

相関係数を用いる削除型ファジィモデル

福元 伸也・宮島 廣美・長澤 庸二

(受理 平成7年5月31日)

A Destructive Fuzzy Model Using a Correlation Coefficient

Shinya FUKUMOTO, Hiromi MIYAJIMA and Yoji NAGASAWA

Numerous studies have been made on fuzzy systems. In many cases, much effort is required for acquiring optimum inference rules. Therefore, a great number of attempts to reduce the effort have been made on tuning the fuzzy inference rules. When we tune the fuzzy inference rules, we aim to minimize inference error, the number of rules, and learning speed. We have already proposed a learning method called a destructive method. However, this method contains several problems concerning learning speed and an overlap of constructed rules. So, we suggest a new learning method that uses a correlation coefficient. In order to verify the validity of this hypothesis, numerical experiments are performed.

1. はじめに

ファジィ推論ルールの自動作成に関するさまざまな研究が行なわれている。多くのファジィシステムにおいて、ファジィルールの構築は、試行錯誤的に行なわれることが多く、推論誤差の小さい最適なルールを得るには、たくさんの労力を必要とする。そのために、ファジィ推論ルールのチューニングに、ニューラルネットワークのもつ学習機能を利用して、ルールの構築を行なう試みがなされている。推論ルールの構築する場合、その目標となるのが、推論誤差、ルール数、学習回数等で、生成型手法は、優れた方法の一つとして知られている¹⁾。しかしながら、この方法では、入出力データとルールの結び付きが強く、データをそれぞれのルールに分散して記憶するという性質は満たさない。その結果、ルール数は必ずしも最適とはいえない。また、多変数の場合にはルールの追加が一度に複数個となり、推論誤差を低く抑える最もよい方法とは必ずしもいえない。そこで、我々は、ファジィ推論のルール構築手法として、多くのルールから不必要なルールを逐次削除していく新たな手法を提案した⁴⁾。そこでは、ルール同士の重なり等の問題があることも指摘した。

本稿では、押野らの提案³⁾した分散と相関を考慮したニューラルネットワークの逐次削減法をファジィシステムに適用した手法を提案し、更にこの手法を文献⁴⁾の削除型モデルと組み合わせた方法を与える。

以下、2章では、逐次型ファジィモデリングと削除型モデルの説明を行ない、3章で関連モデルとそのアルゴリズム、4章で関数近似を行ない、アルゴリズムの効果を示す。

2. ファジィ推論モデル

2.1 逐次型ファジィモデリング

ここでは、これまでに知られている逐次型ファジィモデリングについて説明する^{2,5)}。これは、あらかじめ定めたルールを、与えられた入出力データの関係がうまく表現できるように、最も適したルールへと学習していく方法である。はじめに、システムの入力成分を (x_1, \dots, x_m) 、出力を y とする。この手法の推論ルールは、次式のように表される。ただし、ルール番号を $i=1, \dots, n$ とする。

$$R^i: \text{if } x_1 \text{ is } M_{i1} \text{ and } \dots \text{ and } x_m \text{ is } M_{im} \\ \text{then } y \text{ is } w_i \quad (1)$$

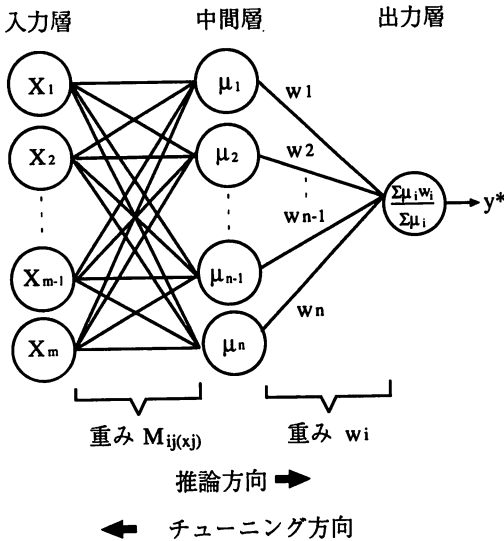


図1: ファジィのネットワーク化

ここで M_{ij} は前件部メンバーシップ関数, w_i は後件部の実数値を表す。

各ルールのもつ出力への適合度 μ_i は次式のように各変数のグレードの積をとる。ただし, $i = 1, 2, \dots, n$ である。

$$\mu_i = M_{i1}(x_1) \cdot M_{i2}(x_2) \cdot \dots \cdot M_{im}(x_m) \quad (2)$$

μ_i により, 最終的な推論結果 y^* は, 次式のように重み付き平均として算出される。

$$y^* = \frac{\sum_{i=1}^n \mu_i \cdot w_i}{\sum_{i=1}^n \mu_i} \quad (3)$$

このファジィ推論の過程を, 図1のようなネットワークに当てはめることができる。このネットワークは3層(入力, 中間, 出力)のネットワークであり, 中間層, 出力層のユニットにおいて各々演算を行う。中間層のユニットでは各ユニット毎に各ルールの全変数によるグレードの積をとる計算を行い, 出力層のユニットでは重み付き平均の計算を行う。また, 入力-中間層間の結合の重みは, 前件部メンバーシップ関数 $M_{ij}(x_j)$, 中間-出力層間の結合の重みは, 後件部実数値 w_i とする。したがって, 入力層より与えられた入力 (x_1, x_2, \dots, x_m) により, 中間層で各推論ルールに対する適合度 μ_i が計算され, 出力層で重み付き平均の計算を行い, 出力 y^* が算出される。

ここでメンバーシップ関数 $M_{ij}(x_j)$ は図2のような

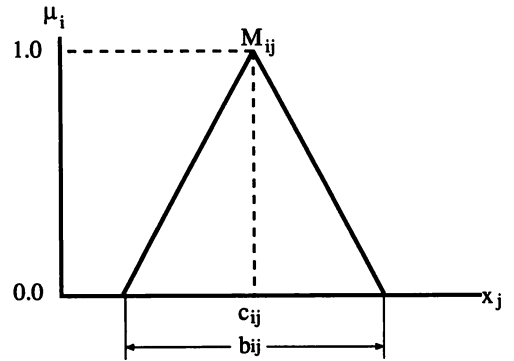


図2: 三角型メンバーシップ関数

三角型を用い, 次のように定義する。

$$M_{ij}(x_j) = \begin{cases} 1 - \frac{2 \cdot |x_j - c_{ij}|}{b_{ij}} & (c_{ij} - \frac{b_{ij}}{2} < x_j < c_{ij} + \frac{b_{ij}}{2}) \\ 0 & (\text{otherwise}) \end{cases} \quad (4)$$

但し, c_{ij} は, メンバーシップ関数の頂点の座標で, $b_{ij} (b_{ij} > 0)$ は, その幅を表す。

このように, ファジィ推論ルールをネットワークに埋め込むことにより, 最急降下法を利用して, 出力層から入力層方向へと結合の重みをチューニングしていくことができる。チューニングのパラメータは, c_{ij}, b_{ij}, w_i となる。

いま, このネットワークにおいて入力 (x_1, \dots, x_m) に対し出力 y^* を得たとする。この場合の理想出力を y^r とする。この場合の学習法は, y^* と y^r の2乗誤差を最小にすることを目的とするので, 次式を評価関数とすることができる。

$$E = \frac{1}{2} (y^r - y^*)^2 \quad (5)$$

y^* はその時の素子間の結合の重み(この場合は, c_{ij}, b_{ij}, w_i)で決まるため, 誤差関数も結合の重みに依存する。誤差関数 E が極小値に達するためには, パラメータ c_{ij}, b_{ij}, w_i を少しずつ変化させていけばよく, このことは, それぞれのパラメータを, 最急降下法にもとづいて変更すればよいことを意味する。

式(2), (3), (4), (5)より, 次の式が得られる。

$$c_{ij}(t+1) = c_{ij}(t) - \frac{K_c \cdot \mu_i}{\sum_{i=1}^n \mu_i} \cdot (y^* - y^r) \cdot (w_i - y^*) \cdot \text{sgn}(x_j - c_{ij}) \cdot \frac{2}{b_{ij} \cdot M_{ij}(x_j)} \quad (6)$$

$$b_{ij}(t+1) = b_{ij}(t) - \frac{K_b \cdot \mu_i}{\sum_{i=1}^n \mu_i} \cdot (y^* - y^r) \cdot (w_i - y^*) \cdot \frac{1 - M_{ij}(x_j)}{M_{ij}(x_j)} \cdot \frac{1}{b_{ij}} \quad (7)$$

$$w_i(t+1) = w_i(t) - \frac{K_w \cdot \mu_i}{\sum_{i=1}^n \mu_i} \cdot (y^* - y^r) \quad (8)$$

但し、 K_c 、 K_b 、 K_w は学習係数で、 t は学習回数を表し、 $sgn(z)$ は、次式のような関数である。

$$sgn(z) = \begin{cases} -1 & ; z < 0 \\ 0 & ; z = 0 \\ 1 & ; z > 0 \end{cases} \quad (9)$$

これらの結果は、初期条件として任意に与えられたネットワークの重み (c_{ij} 、 b_{ij} 、 w_i)を、各入出力データに対し、式(6)から(8)にもとづいて調整すれば式(5)で示される2乗誤差が極小になることを意味している。

2.2 削除型モデル

削除型モデルでは、この逐次型ファジィモデルを利用して推論ルールのチューニングを行う。この方法は、あらかじめ多くのルールを与えておいて学習を開始する。(図3(a))メンバーシップ関数の中心値 c_{ij} と幅 b_{ij} と後件部の実数値 w_i を調整しながら学習を繰り返す。ある程度学習が進むと、式(5)で示される推論誤

差 E の値が小さくなっていく。そして、推論誤差 E と推論誤差の変化率 ΔE の両方が十分小さくなったとき、ルールの削除を行う。削除ルールの選定はまず、いくつかのルールの中の1つを仮に削除して推論誤差を計算する。

次に、このルールを元に戻して別のルールを同様に削除して推論誤差を計算する。この作業を存在するすべてのルールに対して行い、そのなかで最も推論誤差の小さくなる時のルールを実際に削除するルールとする。(図3(b))

その後、ルールの削除とパラメータの調整を行ないながら学習を繰り返し、得られた推論誤差があらかじめ与えたしきい値を越えた場合、すなわち、そのルール数以下では入出力データの関係をうまく表現できない場合に学習を終了する。

この削除型モデルは、たくさんのルールから学習をスタートすることにより、多くのルールに入出力データの関係を埋め込んでいくので、入出力データの情報が構築されたルールの中に分散して記憶されるという特徴を持つ。

以下に、削除型によるルール獲得の手順を整理してみる。

[Step 1] [初期設定]

あらかじめ準備したたくさんのルールに対して、前件部のメンバーシップ関数 M_{ij} の中心値 c_{ij} の初期値は、入力変数 x_j の定義域を等分割するように設定する。また、幅 b_{ij} は各入力変数の隣合うメンバーシップ関数の中心値の間隔よりも大きくし、メンバーシップ関数が十分に重なり合うように設定する。後件部実数値 w_i の初期値はファジィ推論の出力を0にするように決める。

[Step 2]

推論誤差のしきい値 T_1 と推論誤差の変化率のしきい値 T_2 を設定する。

学習回数カウンタ t を1にする。

[Step 3] [ファジィ推論]

以下のチューニング手順 [3-1] ~ [3-6] を行う。

[3-1]

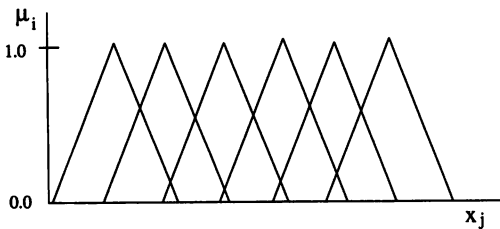
同定用データ番号 p を1にする。

[3-2]

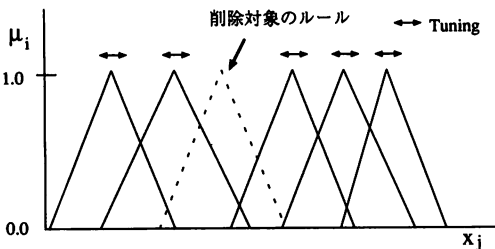
入出力データ (x_1^p, \dots, x_m^p, y^p) を与える。

[3-3]

(2)~(4)式を用いて、入出力データ x_1^p, \dots, x_m^p, y^p に対してファジィ推論を行い、各推論ルール



(a) 初期ルール



(b) ルールの削除

図3：ルールの削除

の適合度 μ_i と推論結果 y^{*p} を求める。

[3-4]

(8)式により後件部の実数値 w_i を調整する。

[3-5]

[3-3] と同じファジィ推論ステップを行う。

[3-6]

前件部のメンバーシップ関数の中心値 c_{ij} , 幅 b_{ij} の調整, および [3-4] の後件部の実数値 w_i の変更は, ファジィ推論出力 y^{*p} およびメンバーシップ関数の適合度 μ_i と出力データ y^p によって行なわれる。

$p=P$ ならば [Step 4] へ。

$p \neq P$ ならば $p=p+1$ として [3-2] へ。

(P は同定用データの全個数)

[Step 4]

推論誤差 $E(t)$, 推論誤差の変化率 $\Delta E(t)$ を次式より算出する。

$$E(t) = \frac{1}{P} \sum_{p=1}^P |y^p - y^{*p}| \quad (10)$$

$$\Delta E(t) = E(t) - E(t-1) \text{ for } t > 1 \quad (11)$$

[Step 5]

推論誤差 $E(t)$ と推論誤差のしきい値 T_1 を比較する。

$E(t) < T_1$ ならば [Step 7] へ。

$E(t) \geq T_1$ ならば [Step 6] へ。

[Step 6]

推論誤差の変化量 $\Delta E(t)$ と推論誤差の変化率のしきい値 T_2 を比較する。

$\Delta E(t) \leq T_2$ ならば [終了]。

$\Delta E(t) > T_2$ ならば $t \leftarrow t+1$ として [Step 3] へ。

[Step 7]

$\Delta E(t) > T_2$ ならば $t \leftarrow t+1$ として [Step 3] へ。

$\Delta E(t) \leq T_2$ ならば [Step 8] へ。

[Step 8] [ルールの選択と削除の方法]

k 番目のルールを削除した時の推論誤差 $E_k(t)$, すなわちルールを $R - \{k\}$ としたときの推論誤差を次のようにおく。

$$E_k(t) = \frac{1}{P} \sum_{p=1}^P |y^p - y^{*p}| \quad (12)$$

for $1 \leq k \leq R$

このとき次の式を満たすルール k_d を選ぶ。

$$E_{k_d}(t) = \min_{k \in R} E_k(t) \quad (13)$$

k_d 番目のルールを削除した場合が最も推論誤差が小さくなる。それゆえ, k_d 番目のルールを削除する。

$$R \leftarrow R - \{k_d\} \quad (14)$$

[Step 9]

$t \leftarrow t+1$ とし, [Step 3] へ。

3. 関連モデル

ここで, 削除するルールを選定する手段として, 相関係数を用いる方法を導入する。押野ら³⁾ は, この関連基準を用いてニューラルネットワークの中間層ユニットの削減を行なった。

ファジィ推論では, ルール間の相関をとった時, その相関係数が高いということはルールとルールがほとんど重なっている状態を意味する。ルール間の相関を1入力変数の場合について考えてみると, ルール i とルール j の相関係数は, 次式により定義される。

$$\rho_{ij} = \left| \frac{\sum^P \{\mu_i(x_p) - \bar{\mu}_i(x)\} \cdot \{\mu_j(x_p) - \bar{\mu}_j(x)\}}{\sqrt{\sum^P \{\mu_i(x_p) - \bar{\mu}_i(x)\}^2 \sum^P \{\mu_j(x_p) - \bar{\mu}_j(x)\}^2}} \right| \quad (15)$$

$\mu_i(x_p)$: ルール i による p 番目の入力データのグレード

$\bar{\mu}_i(x)$: ルール i における全入力データ (x_p ($p=1, \dots, P$)) によるグレードの平均

相関係数 ρ_{ij} が大きい, すなわち1に近いということは, ファジィ推論では, それぞれのルールの重なり部分が大きいということになる。

従って, ルール i, j が図4のような状態にあるとき, すなわち, 2つのルールの重なり部分が非常に大きいとき2つのルールは1つのルールにまとめることができる。よって, 2つのルールのうちどちらか一方を残し, もう片方を取り除いてやる。

そこで, 相関の大きい2つのルールのうち, どちら

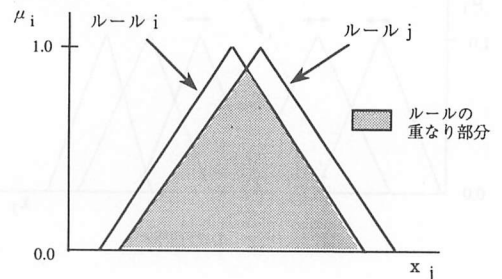


図4: 重なりが多いルール

のルールを削除の対象とするかであるが、ここでは隣接するルール間の中心値の距離の平均を取ることにし、これを分散度 s とする。分散度 s は次式で表される。

$$s = \frac{1}{N-1} \sum_{i=1}^{N-1} l(R^{(i)}, R^{n(i)}) \quad (16)$$

ただし、 $l(R^{(i)}, R^{n(i)})$ は、ルール $R^{(i)}$ とこれに最も近いルール $R^{n(i)}$ の中心値間の距離を表し、 N はルールの個数である。

この分散度をとる意味は、構築されたルールが狭い範囲に偏っているよりも入力空間全体に散らばっている方が、入力データが多くのルールに依存することになり、その結果、同定に用いない他のデータに対する推論誤差が小さくなると考えられるからである。

この分散度 s の値が大きい方が、より広い空間にルールが分布しているといえるので、相関の大きい 2 つのルールのうちこの分散度 s の小さいほうを削除の対象とする。

また、削除されたルールの後件部実数値 w は、残ったもう片方のルールの実数値に加えることにする。

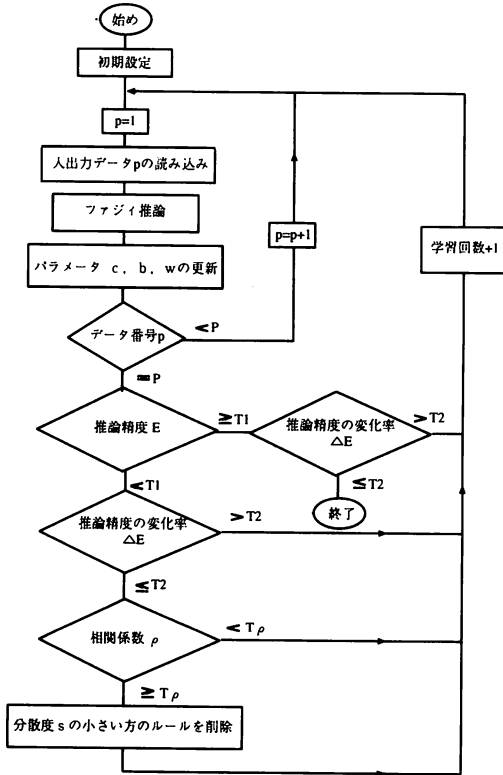


図 5：相関型の学習アルゴリズム

この相関係数を用いてルールの削除を行なう手法を相関型モデルと呼ぶことにし、相関型モデルのルール獲得の手順をフローチャート (図 5) に示しながら説明する。

学習手順は、先に述べた削除型ファジィモデルの [Step 1] から [Step 9] までの手順を基本とする。

まず、初期設定としてパラメータ c_{ij} , b_{ij} , w_i に値を与え、推論誤差のしきい値 T_1 、誤差の変化率のしきい値 T_2 と相関係数のしきい値 T_ρ を設定する。次に、ファジィ推論を行ない、学習の過程において入出力データの間をうまく表現できるようにパラメータ c_{ij} , b_{ij} , w_i の更新を行う。すべての入出力データにおいてこの調整が終わったら、推論誤差 E と誤差の変化率 ΔE を計算し両方がしきい値を下回ったとき、ルール間の相関を取って削除するルールの選定を行なう。相関係数 ρ_{ij} の値が、しきい値 T_ρ よりも大きいルールを削除対象としてこれらの分散度 s を計算する。そして、分散度 s の値が小さい方のルールを削除する。以下、この過程を繰り返し、 E が T_1 より大きく、 $\Delta E \leq T_2$ のとき、すなわち、そのルール数では入出力データの間をうまく表せなくなったとき学習を終了し、その直前のルールの状態を得られた推論ルールとする。

4. 数値例

この相関型モデルの効果を調べるために、既知の関数を使ってその関数を同定し推論ルールを構築する。ここでは、次の 3 つの関数を使って関数近似を行なった。

$$(1) y = \begin{cases} \cos(\frac{\pi}{2} x) & (x \leq 0) \\ 1-x & (x > 0) \end{cases}$$

$$(2) y = \frac{\sin(\frac{\pi}{2} x + 1)}{2}$$

$$(3) y = \frac{\sin(\pi x - \pi)}{2}$$

それぞれの関数の定義域は、 $[-1, 1]$ であり、出力は、 $[0, 1]$ に正規化する。同定用入出力データ (x^p, y^p) を $P (=30)$ 個与える。同定用データは、入力空間を等分割したデータである。推論誤差 E のしきい値 T_1 は、 2.0×10^{-2} で、推論誤差の変化率 ΔE のしきい値 T_2 は、 1.0×10^{-5} 、相関係数のしきい値 T_ρ は、 0.9 を与えた。学習係数は、 $K_c = 0.5$, $K_b =$

表1: 関数近似

		削除型	相関型	生成型
関数 (1)	ルール数	4	4	4
	推論誤差	1.0×10^{-2}	1.4×10^{-2}	1.6×10^{-2}
	学習回数	3972	2426	20
関数 (2)	ルール数	3	4	4
	推論誤差	1.3×10^{-2}	1.0×10^{-2}	3.7×10^{-2}
	学習回数	2493	1703	22
関数 (3)	ルール数	7	7	6
	推論誤差	8.4×10^{-3}	1.2×10^{-2}	2.7×10^{-2}
	学習回数	3972	2810	392

1.0, $K_w=1.0$ で, 初期ルール数は, 100 個とした。

この関数近似の結果を, 表1に示す。表1では, 削除型モデルと, 削除ルールの選定に相関係数を用いる相関モデル, さらに, 比較の対象として, 推論ルール逐次生成手法の1つである荒木らの提案した手法(生成型モデル¹⁾)を用いた結果を同時に示した。

得られたルールの数は, 3つのモデルとも同程度で, 推論誤差は, 生成型よりも, 削除型, 相関型の方で, 若干良い結果が得られた。また, 学習回数は, 3つのモデルの中では, 明らかに生成型の方が少ないが, 削除型と相関型を比較すると, 相関係数を取り入れた方が, 少なくなっている。

次に, 削除型モデルで得られたルールを使って, ルール間の相関をとってさらにルールを削除した場合の結果を表2と図6に示す。

図から削除型モデルでルールの重なりが2ヶ所あるが, 相関係数をとることにより, その重なっているところが1つのルールで置き換わっている。

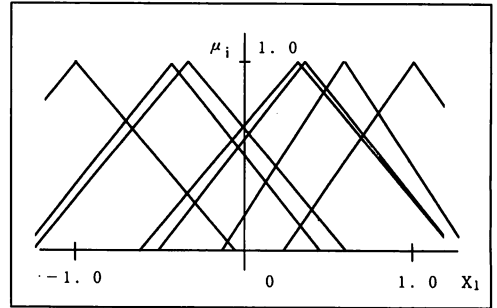
この結果からも分かるように, 削除型モデルでルールの重なりが残っている場合には, さらにルール数を少なくできる。

5. むすび

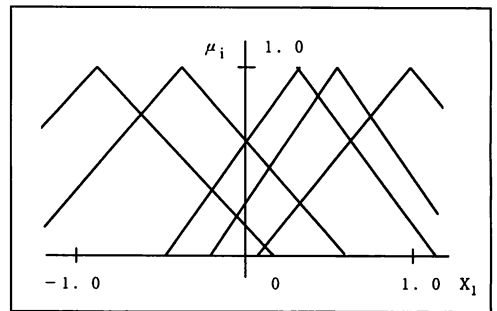
今回, 新たな削除ルールの選定基準として, 相関係数によるルール獲得手法を提案した。ファジィ推論のルールを自動的に構築する手法として生成型モデルや削除型モデルがあるが, 相関係数を利用したモデルでも, これらの手法と同程度のルール数や推論精度を得られ, 削除型モデルと比べると相関型モデルの方が大幅に学習回数が少なくなった。また, 削除型モデルで構築されたルールに相関係数を適用することにより, 重なりの大い推論ルールを取り除くことができた。

表2: 削除型モデル+相関

		関数(1)	関数(2)	関数(3)
削除型 +相関	ルール数	3	3	5
	推論誤差	1.8×10^{-2}	1.3×10^{-2}	9.2×10^{-2}



[削除型]



[削除型+相関係数]

図6: ルールの減少(関数3)

今後, 多次元のデータにおける有効性や他のモデルへの応用も考えていきたい。

参考文献

- 1) 荒木昭一・野村博義・林 勲・若見昇: “ルールを逐次的に生成するファジィモデリングの一提案”, 日本ファジィ学会誌, 4, 4, pp. 722-732 (1992).
- 2) 野村博義・林 勲・若見昇: “ニューラルネットの学習則によるファジィ推論の自動チューニングと障害物回避への応用”, 平成3年電気学会全国大会, s10-27/s10-30 (1991).
- 3) 押野隆弘・尾島潤・山本真司: “誤差逆伝搬学習における中間層ユニット逐次削減法”, 信学論

- (D-II), J-76-D-II, 7, pp. 1414-1424 (1993).
- 4) 宮島廣美・福元伸也・石塚洋一：“ルールの削除に基づくファジィモデリング”，信学論(A), J 77-A, 11, pp. 1555-1562 (1994).
 - 5) 市橋英友・渡辺俊彦：“簡略ファジィ推論を用いたファジィモデルによる学習型制御”，日本ファジィ学会誌, 2, 3, pp. 429-437 (1990).
 - 6) 林 勲・野村博義・若見昇：“ニューラルネット駆動型ファジィ推論による推論ルールの獲得”，日本ファジィ学会誌, 2, 4, pp. 585-597 (1990).
 - 7) 高木英行・香田敏行・小島良宏：“ファジィ推論アーキテクチャに基づくニューラルネット”，日本ファジィ学会誌, pp. 133-141 (1991).
 - 8) 夜久正司・増田達也：“生成的学習法を用いたニューラルネットワークによるファジィ制御ルールの獲得手法”，第8回ファジィシステムシンポジウム講演論文集, pp. 189-192 (1992).
 - 9) 福元伸也・宮島廣美・長澤庸二：“分散と相関を考慮した削除型ファジィシステム”，第10回ファジィシステムシンポジウム講演論文集, pp. 341-344 (1994).
 - 10) Shinya Fukumoto・Hiromi Miyajima・Kazuya Kishida・Yoji Nagasawa: “A Destructive Learning Method of Fuzzy Inference Rules”, Proc. of the International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and the Second International Fuzzy Engineering Symposium, II, pp. 687-694 (1995).