Proceedings of KFIS Autumn Conference 2006 Vol. 16, No. 2.

초기 클러스터를 위한 확장 클러스터링

Expansion Clustering For Initialized Set

이재성[1], 김대원[2]

[1] 서울시 동작구 중앙대학교 컴퓨터공학과
E-mail: curseor@hotmail.com
[2] 서울시 동작구 중앙대학교 컴퓨터공학과
E-mail: dwkim@cau.ac.kr

요 약

본 논문에서는 사용자가 결과를 얻고자 하는 목적 집단의 초기 클러스터를 생성하는 알고리즘을 제안한다. 알고리즘이 생성하는 클러스터는 사용자의 입력을 받지 않고 생성되며, 목적 집단에 포함되는 임의의 두 점을 이용한 확장을 통해 초기 클러스터를 생성한다. 이에 따라 서로의 영역을 침범하지 않는 일반적인 클러스터를 생성하는 것이 가능하다

Key Words : Hyper-Ellipse Expansion, K-means, Fuzzy c-means, Hierarchical clustering

1. 서 론

전통적인 클러스터링 알고리즘들은 일반적으로 각 점 사이의 거리에 의한 클러스터를 생성하고, 사용자가 입력한 클러스터의 개수를 만족할 때까지 병합한다. 그러나 사용자의 입력을 받아야 하는 문제는 사용자가 목적 집단이 몇 개의 클러스터로 이루어져 있는가를 이미 알고 있음을 전제하므로, 대량의 목적 집단이 주어졌을 때 사용자의 입력을 기대하기는 어렵다.

이러한 문제를 해결하기 위해서는 사용자의 입력 없이 클러스터를 생성해야 하지만, 이렇게 생성된 클러스터의 경우 일반적으로 사용자가 원하는 개수의 클러스터로 생성되지 않는다.

이에 따라 생성된 클러스터를 분할하거나 병합할 필요가 있는데, 본 논문에서는 분할이 더 이상 필요 없으되 병합은 필요한 초기 클러스터를 만드는 알고리즘을 제안하며, 제안된 알고리즘은 다음의 사항을 만족한다.

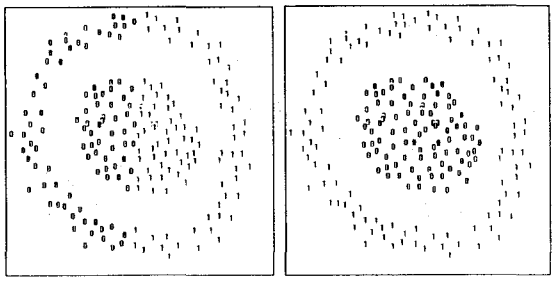
(1) 모든 거리 계산 방식에 무관하다.
각 점 사이의 거리 판단을 할 때, 특정 방식에 무관한 알고리즘을 작성한다. 단지 각 점 사이의 상대적인 거리를 판단하는 것이 가능하다면 어떠한 방식을 사용하든 무관하도록 한다.

(2) 클러스터 생성을 위해 필요한, 최소한의 조건을 사용한다.
주어진 목적 집단에서 가장 가까운 거리를 유지하고 있는 2개의 점이 있다고 할 때, 2개의 점이 같은 클러스터에 포함되지 않는다면, 클러스터 생성이 불가능하다. 제안된 알고리즘은 가장 가까운 2개의 점을 같은 클러스터로 인정한다는 조건만을 사용한다.

(3) 형태에 무관한 클러스터 집단을 생성한다.
주어진 점들이 이루고 있는 각도나, 원형이냐 아니냐에 전혀 무관하게 동일한 결과를 얻도록 한다.

(4) 특정 클러스터에 다른 클러스터에 소속되어야 할 점이 포함되지 않는다.

다음과 같은 결과를 생성하는 클러스터링 알고리즘 두 개가 있다.



K-means Hierarchical Clustering
그림 1. 기존 알고리즘들의 결과 비교

그림 1은 임의의 목적 집단에 대해서 전통적인 K-means와 Hierarchical Clustering 알고리즘을 통해서 생성된 클러스터들이다.

Hierarchical Clustering에서는 서로의 클러스터에 포함된 점이 다른 클러스터를 침범하지 않지만, K-means로 생성된 클러스터는 서로의 클러스터를 침범하고 있다.

다시 말해, K-means에 클러스터 0에 혼재되어 있는 클러스터 1의 점들을 새로운 클러스터 2로 만들어준다면, 사용자는 클러스터 0, 1, 2에 대한 특성을 최초의 클러스터 0, 1보다 정확히 알아낼 수 있을 것이다.

본 논문에서는 이러한 클러스터 생성 방법을 '적합한 클러스터를 생성한다'고 정의하였으며, 제안된 알고리즘을 Hyper-ellipse Shaped Expansion Clustering(이하 HESE)이라 한다.

2. HESE

HESE는 전통적인 클러스터링 알고리즘과 달리 완전히 종료된 클러스터를 생성하지는 않는다. 이는 사용자의 입력 없이 생성된 클러스터의 특성에 의해서 사용자에게 틀린 결과를 제시하지 않기 위함이며, 주어진 목적 집단에서 얻을 수 있는 정보를 기반으로 최대한 군집화 시킬 수 있는 점들만을 대상으로 클러스터를 생성하기 때문이다. 또한 HESE를 통해 생성된 클러스터들은 서로의 영역을 침범하지 않으므로, 사용자가 각 클러스터의 특성을 정의하는데 발생할 수 있는 오류를 최소화할 수 있다.

HESE의 종료조건은 다음과 같다.

(1) 주어진 모든 점에 대해서 '확장'이 실행되면 알고리즘은 종료한다.

위에서 제시된 확장은 다음과 같은 순서로 진행된다.

(1) 임의의 점 P_1에 대해서 가장 가까운 점 P_2를 구한 후, 그 두 점사이의 거리를 구한다. 모든 점에 대해서 거리를 구한 후, 평균을 구하고 그 결과에 α배를 한다.

(2) 위에서 구한 결과를 Pre-distance P_D라 정의하고, 저장해둔다.

(3) 주어진 점 P_1을 가지고 가장 가까운 점 P_2를 찾아서 확장을 시도한다. P_1과 P_2를 제외한 이외의 점들 중에서 P_1과 P_2의 거리(이하 $D(P_1, P_2)$)의 α배를 구한 뒤, 대상 점 P_3가 다음 수식을 만족하면 같은 클러스터로 인정한다.

$$D(P_1, P_3) + D(P_2, P_3) \leq D(P_1, P_2) * \alpha$$

위의 수식은 특정 점 P_1이 존재할 때, P_2가 같은 클러스터라는 가정에 기반하며, 이 때 클러스터의 범위는 거리 $D(P_1, P_2)$의 α배임을 나타낸다.

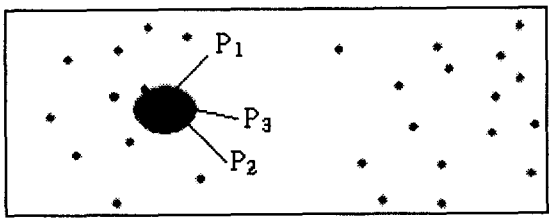


그림 2. 최초 P_1과 P_2를 같은 클러스터로 생성하였을 때, $D(P_1, P_2)$의 범위

현재 주어진 목적 집단에서 P_1과 가장 가까운 점이 P_2일 때, P_2를 같은 클러스터로 인정하지 않는다면 목적 집단에서 P_1이 포함될 클러스터는 존재하지 않기 때문이다. 따라서 P_1, P_2는 같은 클러스터로 인정할 수 있으며, 이 때 α를 3으로 정의하는 것이 좋다. α의 값이 3일 때, $D(P_1, P_2)$만큼 떨어져 있는 P_3도 같은 클러스터로 간주할 수 있기 때문이다. P_3는 P_1과 가장 먼 거리를 유지할 때가 P_1, P_2, P_3가 직선상에 위치할 경우이며, P_3가 같은 클러스터에 포함되는 것을 인정하므로 P_1, P_2가 이루는 거리에 의해서 타원이 형성되게 된다. 이 때 타원 안에 존재하는 모든 점 P_x를 모두 같은 클러스터로 인정한다.

또, 이렇게 얻은 P_x는 $D(P_1, P_x)$, $D(P_2, P_x)$의 거리를 얻어 확장할 수 있으며, 위의 수식을 만족하는 새로운 점을 클러스터에 포함시킬 수 있게 된다.

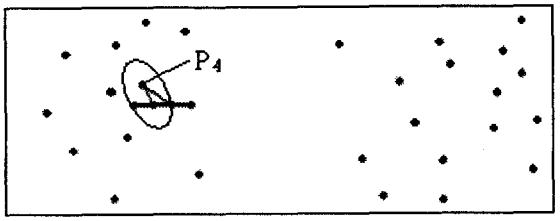


그림 3. 새로운 점 P_4를 통해서 생성된 새로운 클러스터 범위

만약, 특정 점 P_x가 기존 작성된 클러스터 C_1에 포함된다면, C_1은 새로이 작성된 클러스터에 병합되게 된다. 또한 그림 3에서 P_4를 사용하여 생성된 타원에는 새로운 점이 포함되어 있지 않으므로 확장이 종료된다.

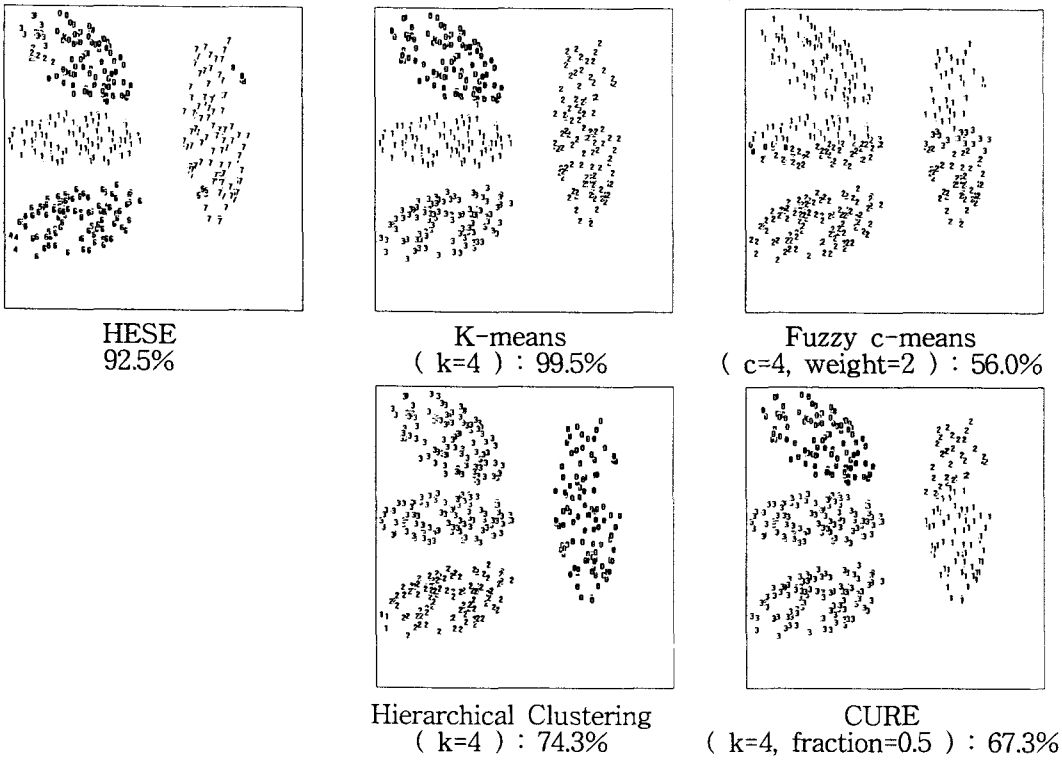
그러나 이렇게 확장을 하게 되면, $D(P_1, P_x)$가 계속해서 커지기 때문에 모든 점이 같은 클러스터로 인식되는 문제가 있다. 따라서 확장은 다음과 같은 제한사항을 가진다.

$$\text{if } (D(P_1, P_x) \leq P_D) \text{ then Expand}$$

위의 수식은 주어진 목적 집단을 이루는 점들의 평균거리를 사용하는데 이를 사용하여

최대한으로 확장할 수 있는 거리를 제한하고 타 클러스터를 침범하지 않는 클러스터 생성을

표 1. HESE에 의해 생성된 클러스터



있다. 주어진 목적 집단의 평균 거리는 각 점들이 서로 떨어져 있는 정도를 나타내므로 이보다 떨어져 있으면 있을수록 같은 클러스터일 가능성이 낮아지게 된다. 그러나 평균거리만으로는 주어진 목적 집단에서 상대적으로 가까운 거리에 있다고 하더라도 모든 점들이 평균 거리만큼 떨어져 있는 것이 아니므로 같은 클러스터로 묶일 가능성이 낮아지게 된다. 따라서 여기에 클러스터 확장을 위해 사용되는 계수 α를 사용하여 확장이 가능한 거리를 생성하여 최대한 확장이 가능하지만, 너무 큰 거리를 사용하여 클러스터를 생성하지 못하게 제한한다.

3. 실험 결과

HESE가 생성하는 클러스터들의 특성을 살펴보고, 위에서 정의한 '적합한 클러스터'를 생성하였는지 확인하기 위하여 표 1을 첨부하였다. 표 1은 K-means, Fuzzy c-means, Hierarchical Clustering 알고리즘에 의한 클러스터 생성과 HESE에 의한 클러스터 생성을 보여주고 있다. 알고리즘에 사용된 목적 집단은 임의로 선정하였으며, 위의 결과에서도 알 수 있듯이 기존의 알고리즘들이 특정 목적 집단에 대해서 타 클러스터를 침범하는 클러스터를 생성하는데 반해, HESE는 타 클러스터를 침범하지 않는 클러스터를 생성하고 있다.

위해 정확도가 낮아지는 결과를 초래하였으나 알고리즘의 기본 목적 달성에는 성공하였다.

표 1의 결과를 포함하여 임의의 목적 집단에 대해 HESE를 적용한 결과 도표는 아래와 같다.

표 2. HESE를 임의의 목적 집단에 적용한 결과

표본 / 알고리즘	ad02.txt	e2.txt	e5.txt	e6.txt	e7.txt
K-means	51.2%	99.5%	76.0%	95.8%	100.0%
Hierarchical	100.0%	74.3%	100.0%	66.8%	100.0%
FCM	53.8%	56.0%	50.0%	66.7%	66.7%
CURE	61.9%	67.3%	69.8%	100.0%	88.2%
HESE	100.0%	92.5%	86.0%	100.0%	100.0%

이 결과는 임의의 목적 집단에 대해서 완전한 정답을 찾지는 못함을 보여주고 있다. 이는 최초에 정의한 것과 마찬가지로 완전히 종료된 결과를 생성하는 것이 아니라 위에서 정의된 적합한 클러스터를 찾는 것이 HESE의 목적이기 때문이다. 그러므로 HESE를 통해 초기 클러스터를 생성하여, 합병을 위한 알고리즘을 함께 사용한다면 더욱 좋은 결과를 얻을 수 있을 것이다.

4. 결과 고찰

위에서 언급한 바와 같이 HESE는 완성된 클러스터 집단을 생성하진 못한다. 따라서 사용자가 완전히 분리가 종료된 클러스터 결과를 얻고자 한다면 HESE는 부적합하다. 그러나 특정 알고리즘을 통해 생성된 클러스터를 가지고 해당 클러스터의 특징을 추출해야 하는 사용자의 입장에서 다른 클러스터에 걸쳐서 분포된 클러스터보다는 다른 클러스터에 걸쳐져 있지 않지만 여러 개로 분리된 클러스터 집단을 얻는 것이 특징 추출에 도움이 될 것이다. 또한 HESE를 통하여 클러스터를 생성할 경우 사용자가 목적 집단에 대해서 미리 알고 있어야 할 정보가 없으므로 미지의 목적 집단에 대해 적용할 경우 HESE의 적합성이 높아질 것으로 보인다.

더욱이 HESE는 초기 집단을 생성하고 종료되므로 사용자가 클러스터링을 하고자 하는 목적 집단의 특성을 가미하여 병합 알고리즘을 별개로 사용할 경우 정확도 및 적합성이 본 논문에서 제시한 결과보다 훨씬 높아질 것이다.

참 고 문 헌

[1] D. Kim, K. Lee, D. Lee. "A novel initialization scheme for the fuzzy c-means algorithm for color clustering," Pattern Recognition Letters 25, pp. 227-237, 2004.

[2] J. Bezdec. "Pattern Recognition with Fuzzy Objective Function Algorithms," New York: Plenum, 1981.

[3] S. Guha, R. Rastogi, and K. Shim. "CURE: An efficient clustering algorithm for large databases," In Proceedings of ACM SIGMOD International Conference on Management of Data, pp. 73-84, New York, 1998.

[4] B. Everitt, S. Landau, M. Leese. "Cluster Analysis," London: Arnold, 2001.