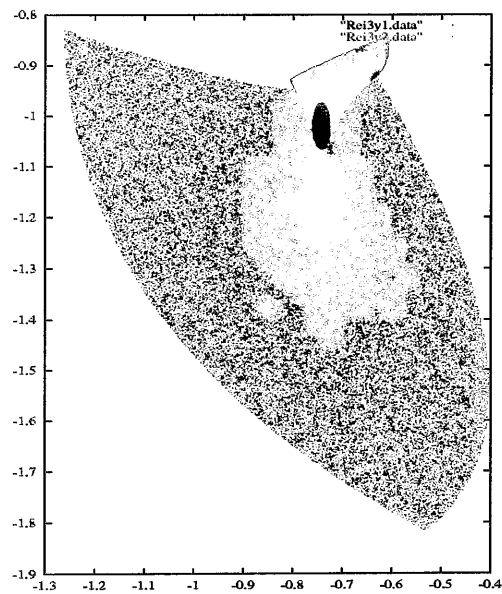


(d) 形成した識別境界 (横軸 x_1 縦軸 x_2)

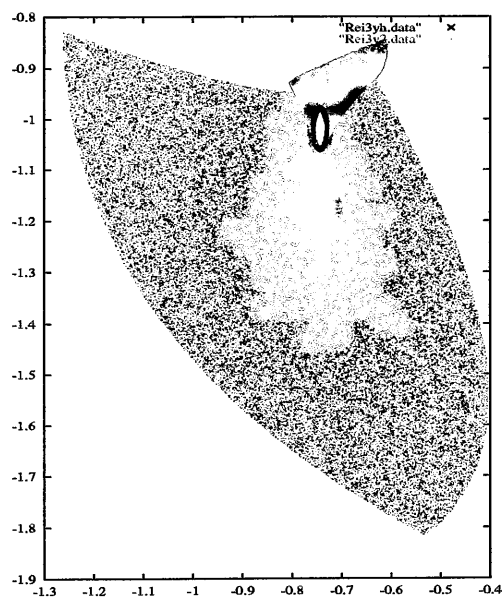
図 4.14: 実験 1-6 認識の様子 (1)



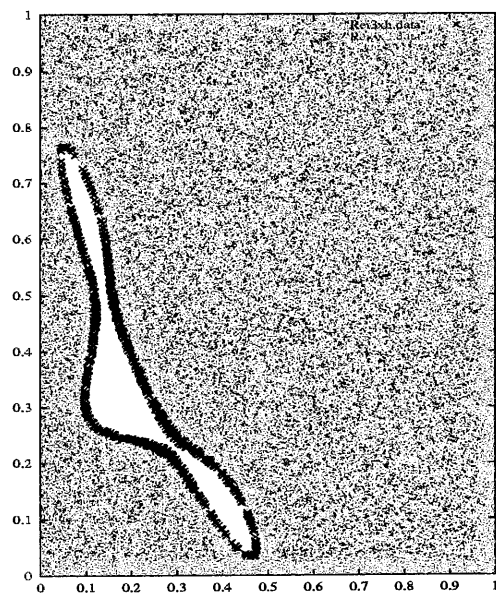
(a) 学習に使用した特徴量 (横軸 x_1 縦軸 x_2)



(b) 中間層の様子 (横軸 y_1 縦軸 y_2)



(c) 中間層において識別境界を生成



(d) 形成した識別境界 (横軸 x_1 縦軸 x_2)

図 4.15: 実験 1-6 認識の様子 (2)

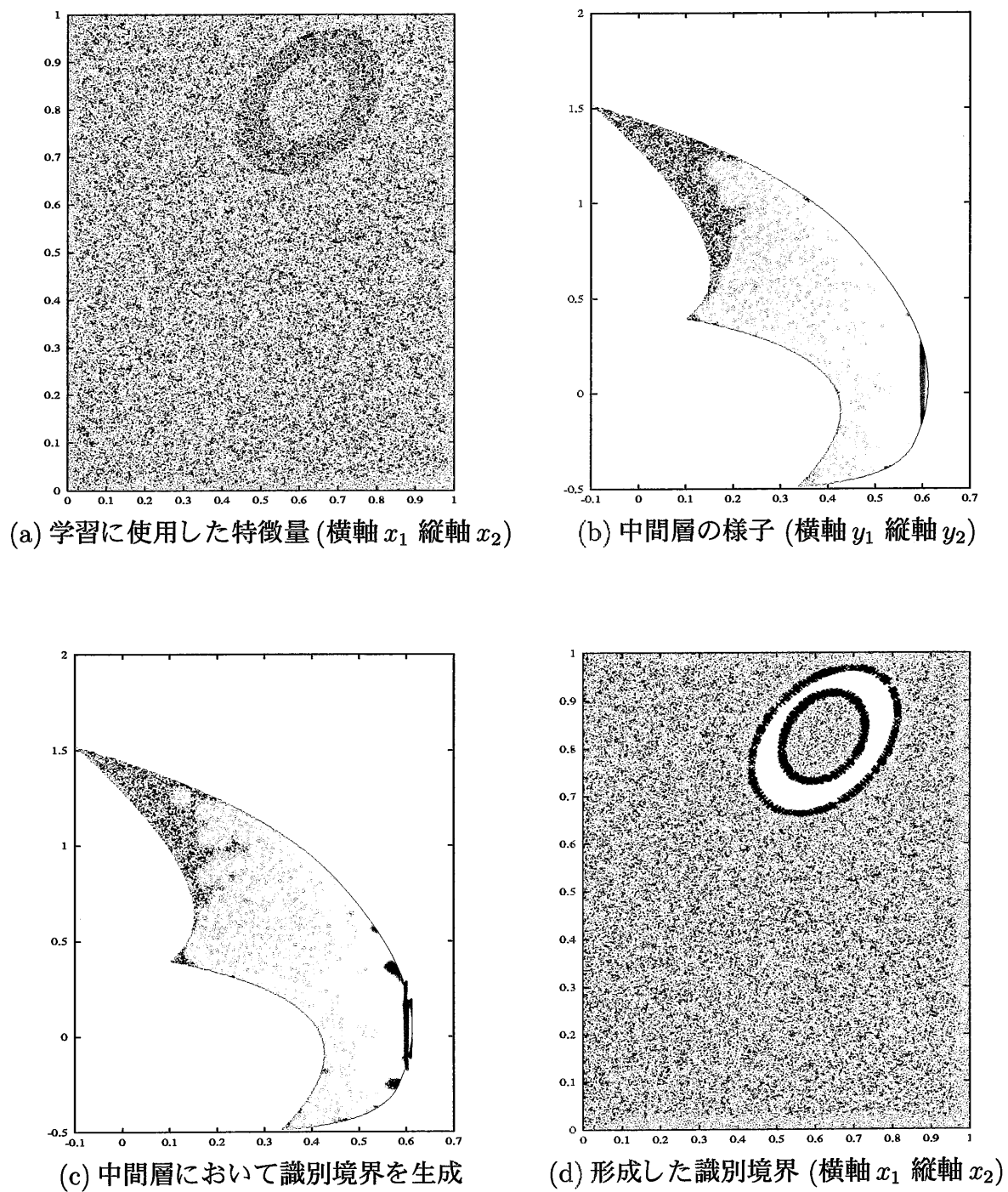


図 4.16: 実験 1-6 認識の様子 (3)

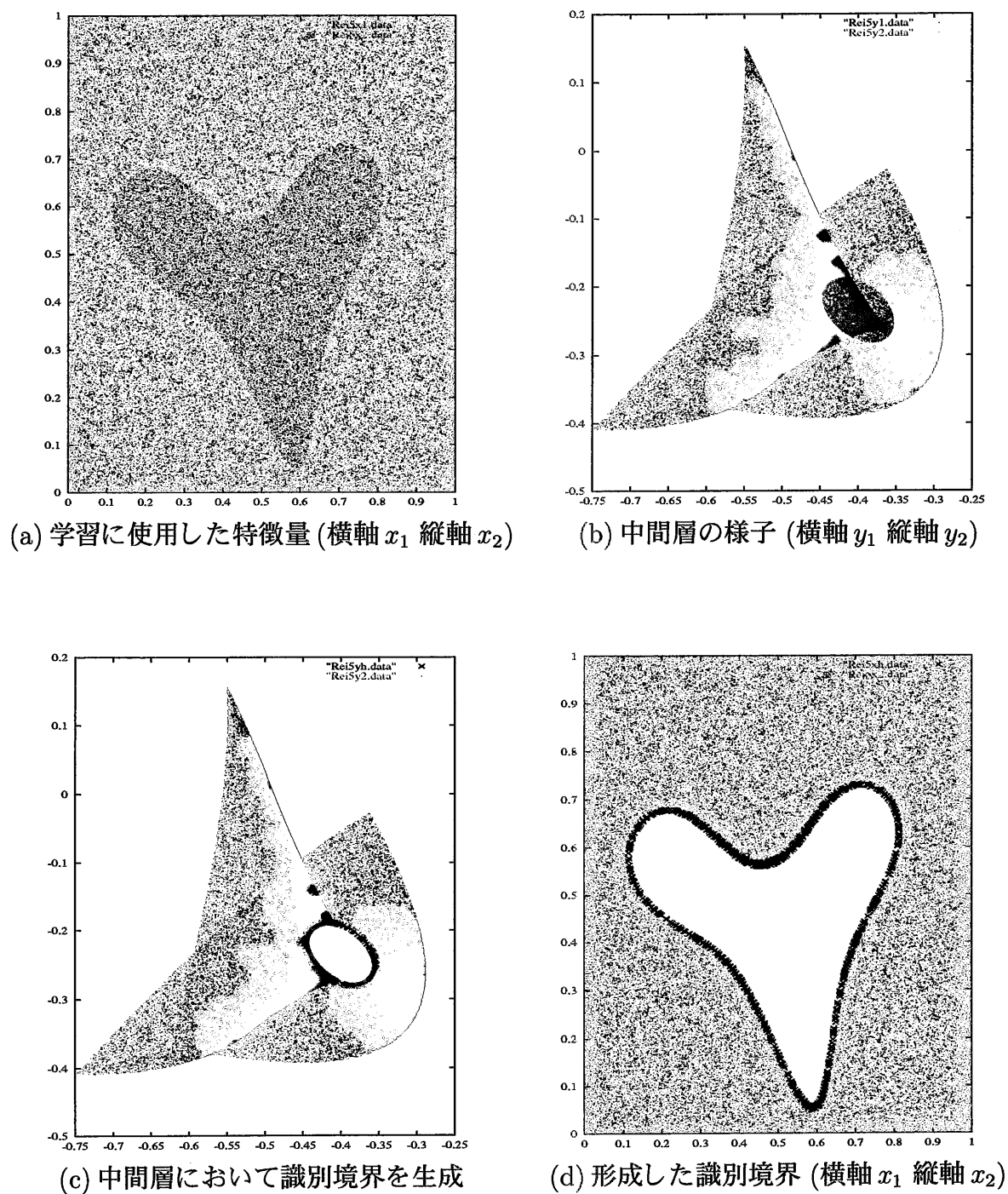


図 4.17: 実験 1-6 認識の様子 (4)

考察1

実験1を考察する。実験1-1～実験1-4の結果から、非線形項を使用することにより、モジュールの中間層において入力空間の長方形領域が片方の軸や対角線を中心に折れ曲がる効果がみられた。そしてそのモジュールは折れ曲がった箇所の一つの楕円状の識別境界を形成することで、入力層の特徴空間において複数の楕円状(あるいは楕円が伸縮したような形状)の識別境界を生成することができることがわかった。これは用いる非線形変換の逆関数が多価関数である場合には、空間的に分離した複数の識別境界面の形成が可能となることを示す。これは自動的にマルチテンプレートを作成することに対応し、さまざまな分布形状の特徴量をもつパターンに対し識別が可能となる。

また実験1-5の結果から、非線形項を使用することにより、中間層において面が極端に伸縮することがみられ、これを利用した特徴量の分布形状の形が確認できた。

実験1-6の結果から、すべての非線形項を追加することで、上記のような識別境界を組み合わせた、さまざまな分布形状をもつ特徴量に対応出来ることがわかった。

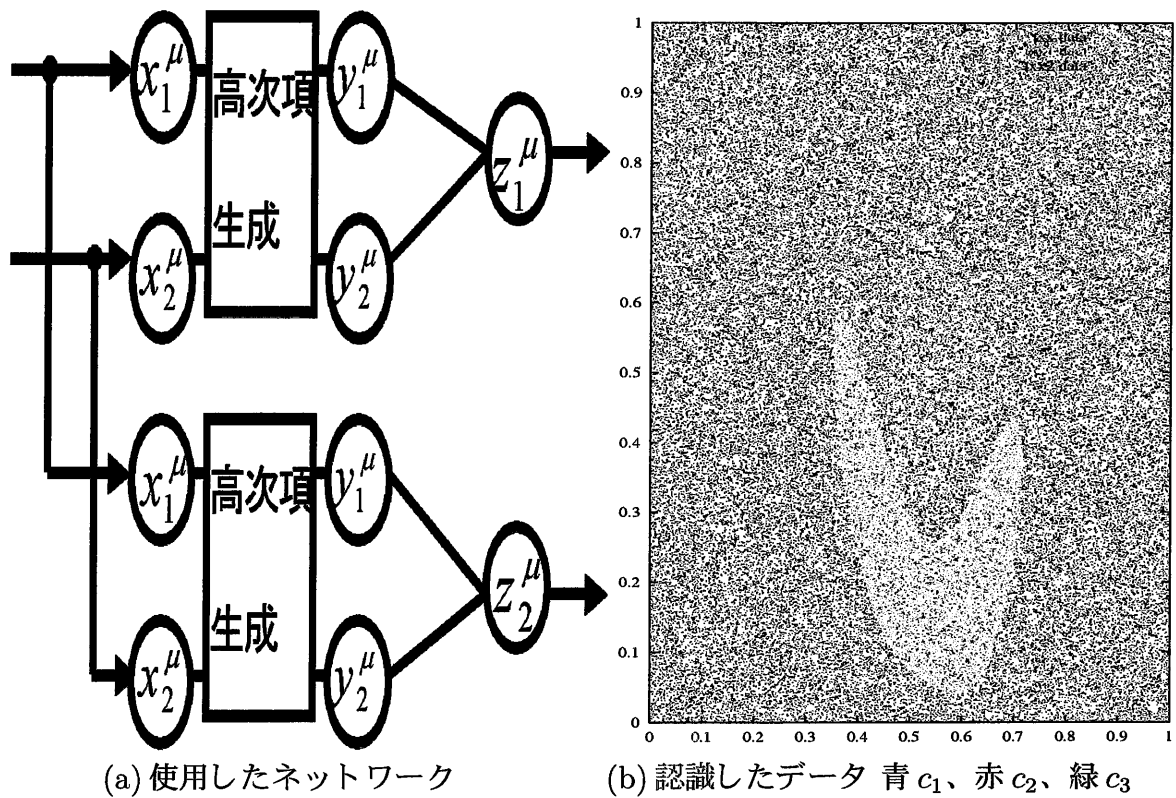


図 4.18: 特徴量が複雑に分布するパターンを含む3クラス問題

4.4.2 実験2 特徴量が複雑に分布するパターンを含む3クラス問題

次に提案した非線形変換を用いたモジュール型ニューラルネットワークを用いて、実際に特徴量が複雑に分布するクラスを含む3クラス問題について実験する。用いる特徴量データは図4.18(b)に示す分布をなすクラスからのもので特徴量が2次元の人工データである。これに対し図4.18(a)で示すような、2次までのすべての高次項生成を取り入れたモジュールを用い、それが2つから構成されるネットワークで学習・認識させる。

また比較のため、高次項を生成しない、第3章で述べたRBF出力素子を用いたモジュール型ニューラルネットワークでも実験を行った。

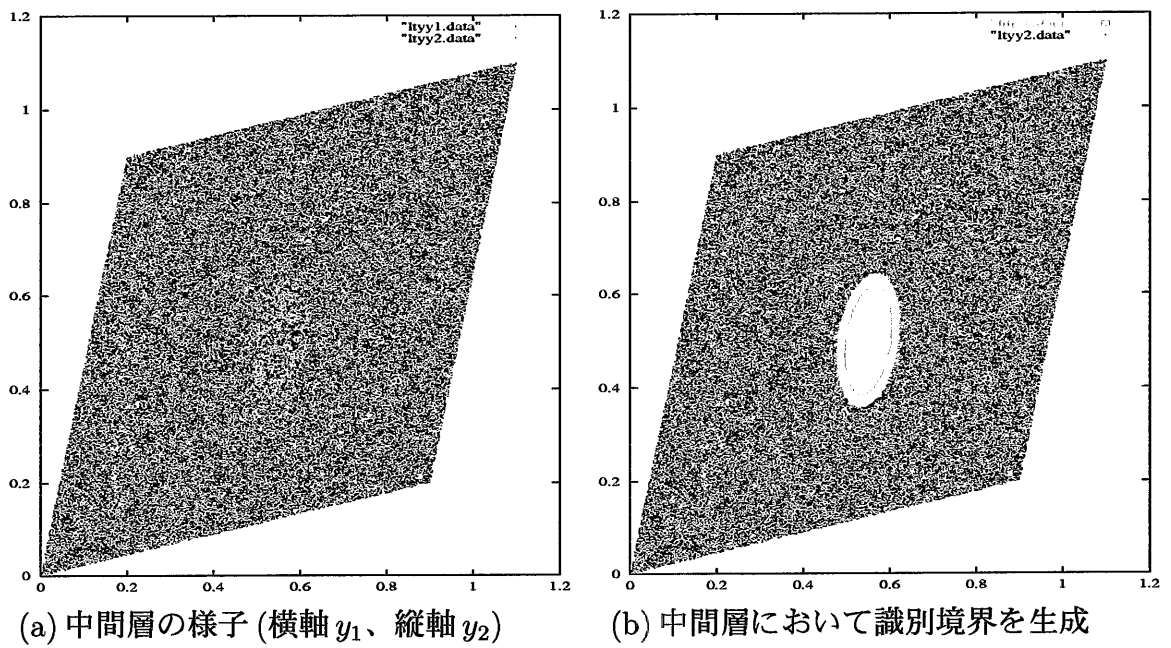


図 4.19: モジュール 1 の中間層における識別境界の生成

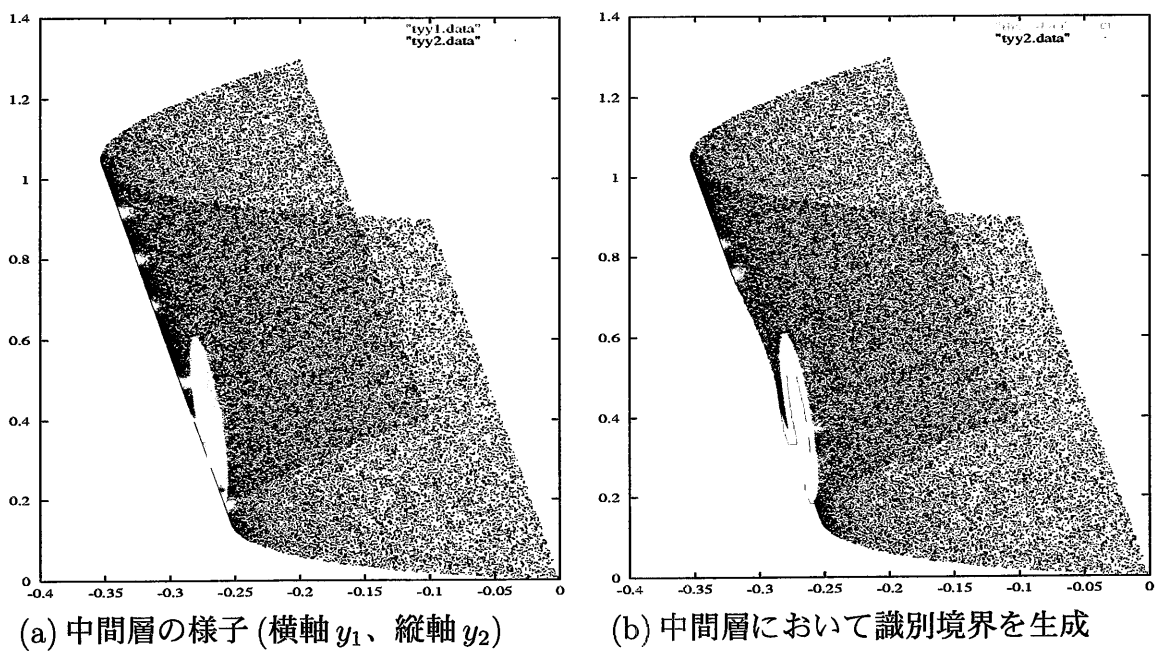


図 4.20: モジュール 2 の中間層における識別境界の生成

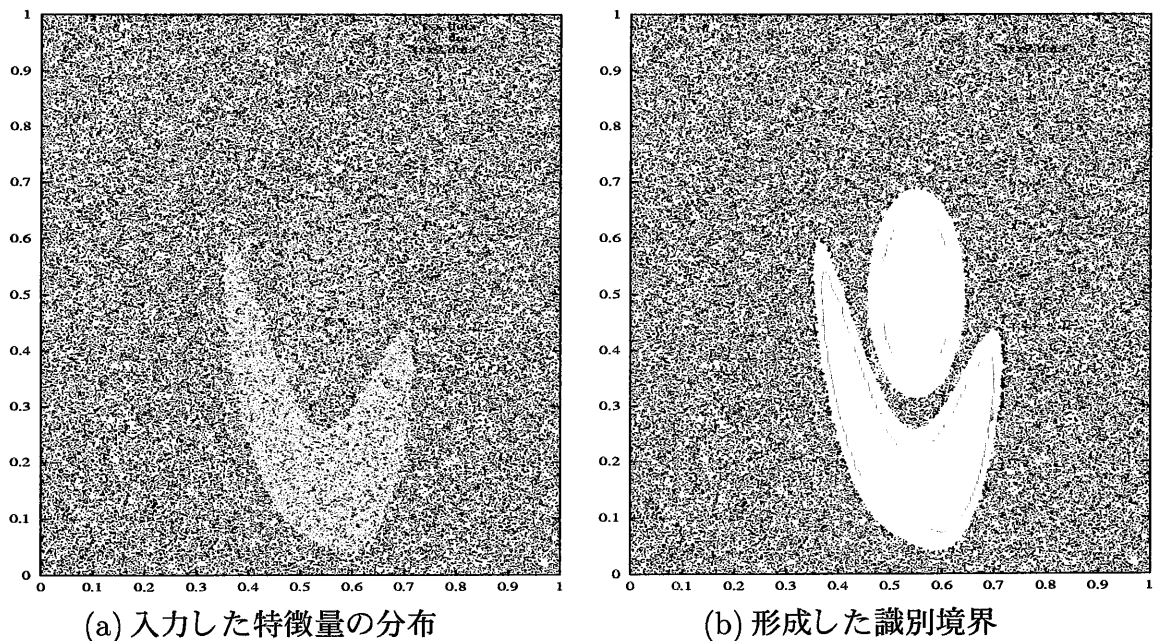


図 4.21: 非線形項を使用したネットワークによる認識

図 4.19(a) は図 4.18(a) の 2 つのモジュールのうち c_1 の学習を担当したモジュール 1 の中間層における様子である。 c_1 は入力される特徴空間の段階で楕円状に分布しているため、中間層において極端な変化は見られず、そのまま楕円状の識別境界を形成している。

次に c_2 を担当したモジュール 2 であるが、凹領域をもつ c_2 に対し、図 4.20 のように、中間層において折り曲げの効果を利用して楕円状に近似し、境界を形成している。

2 つのモジュールで得た境界を組み合わせ、図 4.21(a) のような学習データに対し、図 4.21(b) のような境界を得ることができた。学習データの認識率は 99.98% であった。また非線形項を用いていないネットワークでは、 c_2 の学習を担当するモジュールの学習が収束せず、認識が出来なかった。

考察 2

この実験で使用した非線形変換を用いたモジュール型ニューラルネットワークにおいて、各モジュールで対象となるクラスの特徴量の分布を楕円で近似できるような変換がそれぞれ独立に機能していることが確認できた。

また非線形変換を用いたモジュール 1 で最初から楕円状に分布する特徴量を学

習させた場合、中間層で形成される面において高次項の影響があまりみられなかった。このことから、非線形項を加えてもその項が必要なければ、その項にかかる重みは学習により小さく調整されて認識に悪影響を及ぼさないことがわかった。

比較のために実験した、第3章で示したRBF出力素子をもつモジュール型ニューラルネットワークであるが、 c_1 を担当したモジュールは学習が正しく収束したが、 c_2 を担当したモジュールは、学習が収束しなかった。これはそのモジュールの中間層において c_2 の形状を、楕円状に近似出来ないため、それはいくら重みの更新を重ねても、第3章3.6節で述べた通り、ネットワークで表現できる中間層の面の形状に限界があるためである。その結果、平均二乗誤差の値はある値より下がることはなく、学習が収束しなかったと考えられる。

4.4.3 実験3

次に一般にパターン認識システムにおける性能を評価するために公開されているThe Monk's Problemというパターン認識問題に対して、表3のようなモジュール型ニューラルネットワークで認識を行った。The Monk's Problemは2クラス問題で、各クラスに62の学習データと420のテストデータを持ち、特徴量は5次元である。

それぞれの手法について認識率と、学習に要した更新回数を示す。

表3 実験に使用したモジュール型ニューラルネットワーク

手法名称	中間層出力関数	出力層出力関数
RBF	なし(線形和)	RBF
RBF + シグモイド	シグモイド関数	RBF
RBF + 非線形項(提案手法)	非線形項(2次まで)	RBF

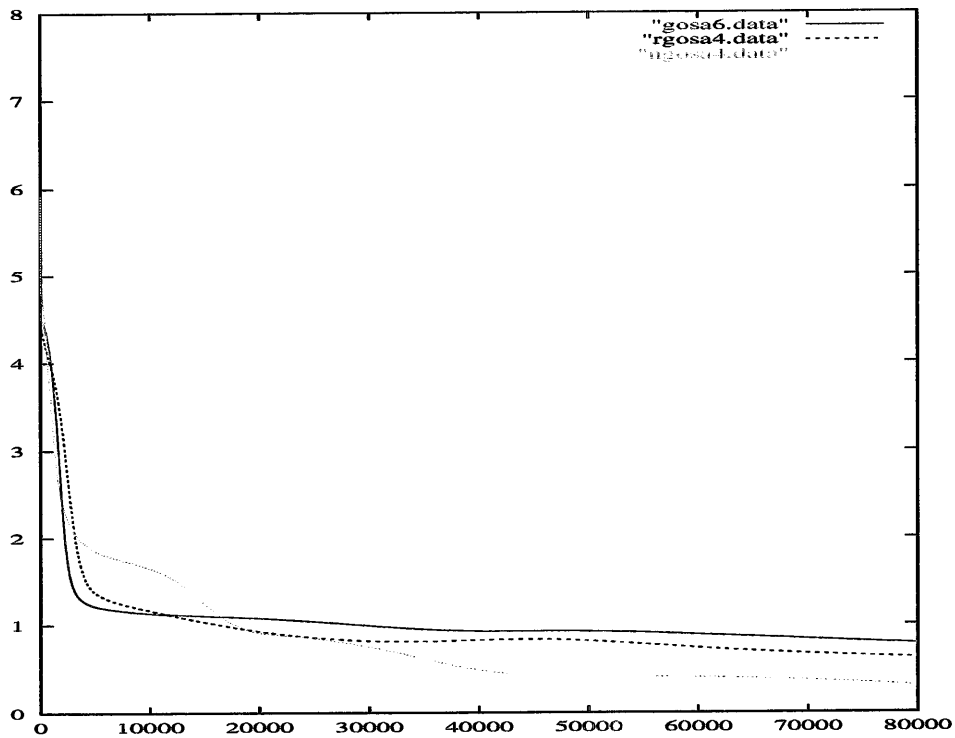


図 4.22: 実験結果 各手法における学習に要した更新回数

図 4.22 において横軸は重みの更新回数, 縦軸は平均二乗誤差を表している。また赤いラインが“RBF”を、青が“RBF + シグモイド”を、紫が“RBF + 非線形項 (提案手法)”をそれぞれ表している。

認識率は表 4 のようになった。

表 4 各手法における認識率

手法名称	テストデータ認識率
RBF	74.5%
RBF + シグモイド	80.7%
RBF + 非線形項 (提案手法)	83.5%

考察3

実験の結果、図 4.22 を見ると “RBF + 非線形変換 (提案手法)” は収束が他の 2 つに比べ遅れてしまっている。これは高次項を新たに生成していることが原因と考えられる。高次項が増えることで、出力層の出力関数において、とり得る値の範囲が他の 2 つに比べ、非常に広がっているため、“plateau” が見られその結果、収束が他の 2 つに比べ遅れてしまったと考えられる。

しかし最終的な平均二乗誤差の値や認識率が他の 2 つと比べて良好な点から “RBF + 非線形変換 (提案手法)” が他の 2 つの手法よりも元のデータの特徴量分布に適切に近付くことが出来たと考えられる。

第5章

結論

5.1 結論

本研究は、特徴量が複雑な分布形状をもつクラスのパターンの認識を実現するシステムの構築を目的とした。そしてその方針としては、階層型ニューラルネットワークに関する下記の三つの手法の組合せでシステムを構築した。

- モジュール型ニューラルネットワーク
- RBF 出力関数
- 非線形変換

第4章において上記3つの手法を組み合わせた非線形変換を用いたモジュール型ニューラルネットを提案した。そして実験1において、提案手法の非線形項の認識に及ぼす作用、有効性を確認した。また実験2において、特徴量が複雑な分布形状をもつクラスに対する認識でその有効性を示した。実験3において実際に公開されているデータベースを用いて認識を行い、その有効性を示した。

5.2 今後の課題

今後の課題としては、まず公開されているデータベースでの実験は The Monk's Problem でしか行っていないので、その他のデータベースを使って実験すること

が1つ挙げられる。

また提案した非線形変換を用いたモジュール型ニューラルネットは、特徴量が比較的次元であれば良いが、もし高次元になると、高次項の生成数が膨大なものになってしまい、学習時間が長くなってしまふことが予想される。よって有効な非線形項を自動的に選択するような工夫が必要であると考えられる。

謝辞

本研究を進めるにあたり、多大な御指導とともにこの研究の機会を与えて下さった東北大学大学院工学研究科 電気・通信工学専攻 阿曾弘具教授に深く感謝致します。

本論文をまとめるにあたり貴重な御意見を頂いた東北大学工学研究科 星宮望教授、東北大学情報科学研究科 堀口剛教授に深く感謝いたします。

日々の研究におきまして研究に対する姿勢など、数々の御指導、御意見を賜った東北大学大学院工学研究科 電気・通信工学専攻 大町真一郎助教授に深く感謝致します。

本研究全般に渡り貴重な専門的御意見、御指導を賜りました東北大学工学研究科 応用物理学科 三宅章吾助教授に心より感謝致します。

東北大学大学院工学研究科 応用物理学科 佐藤俊治助手にはこの研究分野への興味を与えて下さるとともに、常日頃から親身になって貴重な御意見、御助言を頂きましたことを深く感謝致します。

また東北大学大学院工学研究科 電気・通信工学専攻 加藤毅氏、同 下村正夫氏をはじめとする、阿曾研究室の皆様には、常日頃から多岐に渡ってお世話になりましたことを心より御礼申し上げます。

最後に、長きにわたり温かな御支援を続けていただいた両親、家族、友人に感謝致します。

発表予定学会

森島公一，佐藤俊治，下村正夫，三宅章吾，阿曾弘具

“非線形変換を用いたモジュール型ニューラルネットワークによるパターン認識”，
電子情報通信学会ニューロコンピューティング研究会，2002年3月，於玉川大学

参考文献

- [1] 石井建一郎, 上田修功, 前田英作, 村瀬洋: “わかりやすいパターン認識”, オーム社出版局 (1998)
- [2] 上坂吉則, 尾関和彦: “パターン認識と学習のアルゴリズム”, 文一総合出版 (1990)
- [3] 高田直幸: “ニューラルネットワークの構造最適化に関する研究”, 東北大学大学院工学研究科 電気・通信工学専攻 修士学位論文 (2000)
- [4] F.Rosenblatt: “Principle of Neurodynamics”, Spartan,(1962)
- [5] M.L.Minsky and S.A.Papert: “Perceptrons”, The MIT Press,(1969)
- [6] D.E.Rumelhart, G.E.Hinton and R.J.Williams: “Learning Representations by Back-Propagating Errors”, Nature,323,pp.533-536,(1986)
- [7] 鈴木敏, 安藤広志: “モジュール学習による 3次元物体の認識と類別”, 信学技報,NC93-62(1993-12)
- [8] R.Anand, K.Mehrotra, C.K.Mohan, S.Ranka: “Efficient Classification for Multiclass Problems Using Modular Neural Networks”, IEEE Trans. Neural Networks,Vol.6,No.1,pp.117-124 (1995-1)
- [9] 菊川敬二, 永野俊: “モジュラー・ニューラルネットの動作の初期条件と構造への依存性”, 信学技報,NC94-81(1995-3)
- [10] B.L.Lu, M.Ito: “Task Decomposition and Module Combination Based on Class Relations: A Modular Neural Network for Pattern Classification”, IEEE Trans. Neural Networks,Vol.10,No.5,pp.1244-1256(1999-9)

-
- [11] 石原聖司, 永野俊: “RBF 出力素子を用いたモジュール型ニューラルネットワークによるパターン認識”, 信学技報,NC99-172(2000-3)
- [12] B.Nakayama, A.Hirano, I.Ido: “A Multilayer Neural Network with Nonlinear Inputs and Trainable Activation Function: Structure and Simultaneous Learning Algorithm ”, Proc.IEEE and INNS,IJCNN'99,Washington DC,pp.1657-1661 (1999-7)
- [13] 房川実, 中山謙二 平野昇宏: “階層型ニューラルネットワークにおける非線形シナプスの選択学習法”, 信学技報,NC2000-137(2001-3)