

퍼지 신경망을 이용한 패턴 분류기의 설계

김재현, 서일홍, 김태원
한양대학교 전자공학과

Design of a Pattern Classifier using Fuzzy Neural Networks

J.H. Kim, I.H. Suh, and T.W. Kim
Dept. of Electronic Eng., Hanyang University

Abstract

Most of clustering methods usually employ the center of a cluster to assign the input data into a cluster. When the shape of a cluster could not be easily represented by the center of cluster, however, it is difficult to assign input data into a proper cluster using previous methods. In this paper, to overcome such a difficulty, a cluster is to be represented as a collection of several subclusters. And membership functions are used to represent how much input data belong to subclusters. Then the position of each subcluster is adaptively corrected by use of a competitive learning neural network. To show the validity of the proposed method, a numerical example is illustrated, where FMMC(Fuzzy Min-Max Clustering) algorithm is compared with the proposed method.

1. 서론

패턴 분류(pattern classification)는 이미 소속 클래스(class)를 알고있는 표본 패턴 데이터(sample pattern data)를 이용하여 클래스의 경계선을 형성하는 것이다. 이러한 패턴 분류와는 달리 패턴 클러스터링(pattern clustering)은 소속 표시가 있지않은 패턴 데이터로 하여금 가장 어울리는 그룹에 속하도록 하는 것을 목적으로 한다. 어떤 입력 데이터가 주어진 경우, 하드 클러스터링(hard clustering)방법에서는 거리가 가장 가까운 클러스터에 소속도 1을 수여하지만, 퍼지 클러스터링 방법에서는 거리에 비례하여 소속도를 부여하게 된다. 따라서 입력 패턴 데이터와 각 클러스터들 사이의 관계를 퍼지 클러스터링이 더 세밀하게 표현할 수 있다. 이러한 퍼지 클러스터링을 수행하는 대부분의 방법들은[2~7] Bezdek의 FCM(fuzzy c-means)이 기초가 되었다.

FCM류의 많은 알고리즘들[2~7]은 각 패턴 데이터의 모든 클러스터에 대한 소속도의 합이 1이 되도록 하여야

하기때문에 특정 클러스터의 소속도는 특정 클러스터와 입력값과의 거리와 특정 클러스터를 제외한 클러스터들과 입력값과의 거리를 고려하여 상대적으로 결정된다. 따라서 클러스터들의 소속도는 서로 영향을 줄 수 밖에 없었다. FCM 알고리즘의 특징은 다음과 같다. 1) 동일 클러스터내에의 같은 거리에 있는 입력값이라도 입력값의 위치에 따라 다른 소속도를 나타내게된다. 2) 노이즈(noise)가 입력되는 경우, 비록 모든 클러스터들로부터 거리는 상당히 떨어져있다 하더라도 노이즈에 대해 $1/n$ (n: 클러스터 개수)에 가까운 부적당한 소속도를 나타내었다. 이러한 문제를 해결하고자 제약조건을 완화하여 각 특징 벡터의 모든 클러스터에 대한 소속도의 합이 0과 1사이라 간주하고 소속도를 구한 알고리즘이 Krishnapuram 과 Keller의 PCM(possibilistic c-means)알고리즘[8]이다. 완화된 제약으로부터 구한 소속도는 다른 클러스터로부터 어떠한 영향도 받지 않도록 구성되어 있다. 따라서 각 클러스터로부터 거리가 멀리 떨어진 노이즈에 대해서는 상대적으로 낮은 소속도를 나타내었으며, 클러스터로부터 동일 거리에 있는 입력 패턴에 대해서는 동일한 멤버십 값을 출력할 수 있었다. 그러나 여러 클러스터의 패턴 데이터 공유(sharing)라는 관점에서보면 FCM은 나름대로의 타당성을 지니게된다.

FCM이나 PCM에서는 한개의 클러스터당 한개의 정형(prototype)을 갖는 클러스터링을 수행하였다. 그림 1(a)처럼 clover모양의 데이터 군를 생각해 보자. 그림 1(b)는 1개의 정형을 갖는 클러스터의 멤버십 값의 등고선을 나타내었으며 클러스터에 속하는 입력벡터 a와 클러스터에 속하지 않는 입력벡터 b가 같은 소속값을 갖는 모순이 발생됨을 알 수 있다. 이는 한개의 정형을 갖는다는 클러스터 모델링에서오는 결과이다. 따라서 FCM이나 PCM으로 주어진 데이터 군을 정확히 표현하기 위해서는 여러 클러스터를 할당하여 그림 1(c)에서 처럼 구성하여야 한다.

한편, 입력이 순차적으로 주어질 경우, 초기에 필요한 클러스터의 갯수를 추정한다는 것은 매우 어려운 일이다. 따라서 FCM이나 PCM을 이러한 경우에 적용하는것이 어

렵다. Fuzzy min-max clustering(FMMC)신경망 알고리즘[9]은 이러한 어려움을 해결하기 위한 알고리즘들 중 하나이다. FMMC는 클러스터내의 모든 입력에 대해 소속도를 1로 하여 클러스터내의 모든 특징 벡터를 정형이라고 생각하는 접근방법이다. FMMC 신경망 알고리즘의 특징은 다음과 같다. 1) 클러스터의 수가 제한되어 있지 않다. 2) H/W로 구현하기 쉽다. 3) On-line 처리가 가능하다. 4) 조정하여야 할 Parameter들의 수가 적다. 그러나 FMMC 신경망 알고리즘에서는 hyperbox라는 클러스터의 모양이 정해져있고 크기또한 특정 파라미터에 의해 제한되어 있다. hyperbox의 크기를 작게하면 할수록 자세한 클러스터의 표현이 가능하다. 그러나 hyperbox의 크기가 작으면 작을수록, 많은 클러스터를 발생시키고, 비록 계층적인 클러스터링 수행중 가장 기본이되는 클러스터링을 목적으로 한다 하더라도 특징 벡터 공간의 입력 패턴들과 별 차이가 없어지게 되며, 따라서 클러스터링의 의미가 상실된다. 따라서 클러스터의 크기에 대한 제한은 가능하다면 없어지는 것이 좋다. 그러나 FMMC에서 클러스터의 크기에 대한 제한이 없다면 최악의 경우 모든 클러스터링이 커다란 hyperbox 하나만으로 표현될 수 있다. 즉 hyperbox의 크기가 커지면 커질수록 원래 클러스터의 부분이 아닌 부분을 포함할(risk decision) 가능성이 커진다. (그림 2)

본 논문에서는 동적인 클러스터 할당시 FMMC에서 발생하였던 risk decision area를 최소화 하고자 클러스터의 경계선을 학습하는 알고리즘을 제시하고자 한다. 본 알고리즘에서는 클러스터의 크기에 대한 제한을 두지않아 다양한 클러스터 형태에 적용할 수 있으며, 부클러스터(sub-cluster)들에 의해 경계선을 나타내고, 경계선 내부의 모든 입력 값에 대해 소속도 1를 갖고서 경계선 외부에서는 거리가 멀어짐에 따라 감소하는 비선형 소속도를 갖게된다. 제시하고자 하는 알고리즘의 특징은 다음과 같다. 1) on-line 클러스터링 특성 2) 클러스터의 동적인(dynamic) 할당 2) 각 클러스터에 독립적인 소속도 할당 3) cluster들의 표현에 있어서 cluster의 경계선을 표현하는 sub-cluster개념의 NN(neural network) 구현 등이다. 2장에서는 기본적인 클러스터 신경망(basic cluster neural network)과 알고리즘을 설명하였고 3장에서는 이러한 신경망의 확장 network에 대해 설명하였으며 4장에서는 전체 시스템의 성능을 평가할 수 있는 몇가지 예를 들었다. 마지막으로 5장에서 본 논문의 결론및 향후 연구 과제를 다루었다.

2. 기본적인 클러스터링 신경망 (Basic Clustering Neural Network)

본 클러스터링에서 사용하는 신경망은 경쟁학습을 수행하게 된다. 이는 입력패턴을 표준패턴과 비교하여 표준패턴과 더 가까울수록 더 큰 출력값을 내도록 학습시키는

것을 의미한다. 이러한 일반적인 신경망의 구성은 Kohonen의 feature mapping neural network에 기반을 둔 신경망이며, FMMC(fuzzy min-max clustering) 신경망도 이러한 신경망 구성을 이루고 있다. 본 논문에서 사용한 신경망 역시 이러한 부류의 신경망으로 순차적인 입력에 대해 클러스터링(clustering)을 수행하는 기본적인 신경망의 구조는 그림 3과 같다. 신경망의 구조는 입력을 받아들이는 입력층과 클러스터의 경계선을 나타내는 부클러스터층으로 이루어진다. 이러한 단위 신경망을 BCNN(basic clustering neural network)이라고 명명하고 단위 BCNN은 1개의 클러스터를 의미하게 된다.

그림 3에서 패턴 x_i 가 입력되면 모든 클러스터(cluster)의 부클러스터(sub-cluster)들 중 거리가 가장 가까운 부클러스터가 입력값에 대해 반응하게된다. 이때 i 번째 입력 벡터가 j 번째 클러스터의 k 번째 부클러스터에 소속되어질 멤버십 값은 식(1)과 같이 정의된다.

$$u_{ik} = \frac{1}{1+(\lambda d_{ik})^\gamma} \quad (0 < \gamma < 1, \lambda : \text{constant}) \quad (1)$$

$$d_{ik} = \|x_i - w_{jk}\| \quad (2)$$

(w_{jk} : j 번째 클러스터의 k 번째 부클러스터 벡터)

여기서 λ 는 거리에 대한 스케일 상수이고, γ 는 퍼지함의 정도를 나타내는 상수이다.(그림 4참고) d_{ik} 는 M 차 특징공간상에서 식(2)와 같이 정의되며, 유클리디안 거리를 나타낸다. λ 의 선택에 따라 u_{ik} 의 값이 0.5가 되는 지점이 결정된다. 즉 d_{ik} 가 $1/\lambda$ 이 되는 지점에 u_{ik} 가 0.5의 값을 갖게된다. 따라서 λ 값이 커지면 퍼지함의 정도가 커지게 되고, λ 값이 작아지면 퍼지함의 정도가 작아지게 된다. 그러므로 λ 가 크면 해당 입력패턴에 대한 부클러스터들의 소속도가 급격하게 감소하게 되고 소속도 0.5를 넘기는 부클러스터의 수가 감소하게 되어 새로운 클러스터의 발생을 유도하게된다.

γ 값은 1보다 작은 값으로 정해지고 상수 λ 에 대해 독립적으로 u_{ik} 의 퍼지함정도를 결정하게된다. 즉, 어떤 부클러스터의 멤버십 값이 클수록 현재의 입력과 가까움을 나타내며, 다음과 같은 조건을 만족할 때 해당 부클러스터가 입력값에 접근하게 된다.

(조건 1)

$$|x_i - o_j| > |w_{jk} - o_j| \quad (3)$$

(o_j : j 번째 클러스터의 중심벡터)

입력 패턴에 가장 가까운 부클러스터(sub-cluster)로부터 부클러스터가 속해있는 j 번째 클러스터 중심벡터 o_j 사이의 거리가 o_j 로부터 입력패턴까지의 거리보다 커야한다는 것을 의미한다. 만일 (조건 1)이 만족되지 못하면 멤버십의 값은 1로 처리되고 부클러스터(sub-cluster)의 이동도 이루어지지 않는다. 즉 (조건 1)은 어떠한 입력값이 클러스터의 모양을 변형시킬 수 있는가에 관한 규제이며 클러스터

의 모양을 변형시킬 입력값은 현재 부클러스터의 외부에 있는 입력값이라는 것을 의미한다. 이러한 (조건 1)이 만족되면 (조건 2)가 만족되는 지를 판단하여야 한다. 즉, 입력된 값에 따라 모든방향으로 부클러스터를 이동시키면 신경망내의 순서에 따라 배열해둔 부클러스터들의 순서가 망쳐질 수 있기 때문이다. 이러한 규제가 만족되지 못하면 부클러스터의 이동이 이루어지지 않는다.

(조건 2)

부클러스터(sub-cluster) 조정 방향이 그림 5(a), (b)와 같이 되어야 함으로 다음 식 (4-a)나 (4-b)를 만족하여야 한다.

$$\begin{aligned} |\theta_1| &= \left| \cos^{-1} \frac{(w_{jk-1}-o_j)(w_{jk}-o_j)}{|w_{jk-1}-o_j||w_{jk}-o_j|} \right| \\ > |\theta_3| &= \left| \cos^{-1} \frac{(w_{jk-1}-o_j)(x_i-o_j)}{|w_{jk-1}-o_j||x_i-o_j|} \right| \end{aligned} \quad (4-a)$$

or

$$\begin{aligned} |\theta_2| &= \left| \cos^{-1} \frac{(w_{jk}-o_k)(w_{jk+1}-o_j)}{|w_{jk}-o_k||w_{jk+1}-o_j|} \right| \\ > |\theta_3| &= \left| \cos^{-1} \frac{(w_{jk}-o_j)(x_i-o_j)}{|w_{jk}-o_j||x_i-o_j|} \right| \end{aligned} \quad (4-b)$$

$$o_j = \frac{1}{N} \sum_{k=1}^N w_{jk} \quad (4-c)$$

(w_{jk} : j번째 클러스터의 k번째 부클러스터)

(o_j : j번째 클러스터의 중심벡터)

(j-th cluster, k-th sub-cluster, i-th input vector)

식(4-a)와 (4-b)로 인해 부클러스터(sub-cluster)의 진행 방향을 제한함으로써 각각의 부클러스터들이 서로 간섭 받지 않게 된다. 즉 각각의 부클러스터들은 독립적인 클러스터링을 수행하게 된다.

(조건 3)

현재의 입력 패턴에 대해 클러스터들의 소속도가 θ 보다 커야한다. 만일 모든 클러스터내의 모든 부클러스터가 θ 보다 작은 소속도를 나타낸다면 새로운 클러스터를 발생시킨다. 이는 새로운 BCNN을 전체 network에 추가함을 의미한다.

그림 6(a)와 같이 패턴 데이터 X가 입력된 경우 X의 각 클러스터에 대한 소속도가 한계 소속도인 $\theta(0.5)$ 보다 크지 못하므로 A, B 클러스터에 속하지 못하고 새로운 클러스터 C를 만들게되는 과정을 설명한 것이다. 여기서 θ 의 설정은 사용자에게 의해 임의로 설정될 수 있는 변수이다. 그림 6(a)(b)에서 볼 수 있는 것과 같이 A 클러스터내에 8개의 부클러스터가 있으며 부클러스터내의 모든 멤버십 값은 1로 설정되어 있다. 또한 각 클러스터를 표현하는 여러

개의 부클러스터들은 입력 공간에 대해 자신의 독자적인 멤버십 값을 갖추고 있으며, 부클러스터 밖의 멤버십값은 비선형 곡선을 취한다.

만일 위의 조건(1), (2), (3)가 만족하게 된다면 부클러스터의 재조정은 다음 식(5), (6)과 같이 이루어진다.

$$w_{jk} = w_{jk} + \Delta w_{jk} \quad (5)$$

$$\Delta w_{jk} = u_{jk} A(x_i - w_{jk}) \quad (6)$$

여기서 A행렬은 symmetric, positive, definite 행렬이며 m차 특징공간의 각특징들에 대한 가치를 평가하는 스케일링 요소이다. 위와같이 수행한 결과 부클러스터들 사이의 거리가 S보다 커지면 식(8)와 같이 보간을 수행한다. 이를 수행하는 목적은 부클러스터들의 동적인 할당을 통하여 클러스터의 경계선에 대해 좀 더 정확한 표현을 이루기 위함이다. 식(7)는 j번째 클러스터의 k번째 부클러스터와 j+1번째 부클러스터 사이의 길이가 S보다 큰 경우 새로이 생성하여야 할 부클러스터의 갯수를 나타낸다.

$$\eta = \frac{|w_{jk-1} - w_{jk}|}{S} \quad (7)$$

S는 보간하고자하는 단위길이이고, η 는 부클러스터 사이의 길이가 S보다 큰 경우 보간하여 새로이 생성하여야 하는 부클러스터의 갯수를 의미한다. η 번째 새로운 부클러스터는 다음식(8) 처럼 형성된다.

$$w_{jk}^{(\eta\text{-th new})} = \eta \times S \times \frac{(w_{jk-1} - w_{jk})}{|w_{jk-1} - w_{jk}|} + w_{jk} \quad (8)$$

위에서 설명한 BCNN 알고리즘을 정리하면 다음과 같다.

(BCNN 알고리즘)

(step 1) 초기 파라메타 및 부클러스터의 수 결정.

(η, λ, θ, S : 초기 부클러스터 수)

(step 2) 패턴 입력.

(step 3) 식(1)에 의해 u_{ij} 를 결정

(step 4) 클러스터 확장조건 검사.

if(조건(1) 만족 and 조건(2) 만족) (

if(조건(3) 만족) {

식(5), (6) 적용 부클러스터의 위치조정.

goto (step 5)

}

else {

새로운 클러스터 생성

goto (step 2)

}

}

else { goto (step 2) }

(step 5) 클러스터 보간

if(부클러스터들 사이의 거리 > S)

than 식(8)(9)에 의해 보간.
else goto (step 2)

3. 병렬 BCNN으로의 확장

여러 클러스터를 발생한 경우 그림 7와 같은 병렬 구조를 갖는 BCNN이 이루어진다. 이러한 병렬구조의 각 BCNN들은 독립적인 클러스터를 의미하고 각 클러스터내에는 자신의 부클러스터들을 포함하게 된다. 2장의 (조건3)에 의해 새로운 클러스터가 발생하는 경우 새로운 BCNN이 추가로 신경망에 추가된다.

이러한 클러스터링 알고리즘으로 수행되는 과정을 그림 8에 보였다. 2차 특징 공간에서 우선 특징 벡터 a가 입력되고 A 클러스터의 8개 부클러스터들이 8방향에서 단위 거리만큼 떨어진 지점에서 초기화되는 과정을 그림 8(a)에 표현하였다. 그림 8(b)는 특징 벡터 b가 입력되어 A 클러스터의 입력 벡터에 가장 가까운 왼쪽 중앙의 부클러스터의 멤버십값이 0.5를 초과함으로써 A 클러스터가 확장하는 과정을 나타낸 것이다. 그림 8(c)는 특징 벡터 c가 입력되어 새로운 클러스터 B가 만들어지는 과정을 나타내었으며, 마지막으로 그림 8(d)는 이러한 과정을 수행하여 만들어진 클러스터의 모습을 나타낸다.

4. 실험결과

BCNN 클러스터링 알고리즘이 어떤 data군에 대해 적용될 때, 그 결과에 대한 성능 평가는 분류의 평균 에러율로 정의할 수 있다. 이러한 평균 에러율은 적용한 모든 입력 값중 잘못 분류된 입력 값의 비율로서 나타낸다.

Gaussian random variable data군을 분류할 자료로 하여 본 알고리즘과 FMMC와의 성능을 비교 평가하고자 한다. 물론 비교대상의 선정은 순차적인 입력값을 클러스터링 하는 알고리즘들 중에서 선정된 것이다. 우선 가우시안 랜덤 변수(gaussian random variable)를 사용하여 클러스터 a, b, c, d를 만들고 각 클러스터의 confusion matrix를 산정한 것을 그림 9와 표1에 각각 나타내었다. 이러한 confusion matrix의 구성은 우선 각 입력 값에 소속 표시를 붙이지 않은 상태에서 불규칙적인 순서로 입력을 허용한 다음 해당 클래스에 속하는 클러스터들을 분류하고 이러한 클러스터들에 속한 입력 값의 소속을 조사함으로써 이루어졌다. 그림 9에서 수행한 실험의 초기값은 $\eta(0.10)$, $\lambda(1.0)$, $\theta(0.5)$, 초기 부클러스터 8개, S(10)으로 초기화 하였다. 실험에서 알 수 있는 것은 클러스터의 모양에 따라 다르지만 초기 부클러스터의 수는 4개이상이면 안정된 클러스터링을 수행함을 알 수 있다. 특히, 클러스터의 생성에 관련된 η , λ , θ 의 설정이 중요하다. 각 파라미터의 결정 순서는 다음과 같다. 1) 먼저 θ 값을 결정한다. (보통 0.5가 적당하다.) 2) 멤버십 값을 결정하는 η , λ 를 정한다. λ 를 먼저 결정하고(1.0가 적절하다) η 의 값을 정한다. 이때 η 의 값은

식(1)의 1/2 거리 지점 설정이라는 개념하에서 멤버십 값 0.5가 되는 거리를 정하게 된다. 실험에서는 거리 20인 지점을 멤버십 0.5가 되도록 한 것이다. 그림 10과 표2는 FMMC에 의한 클러스터링 결과를 나타낸 것이다. 동일한 입력 값들을 사용하였고 단지 멤버십 값을 산출할 때만 입력 값을 정규화 하였다. FMMC에서는 m차 특징 벡터가 입력 값으로 주어진 경우 모든 차수에 대해 hyperbox가 겹쳐짐을 허용하지 않는다. 따라서 수축(contraction)을 수행하게 되며, 이러한 수축 과정을 통해 기존에 학습하였던 데이터를 보존하지 못하게 된다. 이러한 과정은 그림 10에서와 같다. 그러므로 FMMC 알고리즘은 클러스터의 경계선에 대해 잘못된 판단할 가능성이 많은 것을 알 수 있다. (표 2) 이때, FMMC의 최대 크기 파라미터를 15로 하였다.

그림 11은 BCNN의 η 를 0.15으로 한 경우의 클러스터링 결과이며, 그림 12는 FMMC의 최대 크기 파라미터를 25로 한 경우의 결과이다.

5. 결론 및 고찰

본 논문에서는 입력이 순차적으로 주어질 경우 클러스터 생성과 클러스터의 경계를 나타내는 부클러스터의 확장 및 생성을 통해 클러스터의 모양을 형성하는 클러스터링 알고리즘을 제시하였다. 또한 본 논문을 통해 FMMC의 단점이었던 hyperbox의 형태 및 크기의 제한성을 경계선을 학습하는 부클러스터 확장을 도입함으로써 극복하였고 따라서 risk decision area가 기존의 FMMC에서 보다 상당히 줄어들 수 있었다. 그러나 본 논문에서 제시하는 BCNN 알고리즘은 계층적인 클러스터링 중 초기 클러스터링을 목적으로 하였으므로 궁극적인 클러스터링을 수행하기 위해서는 이러한 초기 클러스터링을 통해 생성되어진 클러스터를 입력값으로 하는 상위의 클러스터링이 필요하다.

(참고 문헌)

- [1] J.C.Bezdek, "Pattern Recognition with Fuzzy Objective Function Algorithms.", New York:Plenum Press, 1981.
- [2] R.N. Dave and K.Bhaswan, "Adaptive C-shell Clustering and detection of ellipses," IEEE Trans. Neural Networks, vol.3, no.5, pp.643-662, Sept. 1992
- [3] J.C.Bezdek and R.H.hathaway "Numerical convergence and interpretation of the fuzzy c-shells clustering algorithm," IEEE Trans. Neural Networks, vol.3, pp.787-793, Sept.1992
- [4] R.Krishnapuram, H.Furugui, "The fuzzy shell clustering algorithms for boundray detection and pattern recognition," in Proc.SPIE Conf.Intelligent Robots and Computer Vision X: Algorithms and Techniques (Boston) Nov. 1991, pp. 458-465

[5] R.N.Dave and K. Bhaswan, "Fuzzy-shell clustering and applications to circle detection in digital images," Int.J.General Systems, vol.16, pp.343-355, 1990

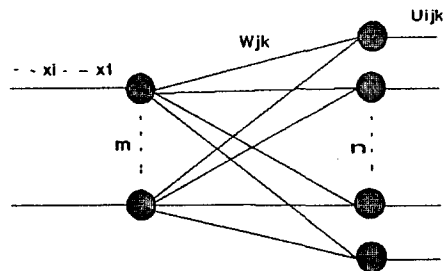
[6] R. Krishnapuram and O. Nasraoui "The Fuzzy C Spherical Shells Algorithm: A New Approach," IEEE Trans. Neural Networks, vol.3, no.5, pp.787-793, Sept.1992

[6] R.N.Dave, "New measures for evaluating fuzzy partitins induced through C-shells clustering," in Proc.SPIEConf, Intelligent Robots and Computer Vision X: Algorithms and Techniques (Boston) Nov.1991, pp.406-414.

[7] J.Davis, T.Warms, and W.Winters, "A neural network implementation of the fuzzy c-means clustering algorithm," in Proc. Int. Joint Conf Neural Networks(Seattle), vol.2, 1991, p.A-953

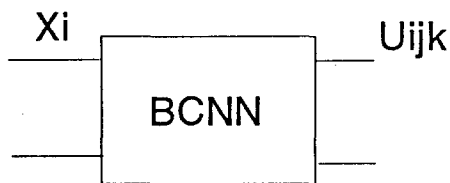
[8] R.Krishnapuram, and J.M.Keller, "A Possibilistic Approach to Clustering," IEEE transactions on Fuzzy Systems, vol. 1, NO. 2, May 1993

[9] P. Simson, "Fuzzy Min-Max Newral Networks-Part 2: Clustering," IEEE transactions on Fuzzy Systems, vol. 1, NO. 1, May 1993



j : cluster vector k: sub-cluster vector
i : input vector

(a) BCNN 구조



(b) 논리적인 표현

그림 3 기본적인 클러스터링 신경망(BCNN)

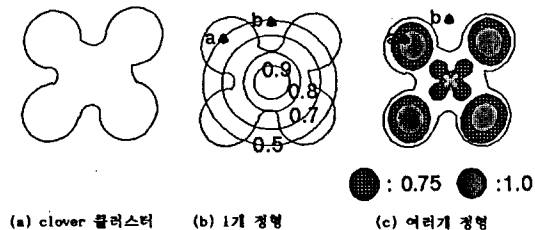
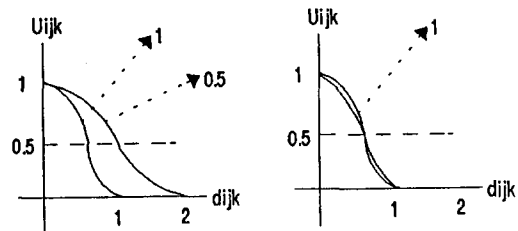


그림 1 membership 함수 분포도



(a) λ 값에 따른 멤버십값의 변화 (b) γ 값에 따른 멤버십값의 변화

그림 4 파라메타 변화에 따른 멤버십값 설정

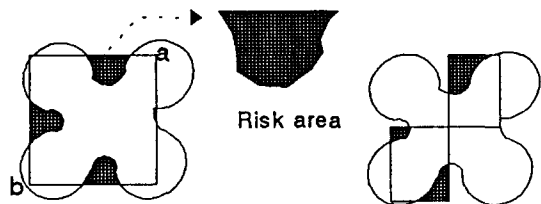


그림 2 hyperbox에 의한 risk decision area

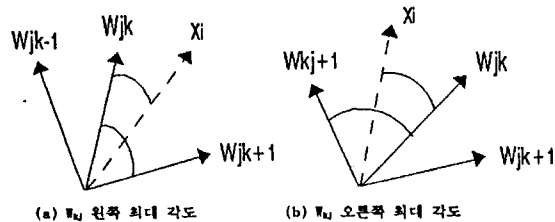
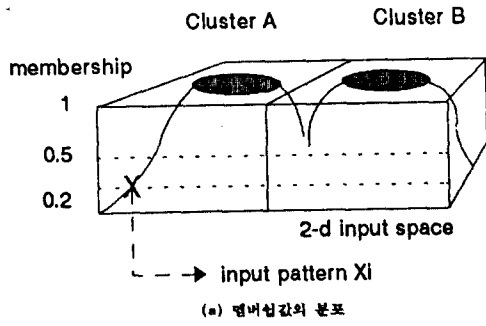


그림 5 부클러스터의 조정 방향의 제약



expansion of cluster A

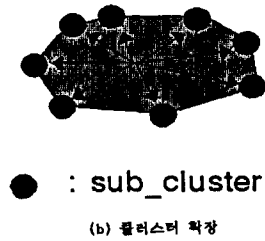


그림 6 클러스터내의 멤버십값의 분포 및 클러스터 확장

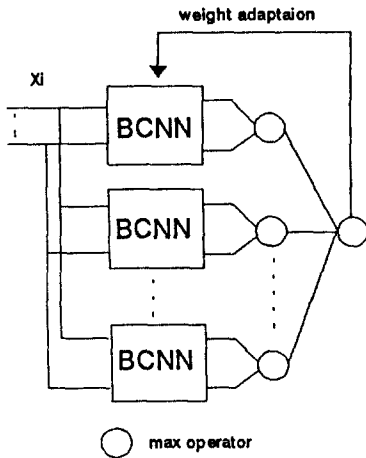


그림 7 parallel BCNN의 구성



(a)



(b)

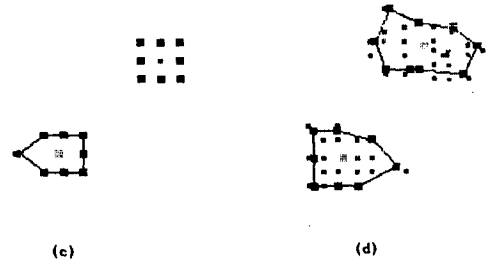


그림 8 클러스터의 확장과정

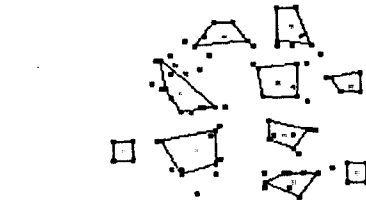


그림 9 gaussian random data 군에 대한 BCNN 클러스터링
($\eta(0.10)$, $\lambda(1.0)$, $\theta(0.5)$,
초기 부클러스터 4개, $S(10)$)

	class 1	class 2	class 3	class 4
class 1	92 %	6 %	0 %	2 %
class 2	0 %	100 %	0 %	0 %
class 3	0 %	2 %	96 %	2 %
class 4	4 %	2 %	6 %	88 %

표 1 gaussian random data 군에 대한 BCNN confusion matrix, 행 : 기준 class, 열 : 할당 class

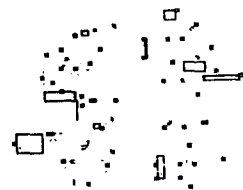


그림 10 gaussian random data 군에 대한 FMC 클러스터링
(클러스터 최대 크기 : 15)

	class 1	class 2	class 3	class 4
class 1	72 %	10 %	2 %	20 %
class 2	14 %	80 %	2 %	2 %
class 3	0 %	2 %	90 %	8 %
class 4	4 %	2 %	16 %	78 %

표 2 gaussian random data 군에 대한 FMC confusion matrix, 행 : 기준 class, 열 : 할당 class

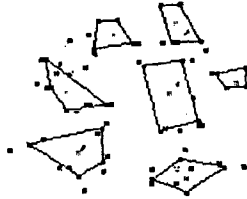


그림 11 gaussian random data 군에 대한 BCNN 클러스터링
 ($\eta(0.15)$, $\lambda(1.0)$, $\theta(0.5)$,
 초기 부클러스터 4개, $S(10)$)



그림 12 gaussian random data 군에 대한 FMC 클러스터링
 (클러스터 최대 크기 : 25)