적응적인 초기치 설정을 이용한 Fast K-means 및 Fuzzy-c-means 알고리즘

(A Fast K-means and Fuzzy-c-means Algorithms using Adaptively Initialization)

강 지 혜 † 김 성 수 ††
(Jee-Hye Kang) (Sung-Soo Kim)

요 약 본 논문에서는 K-means 또는 Fuzzy-c-means 알고리즘에서 클러스터의 중심점을 찾는 과정 중 임의로 선택되는 초기값 선정의 문제를 해결하고, 기존의 단점을 보완하는 새로운 방안으로서 데이타의 분포의 통계적 특성에 따른 초기값 선정 방법을 제안하였다. 기존의 초기값 선정 방법은 초기값에 따라 클러스터링이 매우 민감한 변화를 가져와, 최종적으로 종종 원치 않는 방향으로 가는 문제점을 갖고 있다. 이러한 초기값 선정의 문제가 인지되어 왔지만, 그 문제의 해결방안이 실제적으로 모색된 경우는 없었다. 본 논문에서는 데이타의 통계적 특성을 이용한 초기값 선정 방법을 적용하여, 클러스터링이 형성되는 시간의 단축 및 원치 않는 결과가 생성되는 경우를 약화시켜 시스템의 향상을 가져왔고, 이러한 제안된 알고리즘의 우수성을 기존의 알고리즘과 비교를 통하여 나타내었다.

키워드 : K-means 와 Fuzzy-c-means, 클러스터링, 초기값 선정문제, 균등분할법

Abstract In this paper, the initial value problem in clustering using K-means or Fuzzy-c-means is considered to reduce the number of iterations. Conventionally the initial values in clustering using K-means or Fuzzy-c-means are chosen randomly, which sometimes brings the results that the process of clustering converges to undesired center points. The choice of intial value has been one of the well-known subjects to be solved. The system of clustering using K-means or Fuzzy-c-means is sensitive to the choice of intial values. As an approach to the problem, the uniform partitioning method is employed to extract the optimal initial point for each clustering of data. Experimental results are presented to demonstrate the superiority of the proposed method, which reduces the number of iterations for the central points of clustering groups.

Key words : K-means and Fuzzy-c-means Clustering, Initial Value, Uniform partitioning

1. 서 론

데이타의 클러스터링의 방법에 있어서 많은 연구가들의 관심을 끌어 온 것 중에 하나가 데이타의 비선형 클러스터링 방법이다[1-3]. 이는 패턴인식, 자동제어 및 영상처리 등의 많은 분야에 응용되어 왔으며[4], 정보사회로의 변화에 따라 그 중요성이 더하여지고 있다. 클러스터링의 필요성이 증가함에 따라, 신호처리 및 자동제어 등의 여러 분야에서 널리 응용되는 K-means와 Fuzzy c-means의 성능을 향상시키기 위한 많은 이론과 기법들이 연구되었다[4,5]. 이러한 연구의 필요성은, K-means와 Fuzzy c-means의 이론이 실제로 여러 분야에서 응용되어옴에 따라, 그 중요성이 증가하는 추세에 상응하고 있다[6]. 클러스터링 과정 중에서, 초기값 설정의 중요성은 오래 전부터 널리 인식되어 왔고, 아울러 그 해결 방안의 필요성이 더욱 증대되어 왔다. 그러나 타당성 있는 초기값 선정의 문제는 초기값 선정의 중요성이 날로 증대되어 왔으나, 클러스터링 알고리즘에서의 초기값 선정에 대한 연구는 그리 활발하지 않았다[7-9]. 근본적으로 K-means나 Fuzzy c-means의 초기값 설정은 전체 알고리즘의 최종적인 수행 과정에 밀접한 양상을 보이고 있다. 특히, 데이타 클러스터링의 과정 중 핵심을 이루는 반복 계산 량을 감소시킬 수 있는 중요한 요소가 된다. 그럼에도 불구하고, 기존의 방법은

† 비 회 원 : 충북대학교 전기공학과
k2351181@hotmail.com
†† 정 회 원 : 충북대학교 전기공학과 교수
sungkim@chungbuk.ac.kr
논문접수 : 2003년 2월 4일
심사완료 : 2004년 1월 7일

관측된 데이타에서 단순히 임의로 초기값을 지정해 주었고. 이렇게 임의로 설정된 초기값의 영향은 반복 계산양의 한계를 가져올 뿐만 아니라, 최종 평균값(means) 역시 오차가 커지게 되는 결과를 가져 왔다 [10-13]. 그러한 문제의 해결방안으로서 본 논문이 제안하는 것은, 클러스터링의 대상이 되는 데이타 분포에 따른 해석을 통하여 초기값을 설정함으로써, 효율적인 클러스터링이 이루어지게 하는 방법이다.

일반적으로 K-Means 클러스터링에서는 지정된 클러스터 개수만큼 초기값을 랜덤하게 설정한다. 이러한 초기값 설정은 클러스터링이 시작하는 단계로써, 전체적인 알고리즘에서 매우 중요한 변수로 작용된다[12,13]. 데이타의 분포와 상관관계가 적은 초기값이 설정되는 경우에는 알고리즘을 수행하기까지는 많은 계산과정이 요구되고, 또한 데이타의 크기가 클 경우에는 수행시간이 길어지므로 초기값에 대한 민감성이 상대적으로 높아지게 된다[4,13]. 따라서, 본 논문에서는 데이타의 통계적 특성에 의한 초기값 설정을 통하여 상대적으로 향상된 결과를 가져오는 알고리즘을 제안하였다.

본 논문의 2장은 기존의 K-means나 Fuzzy c-means 알고리즘에 대해서 설명하였고, 3장에서는 제안된 초기값 선정 방법이 적용된 새로운 시스템을 제시하였다. 4장에서는 클러스터링 과정에서, 기존 방법과 제안된 초기값을 적용한 새로운 방법을 제시된 이론적 정립을 바탕으로 시뮬레이션 통한 결과를 보여주고 비교분석함으로써 본 논문의 객관적인 타당성을 보였다. 마지막 5장에서는 본 논문의 총체적 결론을 맺고 향후 연구되어야 할 방향을 제시하였다.

2. K-means와 Fuzzy-c-means 알고리즘

2.1 K-means 알고리즘

K-means 알고리즘에서는 적용하려는 데이타인 각 개체들을 다음과 같이 벡터로 표현하고 $\mathbf{x}=[x_1, \ldots, x_n]$, 식 (1)은 각 개체들 사이의 Euclidean norm을 나타내고, 식 (2)는 두 벡터 x와 z의 차이로 정의된다[4].

$$\|\mathbf{x}\| = \left[\sum_{i=1}^{n} x_i^2\right]^{1/2} \tag{1}$$

$$\|\mathbf{x}-\mathbf{z}\| = \left[\sum_{i=1}^{n} (x_i - z_i)^2\right]^{1/2} \tag{2}$$

전체적인 알고리즘은 초기화 단계, 개체분산단계, 새로운 클러스터의 중심단계로 나누어 볼 수 있는데, 각 단계의 역할과 수렴성에 관한 사항을 간략히 알아본다. 우선, 관측된 n 차원 데이타의 전체 데이타의 개수를 N이라 가정한다.

첫째, 초기화 단계에서는 생성할 클러스터의 개수 K를 정하고, 각 클러스터에 대한 초기값을 설정하는데 특별한 조건 없이 전체 데이타 중에서 식 (3)과 같이 임의로 선택한다.

$$\{z_1,\ z_2,\ \ldots,\ z_K\} \subseteq S_i \qquad i=1, 2, \ldots, N \tag{3}$$

둘째, 개체분산 단계에서는 각 개체들과 각 클러스터의 중심과의 유클리디안 거리(J)를 식 (4)와 같이 구하고, 이때 개체들은 계산된 거리가 식 (5)와 같이 가장 최소가 되는 클러스터($C_l, l=1,2,\ldots,K$)에 속하게 된다. 식 (4)에서 l와 m은 각각의 클러스터를 의미한다.

$$J_{il} = \|x_i - z_l\|^2 \qquad for\ i=1, 2, \ldots, N\ ,\ l=1,2,\ldots,K \tag{4}$$

$$if\ J_{il} < J_{im} \qquad for\ l,m=1, 2, \ldots, K\ ,\ l \neq m \qquad then\ x_i \in C_l \tag{5}$$

여기서 계산된 거리는 개체간의 유사성과 비유사성을 나타낸다. 개체들 간의 거리는 일반적으로 유클리디안 거리측정 방법을 사용한다.

세 번째 단계는 이전 단계에서 새롭게 구성된 개체들을 가지고, 변화된 클러스터의 중심을 식 (6)과 같이 계산한다.

$$z_l(new) = \frac{1}{N_l}\sum(x_i \in C_l) \quad i=1,\ldots,N\ ,\ l=1,\ldots,K \tag{6}$$

여기서, N_l는 각 클러스터에 새롭게 구성된 총 개체의 수를 나타내고, $x_i \in C_l$는 l번째 클러스터에 속한 개체들을 의미한다. 이러한 클러스터의 중심값 $z_l(new)$이 반복적으로 갱신되는데 그러한 반복에 대한 횟수와 전체 수렴성에 대한 조건이 최종 알고리즘의 결과를 좌우하게 된다.

K-means 알고리즘의 수렴여부에 관해서는 식 (7)과 같이 더 이상 각 클러스터의 중심에 변화가 생기지 않을 때 종료되는데, 만일 클러스터의 중심에 변화가 생겼다면 두 번째 단계로 피드백(feedback)되어 반복된다.

$$If \quad z_l(now) = z_l(new)\ then \quad End \tag{7}$$

위의 방법 이외도 수렴 조건은 앞의 두 번째 단계에서 구한 각 클러스터 중심과 개체들과의 거리계산에서 최소의 값들을 모두 합한 것을 각 반복단계에서의 오차(ε_{iter})로 여길 수 있으므로 식 (8)와 같이 지정된 허용오차 임계치($\varepsilon_{\min}$) 보다 작게 되도록 수렴 조건을 설정할 수도 있다[4].

$$M_i = \min(J_{il}) \qquad l=1, 2, \ldots, K \quad ,\ i=1, 2, \ldots, N$$
$$\varepsilon_{iter} = \sum_{i=1}^{N} M_i$$
$$If \quad \varepsilon_{iter} \le \varepsilon_{\min}\ then \quad End \tag{8}$$

2.2 Fuzzy-c-means 알고리즘

기본적으로는 Fuzzy-c-means 알고리즘은 데이타 집합의 유클리디안 거리를 이용하여 각각의 클러스터를 분할하는 K-means 클러스터링과 유사하다[4]. 다만, 차이점은 데이타 개체들이 각 클러스터에 소속하는지(0 혹은 1)에 대한 정보가 아니라, 각 클러스터에 소속하는 정도(0과 1사이의 실수)를 분할 행렬(U)로 표시한다는 점과 초기 중심을 지정할 때 임의로 만들어준 초기분할 행렬을 가지고, 전체 데이타 분포의 중간 정도에 초기 중심을 설정한다는 점이다.

알고리즘의 구성은 우선, 초기화가 이루어지고, 이로부터 클러스터의 중심을 계산하여 얻은 중심과 각 개체 사이의 유클라디안 거리를 구한 뒤, 새로운 분할 행렬을 갱신한다. 이러한 반복적인 갱신과정을 거치면서, 점차 데이타의 특성에 맞는 중심점으로 수렴하게 된다. 앞에서 제시한 K-means 알고리즘과 다른 초기값 설정의 단계에 대해서 간단한 수식을 통해 살펴본다. 우선, 초기화 단계에서는 생성할 클러스터의 개수 K에 대하여 분할 행렬 U을 0과 1사이의 임의의 값으로 식 (9)와 같이 초기화한다.

$$U = \{\mu_{ij} \mid i = 1, \cdots N \ , \ j = 1, \cdots, K\} \tag{9}$$

식 (10)에서 $\{\mu_{ij} \mid i = 1, \cdots N \ , \ j = 1, \cdots, K\}$는 각 클러스터에 대한 소속의 정도를 나타내고, 분할 행렬(U)을 이루는 원소로서 그 전체의 합은 1이 된다.

$$\sum_{j=1}^{K} \mu_{ij} = 1 \ , \ for\ i = 1, 2, ..., N \tag{10}$$

여기서 K는 총 클러스터의 수, N은 전체 데이타 개체들의 개수이다.

두 번째 중심 계산 단계에서, K-means 알고리즘이 각 클러스터에 속한 데이타 개체를 가지고 클러스터 중심을 구하는 반면, Fuzzy-c-means는 분할 행렬을 가지고 중심을 계산한다. 그런 뒤, 분할 행렬 갱신단계에서는 각 클러스터의 중심값과 데이타 개체들과의 거리계산에서 가중치 역할을 하는 소속의 정도 값을 새롭게 바꾸어 준다. 결과적으로 소속도의 정도가 바뀜에 따라 기존의 분할행렬이 갱신되어진다. K-means 알고리즘에서는 최소의 거리를 만족하는 클러스터에 각 개체를 할당하지만, Fuzzy-c-means에서는 각 클러스터와의 소속의 정도를 가지고, 그러한 과정을 수행한다.

일반적으로는 Fuzzy-c-means에서의 수렴여부는 두 번째 단계에서 구한 유클라디안 거리와 소속도의 정도(분할행렬)를 가지고 목적함수(J)를 계산하는데, 목적함수는 전체 오차를 의미하고, 클러스터의 중심이 변화됨에 따라, 점차 감소하여 수렴하게 된다.

전체 알고리즘은 식 (11)과 같이 기존의 목적함수와 갱신된 새로운 목적함수의 차(ε_{iter})가 지정된 임계치($\varepsilon_{\min}$)보다 작으면 반복과정을 끝내면서 종료된다.

$$\left| J(now) - J(new) \right| = \varepsilon_{iter},$$

$$If\ \varepsilon_{iter} \le \varepsilon_{\min}\ then\ \ End \tag{11}$$

3. 새로운 초기 중심값 설정 방법

본 논문에서는 데이타를 구성하는 임의의 차원의 데이타 공간을 균등 영역 분할하여 클러스터의 초기값을 선정하는 새로운 방법을 제안한다. 제안하는 알고리즘은 임의의 데이타 공간상에 랜덤하게 분포되어 있는 데이타를 적응적으로 균등 분할하는 것을 말한다. 이 방법은 데이타 공간상에 분포되어 있는 각 데이타들의 상대적 위치가 데이타의 회전, 이동 또는 확대, 축소 등에 바뀌지 않는 공간상의 분할을 말한다. 이는 데이타를 구성하는 각 요소들의 상호 위치가 데이타의 회전, 이동, 확대(축소)에 따라 변하지 않는 특성에 기반을 두고 있다. 일반적으로 데이타 공간을 균등 분할하는 경우, 분할된 각 데이타 공간의 단위공간에는 서로 다른 개수의 데이타가 포함되어 있다고 가정할 수 있다. 다시 말해서, 균등 분할에 의해 나누어진 데이타 공간들은 각각의 공간에 속한 데이타 개수를 빈도수로 갖게 되어, 분할된 공간 차원의 데이타 밀도 분포를 이루게 된다. 이러한 일련의 과정을 그림 1의 블록 다이어 그램으로 나타내었다.

우선, 원하는 클러스터의 수 K가 결정된 후에 클러스터링을 하기 위해서는 각 클러스터의 초기값을 설정해야 한다. 본 논문이 제안한 초기값 설정 방법은 그림 1에 나타낸 순서대로 다음과 같이 크게 두 단계로 나누

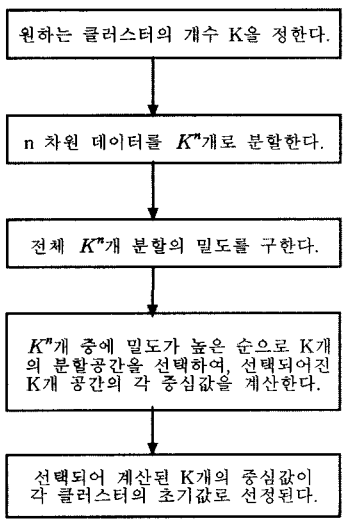


그림 1 제안된 균등 분할법에 대한 알고리즘 순서도

어 설명할 수 있다. 첫 번째로, 변수 X가 클러스터링 알고리즘에 적용하려는 데이타 집합이라면, 이 데이타 X는 N개의 개체로 이루어져 있다고 가정한다. 각 개체는 임의의 n차원의 데이타라면, 전체 집합 X를 이루는 N개의 각 개체는 $x_i=(x_{i1}, x_{i2}, \cdots, x_{in})$, $i=1,2,\cdots,N$으로 나타낼 수 있다. (예를 들어, N=2인 경우에는 평면상의 점을 나타내고, N=3인 경우는 3차원 공간상의 점을 나타낸다.)

물론 원하는 클러스터의 개수는 K≤N라 가정한다. 만약, N개의 데이타를 표현하는데 필요치 않은 차원을 사용하는 경우가 발생할 때, 예를 들어 3차원 공간의 두 점은 2차원 평면상에 상관되는 정보를 가지고 충분히 나타낼 수 있으므로, 임의의 n차원의 데이타원소 N개로 이루어진 데이타 공간을 각 차원마다 K개의 클러스터로 균등 분할하여 나타낼 수 있다. 예를 들면, 2차원 평면상에 N개의 점들로 이루어진 데이타 $x_i=(x_{i,1}, x_{i2,})$, $i=1,2,\cdots,N$ 는 $K\times K$ (=K^n)개로 균등 분할된 임의의 공간상에 속하게 된다. 일반적으로 데이타가 n차원일 경우는 전체 데이타 공간을 K^n 개의 n 차원 균등분할된 부분 공간들로 형성된다. 균등 분할을 하였으므로, 각 부분 공간들은 동일한 공간 면적을 가지고, 임의의 순서로 구분 지을 수 있다. 각각 구분되어진 분할된 공간 내에는 속해 있는 데이타의 개수를 가지고 각 부분 공간에서의 데이타 밀도를 구할 수 있다. 이는 구분되어진 각 부분 공간에서 속해있는 데이타 개체들의 개수는 각각의 부분 공간상의 데이타 밀도라 볼 수 있으므로, 우리가 원하는 클러스터 수 K개만큼을 찾기 위해서 내림차순으로 정렬한다. 정렬된 순서로 즉, 데이타 밀도가 높은 순서부터 K개만큼 선택된 부분 공간들이 전체 데이타 공간에서의 밀집도가 높은 순서로 선택되어진 부분 공간이 된다. 이렇게 선택된 K개의 부분 공간은 클러스터링 알고리즘이 시작될 초기값을 선정하기 위한 부분 공간으로 선택된 것이다. 만일 데이타 밀도가 동일한 여러 분할된 부분 공간들이 존재한다면 데이타 분포의 통계적 특성을 나타내는 분산이 작은 단위 데이타 공간을 우선적으로 선택한다. 그 이유는 분산이 작을수록 데이타의 밀집도가 높아지기 때문이다. 이러한 균등 분할법의 첫 번째 과정으로 초기값을 선택할 부분 공간을 선택하는 것은 제안된 초기값 설정의 가장 중요한 부분이다. 이것은 주어진 데이타의 통계적 특성을 이용하여 데이타 분산이 작은 즉, 정보 공간의 밀집도가 높은 곳에서 클러스터링의 출발 지점인 초기값을 설정하고자 함을 의미한다. 그러한 초기값을 설정하는 이유는 반복되는 알고리즘의 수행시간을 단축시키고 최적화 문제의 가장 큰 단점인 국부 최적점에 도달하게될 위험을 줄일 수 있기 때문이다. 최종적인 클러스터링의 목표는 최종 클러스터의 분산값이 가장 작을 때 이루어지므로 정보공간의 밀집도가 높은 지점의 초기값에서 출발한 경우가 임의로 랜덤하게 지정한 초기값에서 시작된 경우보다 향상된 성능을 보이게 된다.

다음은, 두 번째 단계로 원하는 개수 K만큼 선택된 부분 공간상에서 클러스터링 알고리즘의 시작점인 초기값을 구하는 것이다. 이는 앞 단계에서 선택되어진 각각의 부분 공간에서의 중심값을 구하는 것으로써, 데이타 밀집도가 큰 선택된 K개의 부분공간상에 속해진 데이타들의 평균값을 구하면 곧 각 부분 공간에서의 중심값을 얻을 수 있다. 예를 들어, 첫 번째 과정에서 선택된 K개의 분할공간들의 중심값은 데이타가 n차원일 경우 $(C_{i,1}, C_{i,2,} \cdots, C_{i,n})$, $i=1,2,\cdots,K$ 로 나타낸다. 여기서 $C_{i,j}$ 는 임의로 선택된 i 번째 분할공간 내의 j번째 성분의 평균치이다. 따라서 K개의 데이타 밀도가 높은 분할공간으로 선택되어진 각각의 부분 공간에서의 중심값이 전체 클러스터링 알고리즘의 초기값으로 설정되어 알고리즘을 수행하게 된다.

이러한 방법으로 데이타 클러스터링의 초기 값을 선정하는 것은 데이타의 통계적 특성 중 평균치와 유사성을 소유한다는 사실을 근거로 이를 적절히 이용하는 방법이라 볼 수 있다. 기존의 K-means나 Fuzzy-c-means의 클러스터링 알고리즘에 제안된 데이타 공간의 균등 분할에 의해 계산된 초기값을 적용함으로써 얻어지는 우월성을 시뮬레이션 실험을 통하여 보여준다.

4. 실험 및 고찰

제안된 알고리즘과 기존 알고리즘의 비교를 위한 시뮬레이션에서는 원하는 클러스터의 개수 K를 세 개로 지정하고, 적용할 데이타 집합은 Iris Data를 사용하였다. 그림 2는 K-means 클러스터링 알고리즘을 적용하여 세 그룹으로 나누어진 결과로서, 각 클러스터의 분포와 수렴된 중심값을 나타내고 있다.

여기서, 이용한 초기값 설정 방법은 주어진 데이타를 랜덤하게 조합시킨 후, 원하는 클러스터 개수 K만큼 선택하는 것이다. 이러한 경우, 임의로 조합되어 선택된 값들이 매번 다르므로 안정적인 초기값을 얻기는 불확실하다. 또한, 경우에 따라 최종적인 클러스터링 결과에 이르기까지 반복 과정이 비효율적으로 많아지는 경우가 있다.

표 1은 기존의 방법대로 임의로 선정된 초기값을 사용하여 총 100회에 걸친 시뮬레이션을 통해 매번 랜덤한 초기값에서 최종 중심값을 얻는 데까지 소요된 반복 횟수에 대한 빈도수를 나타내고 있다. 그림 3의 (a)는

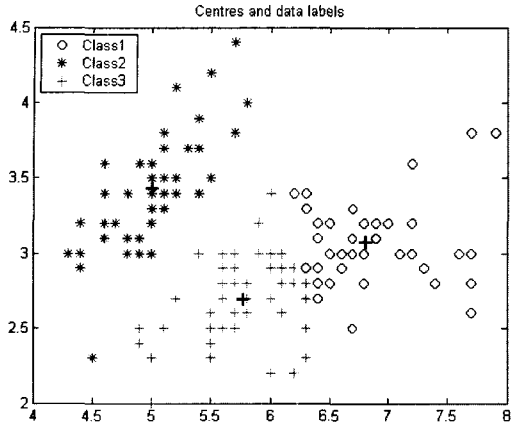
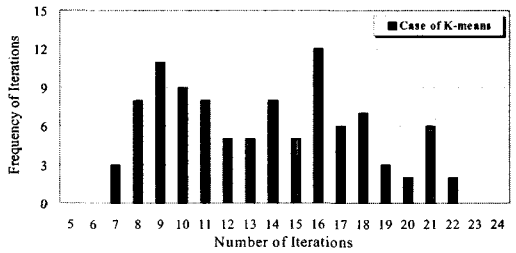
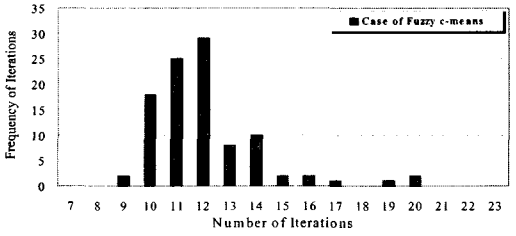


그림 2 K-means를 적용한 결과 (Iris 데이타)



(a) K-means



(b) Fuzzy c-means

그림 3 기존의 방법에 의한 반복계산 횟수의 빈도수

표 1 기존의 방법(랜덤)에 의한 시뮬레이션 반복 횟수 결과 비교

	기존의 방법 (Random)	제안된 방법 (데이타분포의 영역할당법)
K-means	Min = 7 Max = 22 Mean = 13.61 회	5 회
Fuzzy c-means	Min = 9 Max = 20 Mean = 12.03 회	7 회

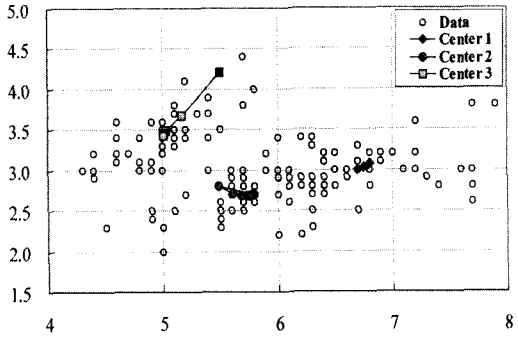
K-means 방법, (b)는 Fuzzy-c-means 클러스터링 방법을 각각 Iris 데이타에 적용시킨 경우의 반복횟수에 대한 빈도수를 보여주고 있다. 그림 3(a)에서, K-means 방법의 시뮬레이션의 경우, 가장 적은 7회의 반복계산이 이루어진 경우가 3%, 가장 높은 빈도수의 반복 회수는 16인 경우로 12%이다. 다시 말해서, 빈도수가 표 1에서 보인 바와 같이 K-means 방법의 경우 반복 횟수가 7회에서 22회까지 다양하고, 평균적으로 13.61회의 반복을 나타내고 있다. 또한, Fuzzy-c-means의 방법 역시 반복횟수가 9에서 20회까지 다양한 반복 횟수를 보인다. 여기서 각각의 기존 방법들이 최소 7회와 9회의 횟수를 가진다고 볼 때, 상대적으로 제한된 방법은 적용한 K-means의 경우는 5회, Fuzzy-c-means의 경우는 7회의 반복 횟수를 필요로 한다.

본 논문에서 제안한 초기 값 선정을 적용한 클러스터링 알고리즘을 동일한 Iris Data를 가지고 시뮬레이션을 실행하였다. 데이타 분포 공간의 균일 분할법을 이용한 최대 빈도 수 정렬에 의해 구해진 초기값 선정방법으로 K-means 알고리즘의 결과를 기존 방법과 비교해 본다. 그림 3은 초기값에서부터 최종 중심값을 찾는 과정을 기존 방법과 제안된 방법 모두 차례로 보이고 있다.

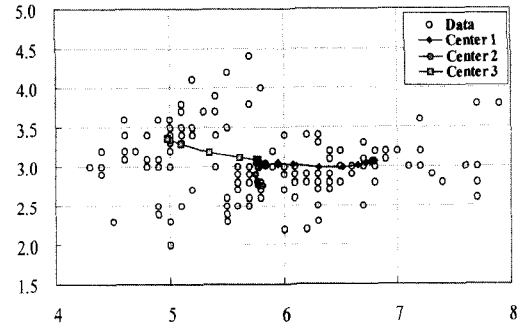
그림 3의 (a)는 기존의 K-means 방법으로 100회의 시뮬레이션 중에서 가장 적은 반복회수를 나타낸 7회의 경우로 초기값에서 최종 중심값을 찾아가는 경로를 보이고 있다. 그림 3의 (b)는 Fuzzy-c-means 방법의 경우로, 역시 100회의 시뮬레이션 중에서 가장 적은 9회의 반복회수를 지닌 경우로 클러스터링의 진행 과정을 보이고 있다. 반면에, 표 2는 제안된 방법이 적용된 초기값에서 최종 중심값까지 K-means와 Fuzzy-c-means의 경우로 나누어 제시하고 그러한 클러스터링 수행 과정을 그림 4의 (a)와 (b)에서 각각 나타내고 있다.

그림 5는 임의의 Fuzzy-c-means의 경우로, 한 번 알고리즘이 수행되기까지의 반복 과정 동안 오차를 의미하는 목적함수의 수렴과정을 나타내고 있다. 기존의 초기값을 이용한 경우는 서서히 감소하는 반면에, 제안된 방법의 초기값을 적용한 경우는 처음에는 오차가 크게 나타나고 있지만, 두 번째부터는 급격히 감소하고 있다. 결국, 제안된 시스템의 수렴성이 기존의 시스템보다 우월한 특성을 갖고 있다고 볼 수 있다.

그림 5에서 보이는 반복횟수에 따른 목적함수의 변화에 대해 좀 더 자세히 살펴본다. 기존방식대로 임의의 초기 분할행렬을 가지고 계산된 목적함수는 종료조건 ($\varepsilon=0.01$)을 만족하기까지 반복 횟수와 제안된 방법을 적용한 결과와 비교해보면, 최종 수렴되기까지 반복 횟수가 감소한 것을 알 수 있다. 데이타의 통계적 성질을 이용한 제안된 방법과 임의로 생성된 분할행렬을 초기값으로 하는 기존방법 중 초기 목적함수의 값이 기존의 방법이 더 낮은 것을 볼 수 있다. 그것은 당연한 결과로, 기존의 방법에서 지정된 초기값은 데이타분포에서



(a) 기존의 K-means



(b) 기존의 Fuzzy c-means

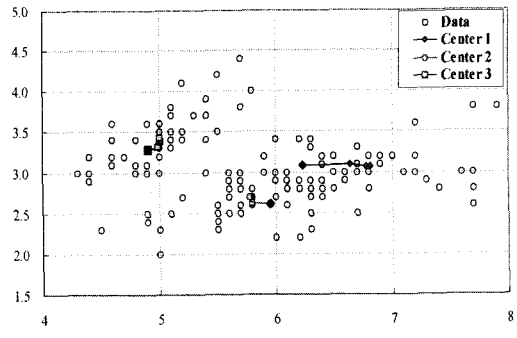
그림 4 기존의 방법에 의한 초기중심설정과 중심변화과정

표 2 제안된 방법에 의한 시뮬레이션 결과 비교

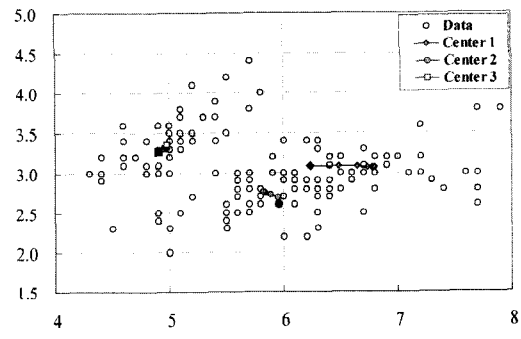
	초기 중심값		최종 중심값		반복횟수
K-means	6.2368	3.0868	6.8128	3.0745	5
	5.9583	2.6056	5.7736	2.6925	
	4.9027	3.2784	5.0060	3.4280	
Fuzzy c-means			6.8022	3.0690	7
			5.8236	2.7597	
			4.9794	3.3557	

중간값을 나타내므로, 각 데이타들과 초기값 사이의 거리 합이 작게 나타나기 때문이다. 반면에, 처음의 목적함수 값이 비록 높다고 하여도, 새로 갱신된 다음 번 과정에서 계산된 목적함수는 뚜렷한 감소를 보인다. 이것은 제안된 방법에 의한 초기 값의 역할이 기존과 다르다는 사실과, 그것에 의한 효과로 볼 수 있다. 이런 과정을 계속 거치면서 최종 종료에 이르기까지 제안된 방법이 비교적 더 빨리 수렴됨을 알 수 있다.

다음은 기존의 초기값 설정방법으로 인해서 클러스터링의 결과가 원치 않은 방향으로 초래되는 경우에 있어서의 제안된 방법과 기존의 방법을 비교하였다. 초기값 설정으로 원치 않은 클러스터링의 결과를 가져오는 경우는 충분한 반복 과정이 제한되어 국부 최소 점에 도달할 때이다. 여러 분야에서 응용되고 있는 K-means나



(a) 제안된 방법의 K-means 중심 변화



(b) 제안된 방법의 Fuzzy c-means 중심 변화

그림 5 제안된 방법에 의한 초기중심설정과 중심변화과정

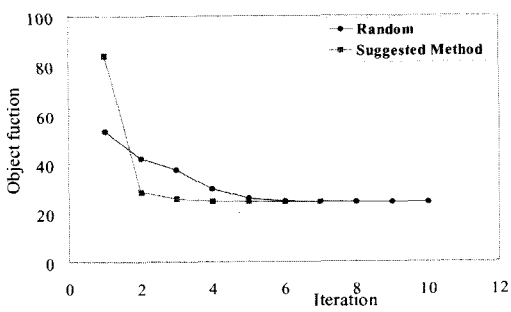


그림 6 목적함수(Object function) 값의 변화

Fuzzy-c-means알고리즘은 대부분 그 반복 횟수가 한정적으로 지정되어 최적 해에 도달하지 못하는 경우가 많다[4,12]. 이것은 무한 반복이 일어나는 경우를 사전에 대비하는 것일 수도 있지만, 반복 시간을 단축시키므로 알고리즘의 수행시간을 줄이려는 목적에서이다. 이러한 경우는 충분한 반복을 통한 최소의 에너지 지점으로 도달할 수 없으므로, 그 초기값의 역할이 더욱 중요해진다. 다음에 제시된 표 3은 반복회수가 5번으로 제한된 경우, 기존의 방법과 제안된 방법과의 클러스터링 결과

와 오차를 나타낸 것이다. 여기서 제한된 반복횟수는 기존의 방법에서와 제안된 방법에서의 평균적인 반복횟수보다 더 적은 것으로써, 적은 반복과정으로 제한된 경우에 있어서 그 초기값의 영향을 보여주고 있다. 기존의 경우는 매번 수행 할 때마다 그 값이 변하게 되는데 총 100회의 수행결과 중 높은 빈도수를 가지는 2가지의 경우와 제안된 방법을 가지고 비교해 보았다.

표 3에서 제시된 오차는 다음 반복과정을 수행하기 전에 현재 생성된 클러스터의 중심값으로부터 각 데이타와의 유클리디안 거리의 제곱의 전체 합을 나타낸 값들이다. 이러한 오차는 각 클러스터의 중심값과 데이타들과의 분산이 기존 방법의 결과보다 제안된 방법의 결과가 더 작게 나타나고 있다. 또한 실제 데이타 분포의 중심값 (0,0) (2,3.5), (0,2)과 비교해 볼 때 기존 방법의 결과 보다 제안된 방법의 결과가 더 적은 오차를 갖는다.

다음의 그림 7과 그림 8은 각각 표 3에서 기존 알고리즘(1), 기존 알고리즘(2)의 경우로 임의의 초기값에 의한 클러스터 분포와 최종 제한된 조건까지의 5회 반복 과정을 거친 클러스터링 결과를 보여준다.

다음의 그림 9는 제안된 방법에 의한 초기값에 의한 클러스터링의 분포와 5회의 제한 반복 과정의 클러스터링 결과를 보여주고 있다. 그림 9에서 보듯이, 제안된

표 3 제한된 조건하에서 기존방법과 제안된 방법의 비교

	기존의 K-means 알고리즘(1)	기존의 K-means 알고리즘(2)	제안된 K-means 알고리즘
클러스터링 결과 중심값 (제한반복횟수 : 5회)	-0.3399 0.4867 1.6982 3.3039 0.5526 -0.0871	-0.0121 0.3142 1.9877 3.3121 0.9942 3.4419	0.0421 -0.0288 1.9271 3.4272 -0.1385 1.8886
오차	402.2904	452.7246	272.217

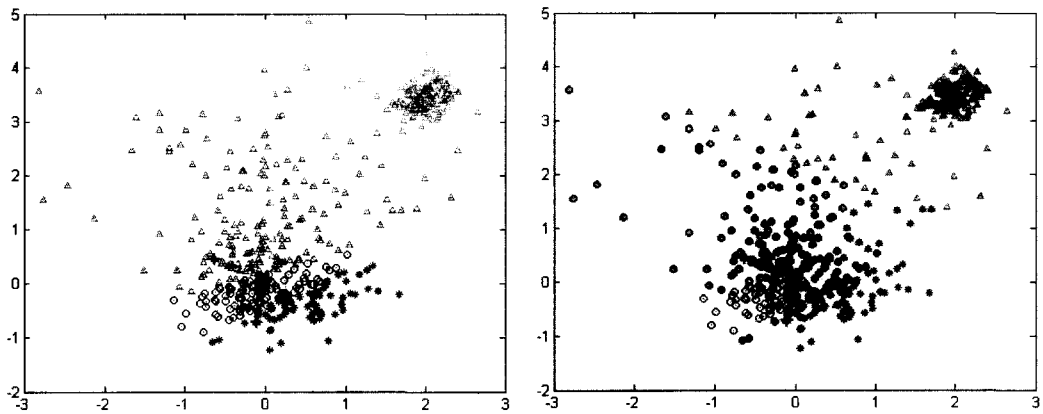


그림 7 기존 알고리즘(1)의 초기값에 의한 분포와 제안된 횟수의 최종 클러스터링 분포

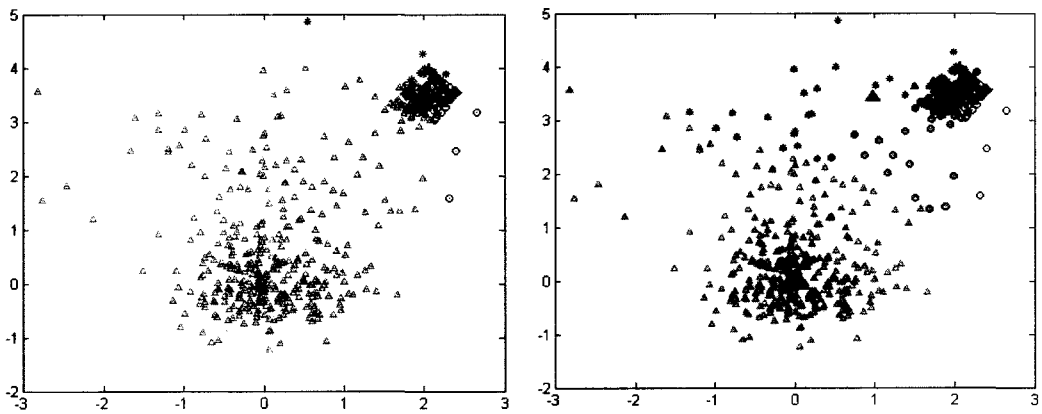


그림 8 기존 알고리즘(2)의 초기값에 의한 분포와 제안된 횟수의 최종 클러스터링 분포

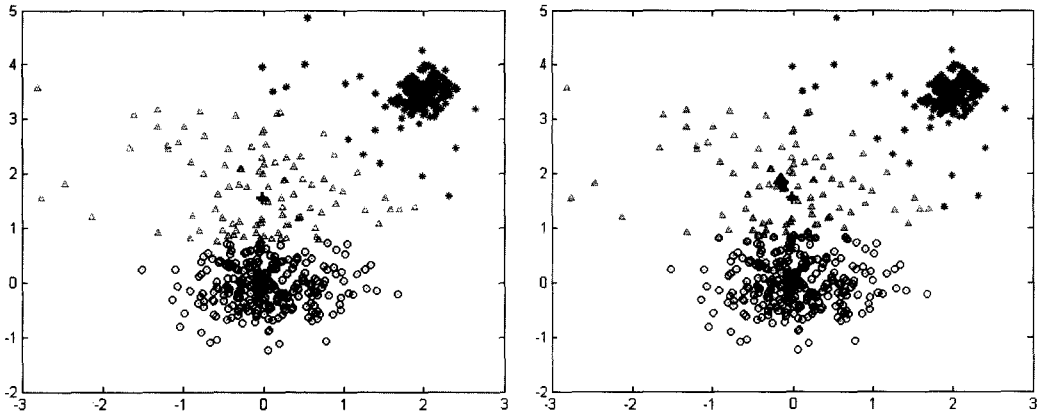


그림 9 제안한 방법의 초기값에 의한 분포와 제안된 횟수의 최종 클러스터링 분포

표 4 여러 타입의 데이타분포에 따른 결과비교

데이타 분포	(평균)반복횟수		오차값 (제한조건 : 5회)	
	기존의 방법	제안된 방법	기존의 방법	제안된 방법
full	14	9	622.8678	478.0836
spherical	13	8	402.2904	272.2170
diagonal	12	8	485.7785	408.3695

알고리즘에 의한 초기 클러스터 분포는 최종 결과와 거의 일치됨을 볼 수 있고, 따라서 반복 횟수가 제한된 조건 환경에서의 초기값 영향에 대한 중요성을 상대적으로 나타내고 있다.

다음 표 4에서는 Iris 데이타 이외 여러 다른 타입의 분포 특성을[11] 지닌 데이타들의 시뮬레이션 결과에 대한 비교를 통해 본 논문이 제안한 알고리즘의 우수성을 보였다. 표 4에 제시된 평균 반복횟수는 기존의 방법에서 매번 종료 반복횟수가 달라지므로 총 100회 시뮬레이션 했을 경우 그 평균값을 말하는 것이고, 비교되는 오차는 앞에서 언급한 바와 같이 반복횟수에 대한 제한조건이 있는 경우, 최종 중심값으로부터 데이타들과의 분산을 나타낸다.

6. 결 론

본 논문에서는 신호처리 및 자동제어 등의 여러 분야에서 응용되는 K-means 또는 Fuzzy c-means의 초기값 설정에 대한 이론과 기법에 대하여 연구하였다. 임의로 선택되는 기존 초기값 선정에 의한 문제점과 그 원인을 분석하여 해결방안으로서 알고리즘의 대상이 되는 데이타 분포에 따른 통계적 특성을 이용한 방법을 제시하였다. 제안된 방법으로 반복 계산과정을 단축시킴으로써 효율적인 알고리즘 수행의 향상된 결과를 보였다. 이처럼, 초기값 선정 문제는 알고리즘 수행과정과 결과 획득에 있어 매우 중요한 요소가 된다. 앞으로 이런 관점에서의 연구가 좀더 활발히 진행될 것이며, 여러 가지의 새로운 방법들이 추가로 제시될 필요가 있다.

참 고 문 헌

[1] C. N. Schizas and C. S. Pattichis, "Neural networks, genetic algorithms and the K-means algorithm: in search of data classification," COGANN-92. International Workshop, pp. 201-222, Jun 1992.

[2] R. N. Dave, "Robust fuzzy clustering algorithms," IEEE Fuzzy Systems International Conference, vol. 2, pp. 1281-1286, 1993.

[3] G. Beni and Liu Xiamoin, "A least biased fuzzy clustering method," IEEE Pattern Analysis and Machine Intelligence Trans, vol. 16, pp. 954-960, Sep 1994.

[4] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995.

[5] M. Singh and P. Patel, and D. Khosla, "Segmentation of functional MRI by K-means clustering," IEEE Nuclear Science Symposium and Medical Imaging Conference, vol. 3, pp. 1732-1736, Oct 1995.

[6] Timothy J. Ross, Fuzzy Logic with Engineering Application, McGraw-Hill, Inc., 1995.

[7] R. Krishnapuram and J. Kim "Clustering algorithms based on volume criteria," IEEE Fuzzy

Systems Trans, vol. 8, pp. 228-236, Apr 2000.
[8] Wong Ching-Chang and Chen Chia-Chong, "K-means-based fuzzy classifier design," IEEE Fuzzy Systems International Conference, vol. 1, pp. 48-52, May 2000.
[9] K. K. Paliwal and V. Ramasubramainan, "Modified K-means algorithm for vector quantizer design," IEEE Image Processing Trans, vol. 9 pp. 1964-1967, Nov 2000.
[10] S. S. R. Abidi and J. Ong, "A data mining strategy for inductive data clustering: a synergy between self-organising neural networks and K-means clustering techniques," TENCON 2000. Proceedings , vol. 2, pp. 568-573, 2000.
[11] Ian T. Nabney, NETLAB Algorithms for Pattern Recognition, Springer, 2001.
[12] R.O. Duda and P.E. Hart, Pattern Classification, Willey, 2001.
[13] Su Mu-Chun and Chou Chien-Hsing, "A modified version of the K-means algorithm with a distance based on cluster symmetry," IEEE Pattern Analysis and Machine Intelligence Trans, vol. 23, pp. 674-680, Jun 2001.

강 지 혜
2003년 2월 충북대학교 전기전자 컴퓨터 공학부 졸업(공학사). 2003년 5월~현재 충북대학교 전기공학과 석사과정. 관심분야는 통계 신호처리, 패턴인식, 인공지능, 디지털 통신

김 성 수
1983년 2월 충북대학교 전기공학과(B.S) 1989년 2월 University of Arkansas-Fayetteville(M.S.). 1997년 12월 University of Central Florida (Ph.D.). 1998년 2월~1999년 3월 시스템공학연구소/전자통신연구원. 1999년 3월~2001년 8월 우석대학교 전기공학과 조교수. 2003년 5월~현재 충북대학교 전기공학과 조교수. 관심분야는 신호처리, 통신이론, 인공지능, 해석학