大学院論文輪講 資料 June 26 '70

学習機械について NO. 2

Fuzzy Automata の場合

斉藤(正)研 M2 中所武司

目　次

1. はじめに

我々の世界にある物や状態は、それがあるクラスに属するか、否かを判定する基準が明確には定まっていない場合がしばしばある。例えば「美人」,「右翼」,「大きな数」,などがそれである。これらは明らかに通常の数学的な意味における集合を構成していない。しかしながら、そのように不正確に定義された概念は、人間の思考に関するもの、とりわけ、人工知能、パターン認識、情報処理などに重要な役割を果している。[2]

そこで、カリフォルニア大学の Zadeh は、1965 年に、membership 関数なるものを導入して、定義のあいまいな概念を量的に特徴づける fuzzy 集合を提案し、理論を展開した。[1]

その後この考えは、多方面で応用されつつあるが、本輪講では、特に、Wee and Fu の論文[3]を中心に、fuzzy 集合の、学習系への応用について考えてみたい。

2. fuzzy 集合[1]

空間 $X=\{x\}$ における fuzzy 集合 A とは、X の中の各点 x を、区間 $[0,1]$ の中の実数に対応づける membership 関数 $f_A(x)$ によって、特徴づけら

れる集合のことである。従って $f_A(x)$ が 1 に近ければ，x は A に属する度合が強く，0 に近いときはその度合が弱いことになる。

(例)

X をすべての実数の集合，A を 1 よりも非常に大きな数の fuzzy 集合とすると，$f_A(x)$ を次のようなものと考えることができる。

$f_A(0)=0$, $f_A(1)=0$, $f_A(5)=0.01$, $f_A(10)=0.2$
$f_A(100)=0.95$, $f_A(500)=1$ etc　　　　(例終)

さてここで普通の集合における定義に対応して，いくつかの事柄について，fuzzy 集合における定義を示しておく。

◎ $^\forall x \in X$, $f_A(x)=f_B(x)$ の時に fuzzy 集合 A と B は等価であるといい，$A=B$ で表わす。
◎ fuzzy 集合 A の補集合 A' は $f_{A'}=1-f_A$ で定義される。
◎ $^\forall x \in X$, $f_A(x)=0$ の時，fuzzy 集合 A は空集合 ϕ である
◎ $^\forall x \in X$, $f_A(x) \leqq f_B(x)$ の時，A は B に含まれるという。
◎ A と B の和集合を C とすると $f_C(x)=\max[f_A(x), f_B(x)]$。
◎ A と B の積集合を C とすると $f_C(x)=\min[f_A(x), f_B(x)]$。
◎ $A \cap B = \phi$ ならば，A と B は互いに素であるという。

上で定義した和，積，補集合の演算に関しては，普通の集合の場合と同じように，いくつかの基本的な性質をもっている。一例としては，

$(A \cup B)' = A' \cap B'$　ド・モルガンの法則

$C \cup (A \cap B) = (C \cup A) \cap (C \cup B)$　分配律

があり. これらの意味は, 定義から

$$1 - \max[f_A, f_B] = \min[1 - f_A, 1 - f_B]$$

$$\max[f_C, \min(f_A, f_B)] = \min[\max(f_C, f_A), \max(f_C, f_B)]$$

である。

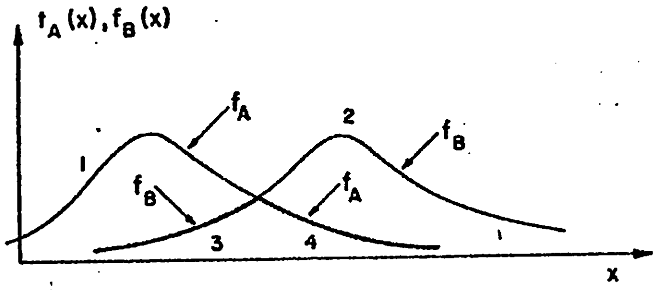


図1 Illustration of the union and intersection of fuzzy sets in R^1

fuzzy集合に関する演算としては上述の他に, 代数積, 代数和, 絶体差が, それぞれ $f_{AB} = f_A f_B$, $f_{A+B} = f_A + f_B$, $f_{|A-B|} = |f_A - f_B|$; で定義されるが, 代数和は $\forall x \in X$, $f_{A(x)} + f_{B(x)} \leqq 1$ の条件のもとでのみ可能である。

次にfuzzy集合を応用(例えば後に述べるFuzzyオートマトン)する上で重要な役割を果すfuzzy関係について見てみよう。空間 X におけるfuzzy関係とは, 積空間 $X \times X$ におけるfuzzy集合 A のことで, membership関数 $f_A(x, y)$; $x, y \in X$ によって特徴づけられる。

(例)

X をすべての実数の集合とし, $x, y \in X$ として,

$x \gg y$ なるfuzzy関係を $X \times X$ におけるfuzzy集合Aとすると，$f_A(x, y)$ を次のように考えることができる。
$f_A(10, 5) = 0$ ，$f_A(100, 10) = 0.7$，$f_A(100, 1) = 1$, etc

（例終）

更に一般化すれば，空間Xにおける(n) fuzzy関係とは，積空間 $X \times X \times \cdots \times X$ におけるfuzzy集合Aのことで，membership関数 $f_A(x_1, x_2, \cdots x_n)$; $x_i \in X$ によって特徴づけられる。

また，Xにおける2つのfuzzy関係A, Bの合成とは，$A \circ B$ で表され，次のようなmembership関数をもったfuzzy関係である。

$$f_{A \circ B}(x, y) = \sup_v \min[f_A(x, v), f_B(v, y)] 。$$

この場合，結合の法則が成立するので

$$A \circ (B \circ C) = (A \circ B) \circ C 。$$

3. fuzzyアルゴリズム[2]

あいまいな量の処理は，普通のアルゴリズムの一般化されたものとして見ることができるので，それを，fuzzyアルゴリズムと呼ぶことにする。

まず，はじめに，fuzzyアルゴリズムの一部分をなすfuzzy命令の例を示すと

(a)「もし x が近似的に5に等しいならば y を近似的に10に等しくせよ。」

(b)「もし x が大きいならば，y をseveral units増大せよ。」

などである。(a)の場合を考えると、近似的に5に等しい数のあつまりをfuzzy集合Aとし、近似的に10に等しい数の集りをfuzzy集合Bとすれば、f_A, f_Bによって(a)の意味は明確になる。

しかし、fuzzy命令の意味の明確さが、それがどのように実行されるかのあいまいさを解決するわけではない。そこでこのことを考えるための簡単な例として「several steps前へ移動せよ。」をとりあげる。several stepsのfuzzy集合をAとし、f_Aを次のように仮定する。
$f_A(0)=f_A(1)=f_A(2)=f_A(3)=0$, $f_A(4)=0.8$, $f_A(5)=f_A(6)=f_A(7)=1$, $f_A(8)=0.7$, $x \geq 9$では$f_A(x)=0$。

まず確率的な実行の場合を考えると、xが選ばれる確率は$f_A(x)$に比例しているとして、$x=0,1,2,3$は確率0, 4は$0.8/4.5$, 5, 6, 7は$1/4.5$, 8は$0.7/4.5$ 9以上は0の確率で選ばれる。

この他、閾値を用いた場合は、$f_A(x) \geq \alpha$なるxの集合A_αを考えると、$\forall x \in A_\alpha$は選択可能である。このA_αに対し、上の確率的実行を用いてもいいし、特別な場合として、αを、A_αが空集合にならないような最大値をとることにすれば、A_αは最大のmembership関数の値をもつxの集合になる。

このようなfuzzy命令を考えることによって、普通のアルゴリズムが、Turing機械と一対一対応することによって、可附番集合の前後関係の中で、正確に定義された

のと同じように，fuzzy アルゴリズムは fuzzy Turing 機械と一対一対応することで正確な意味が与えられる。即ち，時刻 n の状態を q^n，入力を u^n とすると Turing 機械の状態遷移は

$$q^{n+1} = f(q^n, u^n)$$

で与えられ，fuzzy Turing 機械の場合は，membership 関数 $f_R(q^{n+1}, q^n, u^n)$ で特徴づけられた fuzzy 関係 R で表される。

4. fuzzy 集合の応用

4・1 fuzzy オートマトン[2][3]

fuzzy オートマトンは quintuple (I, V, Q, f, g) で表される。I, V, Q はそれぞれ入力，出力，内部状態で f は $Q \times I \times Q$ の中の fuzzy 集合の membership 関数，g は $V \times I \times Q$ の中の fuzzy 集合の membership 関数である。即ち

$$f_A(q_l, i_j, q_m) = f\{q(k) = q_l,\ i(k) = i_j,\ q(k+1) = q_m\}。$$

ここで，入力 i_j が来たときに，状態が q_l から q_m へ遷移することがありえるかどうかの決定は次のように行う。$0 < \alpha, \beta < 1$

1) $f_A(q_l, i_j, q_m) \geq \alpha$ ならば "存在"

2) $f_A(q_l, i_j, q_m) \leq \beta$ ならば "不存在"

3) $\beta < f_A(q_l, i_j, q_m) < \alpha$ ならば "未決定"。

ある特定の入力 i_j に関する fuzzy 遷移行列を

表1とすると、全体のfuzzy遷移表は表2のようになる。但し、行列$[A_0]$の方の要素は $g(v_j, a, q_i)$ である。

特定のn個の入力系列に対する遷移行列は積空間 $T_1 \times \cdots \times T_n$ における(n) fuzzy関係により定義される。即ち $I_n(k)$ はn個の入力系列 $i(k), i(k+1), \cdots i(k+n-1)$ を表すとすれば、

	i_j	
$q(k)$	$q(k+1)$ $q_1, q_2, \cdots, q_m, \cdots, q_s$	
q_1	$\cdot$	
$\cdot$	$\cdot$	
$\cdot$	$\cdot$	
q_l	$\cdots f_A(q_l, i_j, q_m)$	$;\ i_j \in I.$
$\cdot$		
$\cdot$		
q_n		

表-1

	$q(k+1)$			$v(k)$		
	$i(k)$			$i(k)$		
$q(k)$	a	b		a	b	
q_1						
q_2	$[A]$	$[B]$	$\cdots$	$[A_0]$	$[B_0]$	$\cdots$
$\cdot$						
$\cdot$						
q_n						

表-2

$f_A(q_l, I_n(k), q_s)$
$= f_A(q_l, i(k), \cdots, i(k+n-1), q_m)$ となる。ここで P. 3 で述べた2つのfuzzy関係の合成を用いれば、$n=2$の時

$$f_A(q_l, I_2(k), q_r) = f_A(q_l, i(k), q_m; q_m, i(k+1), q_r)$$
$$= \max_{q_m \in Q} \min[f_A(q_l, i(k), q_m), f_A(q_m, i(k+1), q_r)]$$

一般には

$$f_A(q_l, I_n(k), q_s) = \max_{q_o, q_p, \cdots q_r} \min[f_A(q_l, i(k), q_o), f_A(q_o, i(k+1), q_p), ----, f_A(q_r, i(k+n-1), q_s)]$$

[例]

入力i_1に対するfuzzy遷移行列を表3とすると、

$f_A(q_3, i_1, q_2) = f_{32}$

$f_A(q_3, i_1^2, q_2) = \max_{q_x} \min[f_A(q_3, i_1, q_x), f_A(q_x, i_1, q_2)]$

$= \max\{\min[f_{31}, f_{12}], \min[f_{32}, f_{22}], \min[f_{33}, f_{32}], \min[f_{34}, f_{42}]\}$

となる。

$q(k)$	$q(k+1)$ q_1	q_2	q_3	q_4
q_1	f_{11}	f_{12}	f_{13}	f_{14}
q_2	f_{21}	f_{22}	f_{23}	f_{24}
q_3	f_{31}	f_{32}	f_{33}	f_{34}
q_4	f_{41}	f_{42}	f_{43}	f_{44}

表－3

以上が，fuzzyオートマトンの概略であるが，ここで，f_Aが0か1の値しかとらず，しかも，行列[A]，[B]，… [A_0]，[B_0]，…の各行は1つだけ1で他は0であるとすれば，このfuzzyオートマトンは普通の決定オートマトンになる。また各行の和が1になるように正規化されたfuzzyオートマトンは，確率オートマトンと同じ形になる。

4・2 学習システムへの応用[3]

ここでの学習システムは，パターン認識における学習識別を考える。決定機構としては，membership関数の最大値選択を基礎とする。即ち，$\Omega = \{w_i ; i=1, \cdots, R\}$をパターンクラスの集合とし，入力$x$がクラス$w_i$に含まれる度合を$f_{w_i}(x)$で表すと

$f_{w_k}(x) = \max_i f_{w_i}(x)$

の時，$x \in w_k$と決定される。もし"未決定"の態度が許されるならば，$\max_i f_{w_i}(x) \geq C$の時だけ決定されるようにすれば，ある信頼度に達するまでは，ランダム

な手法が使用できる。

次に、n-状態数 fuzzy オートマトンの学習機構及びその収束性について述べる。$\hat{f}_j(k)$ を、時刻 k に、オートマトンが状態 s_j になる membership 関数，$f^{\ell}_{ij}(k)$ を

$$f^{\ell}_{ij}(k) = f\{s(k)=s_i,\ i(k)=i_\ell,\ s(k+1)=s_j\}$$

とすると

$$\hat{f}_j(k+1) = \max_m \min\{\hat{f}_m(k),\ f^{\ell}_{mj}(k)\}$$

ここで、前述の定常的 fuzzy 遷移行列を非定常にすることで、学習性が与えられるが、簡単な例を次に示す。

$$f^{\ell}_{mj}(k) = f^{\ell}_{jj}(k-1) \quad ;\ 但し\ m \neq j\ のとき$$

$$f^{\ell}_{jj}(k) = \alpha_j f^{\ell}_{jj}(k-1) + (1-\alpha_j)\lambda_j$$

$$0 < \alpha_j < 1,\quad 0 < \lambda_j \leq 1,\quad j = 1, \cdots, n$$

この場合、学習部分を図2ようにすると収束性は次に示される。$k \to \infty$ のとき

まず　$\hat{\lambda}_j(k) \longrightarrow \hat{\lambda}_j$

従って　$\lambda^{\ell}_{ij}(k) \longrightarrow \hat{\lambda}_j$

かつ　$\min\{\hat{f}_m(k)\} \leq \lambda_j \leq \max\{\hat{f}_m(k)\}$ なので

$$\hat{\lambda}_j = \max_m \min[\hat{f}_m(k), \hat{\lambda}_j] = \min_m \max[\hat{f}_m(k), \hat{\lambda}_j]$$

従って $\hat{f}_j(k) \to \hat{\lambda}_j$

しかし、この証明は「収束すれば収束する」という論法で不十分である。

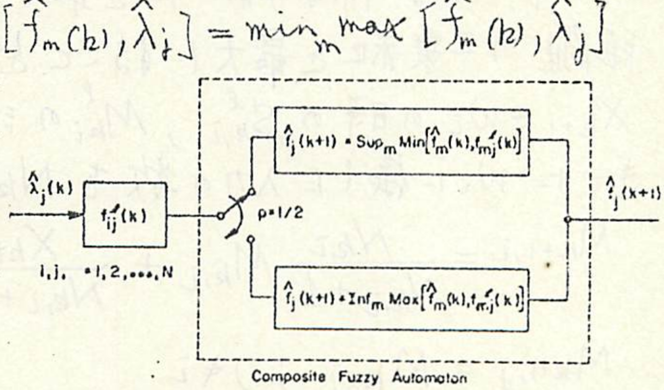


図2　Learning section.

4・3 パターン識別への応用[8]

図3は教師なしで学習するパターン認識系である。決定機構は複数個の識別関数を持ち、ある評価基準に従って最適の識別関数を見つけて、その答を採用する。

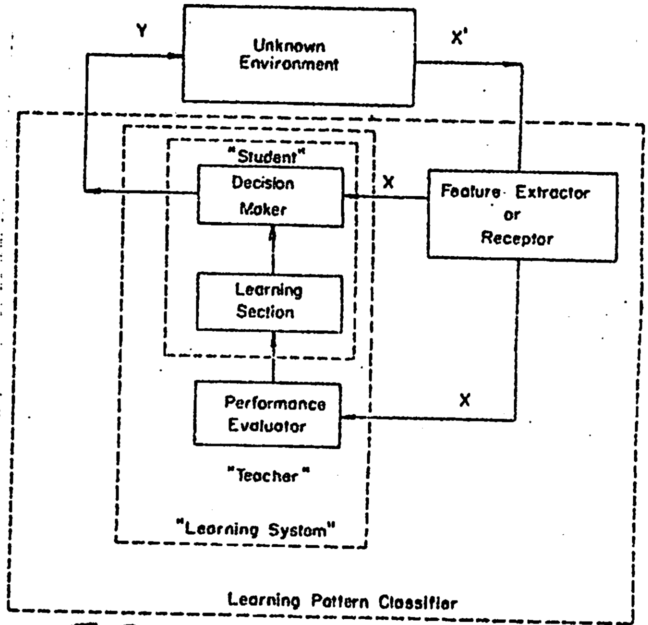


図3 Learning pattern classifier.

今、$\Omega=\{\omega_i\,;\,i=1,\cdots,R\}$ をパターンクラスの集合、$X_1, X_2, ---$ を入力系列、あらかじめ与えられた識別関数の数 N とする。また、時刻 k において l 番目の識別関数に関して、i 番目のクラスの標本平均、標本分散を M^l_{ki}, S^l_{ki}, 評価を K^l_k とすると、

$$K^l_k = \left| w_1\left[\sum_{i=1}^{R} S^l_{ki}\right] - w_2\left[\sum_{i>j}(M^l_{ki}-M^l_{kj})'(M^l_{ki}-M^l_{kj})\right]\right| 。$$

即ち評価は標本分散の和を最小にし、標本平均間の距離の2乗和を最大にすることを基礎にしている。

$X_{k+1}\in\omega_i$ の時の $S^l_{k,i}$, M^l_{ki} の漸化式は、時刻 k までに ω_i に属した入力の数を $N_{k,i}$ として

$$M_{k+1,i} = \frac{N_{k,i}}{N_{k,i}+1}M_{k,i} + \frac{X_{k+1}}{N_{k,i}+1}$$

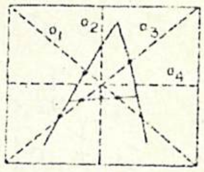
$$M_{k+1,j} = M_{k,j} \; ; \; j\neq i$$

$$S_{k+1,i} = \frac{N_{ki}}{N_{ki}+1} S_{ki} + \frac{N_{ki}}{(N_{ki}+1)^2}(X_{k+1}-M_{ki})'(X_{k+1}-M_{ki})$$

$$S_{k+1,j} = S_{k,j} \quad j \neq i$$ 。

そして、$\{K_k^l ; l=1,\cdots,N\}$ の中の最小のものが、Nコの識別関数の中から最適のものを与える。

次に、英文字AとBを識別する場合の例を示す。特徴抽出器は図.4のようなもので、x_iは直線a_iとパターンとの交点の数を示す。時刻kにおける、Nコのmembership関数$\hat{\lambda}_j(k)$は



Sample Feature $\{x_1, x_2, x_3, x_4\} = [3232]$

図4 Description of feature extraction.

$$\hat{\lambda}_j(k) = 1 - \frac{K_k^j}{C} \quad ; j=1,2,\cdots N$$

但し、$C \geq \max_{k,j} K_k^j$, $\lim_{k\to\infty} \hat{\lambda}_j(k) = \hat{\lambda}_j$

である。図5は識別関数を10コ用い、60コずつの訓練パターンで計算機シミュレーションをしたものである。

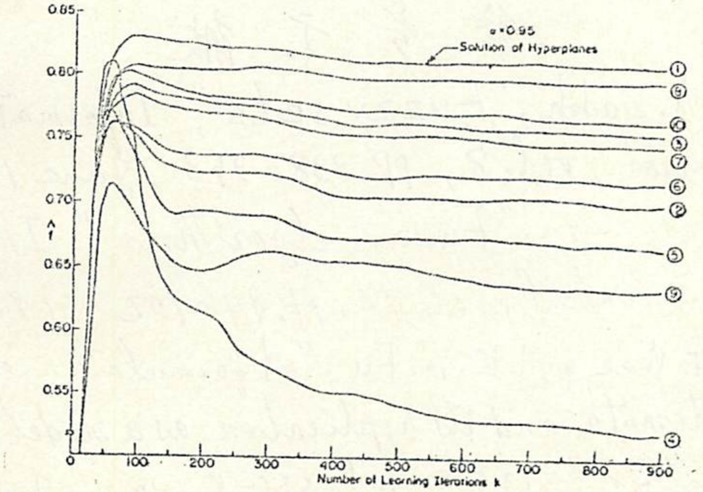


図5 J versus k for two-class pattern classification problem.

5. おわりに

fuzzy集合はそのmembership関数を0と1に限れば、普通の集合になることからわかるように、集合概念の一般化であり、従って、fuzzyオートマトンは普通の決定オートマトンや確率オートマトンの一般化である。その為、理論展開が、従来のものと似ることは避けられないが、ここで述べたWee and Fuのパターン認識への応用にもその価値がfuzzyオートマトンによりも、むしろ従来の統計的手法にあるとも考えられ、まだ十分にfuzzy集合の概念を生かした応用とは言えない。その点で、既成の意味の拡張ということにとらわれず、fuzzy集合の概念だけを原点にした独自の展開が望まれる。

いずれにせよ、このようなfuzzy問題を扱うとき、その扱い方がfuzzyにならないように気をつけなければならない。

参考文献

[1] L. A. Zadeh, "Fuzzy sets," Information and Control, vol. 8, pp. 338~353, June 1965.

[2] ——————, "Fuzzy algorithms," Information and Control, vol. 12, pp. 94~102, 1968

[3] W. G. Wee and K. S. Fu, "A formulation of fuzzy automata and its application as a model of learning systems," IEEE, vol. SSC-5, no. 3, July 1969