

Clustering Algorithm Using a Center of Gravity for Grid-based Sample

Hee-Chang Park¹⁾ · Jee-Hyun Ryu²⁾

Abstract

Cluster analysis has been widely used in many applications, such as data analysis, pattern recognition, image processing, etc. But clustering requires many hours to get clusters that we want, because it is more primitive, explorative and we make many data an object of cluster analysis. In this paper we propose a new clustering method, 'Clustering algorithm using a center of gravity for grid-based sample'. It reduces running time by using grid-based sample and keeps accuracy by using representative point, a center of gravity.

Keywords : 그리드, 무게중심, 유클리드 제곱 거리, 최단연결법, 클러스터링

1. 서론

데이터 마이닝은 방대한 양의 데이터 내부에 내재된 의미 있는 정보인 상관관계, 패턴, 경향 등을 찾아내는 일련의 과정이다. 데이터마이닝의 기법에는 각종 통계적 기법뿐만 아니라 신경망, 의사 결정 나무, 연관성 규칙, 유전자 알고리즘, 메모리-기반 추론, 그리고 본 논문의 주제와 관련 있는 클러스터링 기법 등이 있다.

클러스터링은 아주 방대한 양의 데이터를 대상으로 하고 있으며, 다양한 특성을 지닌 관찰대상을 유사성을 바탕으로 동질적인 집단으로 분류하는 데 쓰이는 기법이다. 이러한 클러스터링 기법이 Tryon(1939)에 의해 처음 소개된 이후로 Tryon과 Bailey(1970), Jardine과 Sibson(1968), Anderberg(1973), Hartigan(1975), Jain과 Dubes 등(1988)에 의해 발전되었다.

기존의 클러스터링 기법을 알고리즘별로 분류하면 계층적인 방법, 분할적인 방법, 밀도 기반의 방법, 그리드 기반의 방법, 모델 기반 방법 등으로 나누어진다. 본 논문

1) First Author : Professor, Department of Statistics, Changwon National University, Changwon, 641-773, Korea
E-mail : hcpark@sarim.changwon.ac.kr
2) Graduate Student, Department of Statistics, Changwon National University, Changwon, 641-773, Korea

에서는 계층적인 클러스터링의 기법에서 그리드를 기반으로 한 클러스터링 기법을 제안하고자 한다. 계층적인 클러스터링 방법은 비슷한 성질의 데이터 개체들을 그룹화하여, 그 결과 생성된 군집을 트리 형태로 만들어 나가는 방법이다. 이에는 Kaufman과 Rousseeuw(1990)의 AGNES(AGglomerative NESTing)와 DIANA(DIvisive ANalysis), Zhang 등(1996)의 BIRCH(Balanced Iterative Reducing and Clustering Using Hierarchies), Guha 등(1998)의 CURE(Clustering Using REpresentative), Karypis 등(1999)의 Chameleon(A Hierarchical Clustering Algorithm Using Dynamic Modeling), 그리고 Guha 등(1999)의 데이터 개체의 범주형 속성에 대해서 클러스터링을 가능하게 한 ROCK(RObust Clustering using linKs) 기법 등이 있다.

본 논문에서는 정확성을 유지함과 동시에 수행속도를 높일 수 있는 방법으로 그리드 기반 표본의 무계중심을 이용한 클러스터링 기법을 제안하고자 한다. 2절에서 전형적인 클러스터링 방법에 대해서 알아보고, 3절에서는 그리드 기반의 표본의 무계중심을 전형적인 클러스터링 기법에 이용하는 방법에 대해서 알아보하고자 한다. 4절에서는 예제 및 모의실험을 통해 본 연구에서 제시한 기법과 전형적인 클러스터링 기법을 비교하여 수행속도와 정확도에서 만족할 만한 수준의 결과가 얻어짐을 확인하고자 한다. 마지막으로 5절에서 본 연구의 결론을 맺는다.

2. 클러스터링

클러스터링에서는 군집의 수 혹은 군집의 구조에 대한 가정이 없으며, 오직 데이터들 사이의 유사성에 의하여 군집을 형성하고, 형성된 군집의 특성을 파악하여 군집들 사이의 관계를 분석한다. 클러스터링을 수행하기 위해서는 먼저 데이터 개체에 대해 고려하고자 하는 변수를 관찰하여 자료행렬을 만든다. 그 다음에는 묶여지는 각 데이터간의 유사성의 정도를 측정하는 기준척도가 필요하다. 보통 이러한 정도를 측정할 때에는 유사성을 거리로 환산하여 거리가 가까운 대상들을 동일한 집단에 포함시키므로 어떠한 변수들을 설정할 것인가 하는 것이 먼저 해결되어야 한다. 클러스터링에서는 의미 없는 변수를 제거하는 과정이 없으므로 선택된 변수들이 모두 동일한 비중을 가진다. 두 개체 사이의 거리의 종류에는 유클리드 거리, 유클리드 제곱 거리, Mahalanobis 거리, 그리고 Minkowski 거리 등이 있다. 본 논문에서는 이들 중에서 식 (2.1)과 같이 정의되는 유클리드제곱 거리를 이용한다.

$$d_{ij}^2 = (X_i - X_j)'(X_i - X_j) \quad (2.1)$$

기준 척도를 정한 후에는 실제로 대상들에 대해서 군집화한다. 군집화 방법에는 크게 계층적 군집화 방법과 비계층적 군집화 방법이 있다. 계층적 군집화 방법에는 최단 연결법(Single Linkage Method), 최장 연결법(Complete Linkage Method), 평균 연결법(Average Linkage Method), 중심 연결법(Centroid Linkage Method), 중위수 연결법(Median Linkage Method), 그리고 Ward의 방법 등이 있다. 본 논문에서는 군집간의 최단 거리 중 가장 최소 거리를 가지는 군집끼리 병합하는 방법인 최단 연결법을 이용하였다. 이 방법에서의 최단 거리는 식 (2.2)와 같이 정의된다.

$$d(U, V) = \min [d(x, y) | x \in U, y \in V] \quad (2.2)$$

여기서 U 와 V 는 임의의 군집이며, x 와 y 는 해당 군집의 임의의 두 개체이다. 두 군집간의 거리를 각 군집에 속하는 임의의 두 개체들 사이의 거리 중 최단거리로 정의하여 가장 유사성이 큰 군집을 묶어 나간다. 최단 연결법은 Jardine과 Sibson(1968)이 제시한 바와 같이 수리적인 면이 우수하며, 컴퓨터 처리시간이 다른 방법들에 비해 빠르다는 장점을 가지고 있다.

3. 그리드 기반 표본의 무게중심을 이용한 클러스터링

3.1 그리드 기반 샘플링

전형적인 방법의 클러스터링은 아주 방대한 양의 데이터를 대상으로 하며, 클러스터링 자체가 원시적이며 탐색적으로 접근하기 때문에 만족할 만한 최종 클러스터들을 얻기까지 매우 많은 계산과정을 거치게 되며, 그만큼 많은 시간을 요구하게 된다. Park과 Ryu(2002)는 클러스터링의 수행시간을 최소화하기 위해 샘플링 이전에 그리드를 사용하여 데이터 개체들을 그리드 간격으로 분할하였다. 이러한 그리드 기반 샘플에 의한 클러스터링에서는 그리드 간격이 넓으면 넓을수록 계산 과정이 줄어들며 그만큼 시간이 단축되지만 정확도는 떨어질 것이고, 그리드 간격이 좁아지면 계산 과정이 늘어나서 시간은 늘어나지만 정확도는 더욱 향상된다. 따라서 정확도 대비 속도의 적절한 균형점을 찾아야 할 것이며, 이는 곧 그리드 간격을 어떻게 설정하느냐 하는 문제로 귀착된다. 그들이 제시한 알고리즘의 그리드 간격을 GI (Grid Interval)라고 할 때, GI 를 다음과 같이 설정하였다.

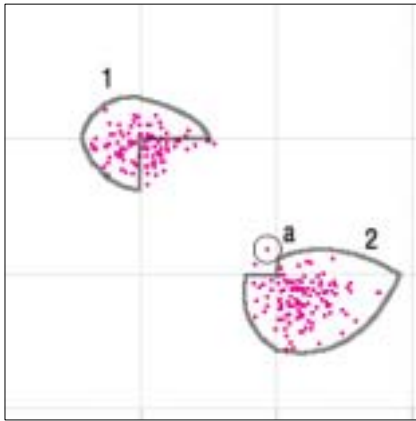
$$GI_v = \frac{\max_v - \min_v}{n^{\frac{1}{p}}} \quad (3.1)$$

여기서 v 는 각 변수를 나타내며, \max 와 \min 은 각각 해당 변수의 최대값과 최소값을 나타낸다. p 는 차원수 즉, 변수의 수이다. n 은 결측치가 없다고 가정할 때, 각 변수들의 데이터들이 이루는 쌍의 개수가 되며, 그 쌍은 곧 거리를 위한 좌표점으로 나타낸다.

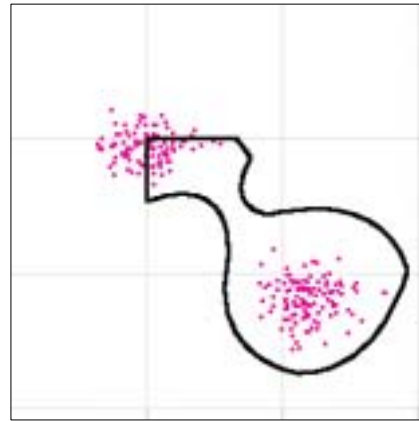
3.2 그리드 기반 표본의 무게중심

Park과 Ryu(2003)가 구현한 그리드 기반 표본의 클러스터링은 기존의 전형적인 클러스터링에 비해서 수행 시간이 훨씬 단축된다고는 할 수 있으나 클러스터링 결과가 정확도 측면에서 기존 클러스터링 기법의 수준을 유지할 수 있다고는 보장할 수 없다. 이는 적절한 그리드 간격의 설정과 샘플링 부분에서 얼마만큼 원본 데이터를 잘 대표하느냐의 문제이다.

다음의 <그림 1>과 <그림 2>에서 보는 바와 같이 그리드 간격이 넓거나 셀에 포함된 데이터 중에서 이상치가 포함되어 있다면 한 점의 샘플링은 그 셀을 잘 대표하지 못해서 클러스터링을 잘못 수행하는 결과를 보일 수도 있다. <그림 1>의 점 a는 그 점이 속한 셀에서는 이상치로 볼 수도 있다. 만약 가운데 셀에서 점 a가 샘플링 되었다면 <그림 2>에서 보는 바와 같이 그 점은 해당 셀의 대표점이 되어 클러스터링 수행시 그 셀의 모든 데이터가 오른쪽 아래의 데이터들(<그림 1>의 2)과 병합되는 잘못된 결과를 낼 것이다. 만약에 점 a가 샘플링 되지 않고 왼쪽 위의 데이터들(<그림 1>의 1)과 가까운 점이 샘플링 된다면 점 a와 그 왼쪽 아래의 점은 <그림 1>의 2-데이터들과 병합되지 않고 <그림 1>의 1-데이터들과 병합될 것이다. 따라서 이를 해결하기 위해서는 그리드 간격을 줄여야 할 것이다. 그러나 그리드 간격을 줄이는 만큼 정확도는 높아지지만 수행시간이 늘어나기 때문에 간격을 줄이는 것만으로는 적절한 대안이 될 수는 없다.



<그림 1> 한 점의 샘플링



<그림 2> 군집 결과

본 연구에서는 데이터의 차원별 세 점을 추출하여 얻어진 대표점으로 클러스터링을 수행하는 방법을 제안한다. 임의의 한 점으로는 각 셀을 충분히 대표하기에 어려운 점이 있으므로 최소의 샘플링 횟수로 해당 셀을 잘 대표할 수 있는 방법을 제시하고자 한다.

각 셀 단위로 비복원 추출로 랜덤 샘플링을 하며 샘플은 최대 3개까지 뽑는다. 샘플링된 점들로부터 구한 삼각형의 무게 중심을 *COG*(Center Of Gravity)라고 했을 때, *COG*는 다음과 같다.

$$COG = \left(\frac{x_{11} + x_{12} + x_{13}}{3}, \frac{x_{21} + x_{22} + x_{23}}{3}, \dots, \frac{x_{p1} + x_{p2} + x_{p3}}{3} \right) \quad (3.2)$$

여기서 p 는 차원의 수이며, $x_{11}, x_{12}, x_{13}, \dots, x_{p1}, x_{p2}, x_{p3}$ 는 샘플링 된 데이터들

이다. 위에서 얻어진 무게 중심을 해당 그리드의 대표점으로 지정하며, 그리드에 포함된 점의 수가 하나인 경우 그 점 자체가 대표점이 된다. 그리드에 포함된 점의 수가 두개인 경우는 그 두 점의 중심을 대표점으로 지정한다. 대표점으로는 반드시 세 점의 무게 중심이 될 필요는 없지만, 정확도 대비 속도의 적절한 균형점으로 세 점에 의한 삼각형의 무게 중심이 대표점으로 더욱 적절하다. 만약에 속도보다도 정확성을 더욱 중요시한다면 이 부분에서 샘플링의 수를 적절히 늘리면 될 것이다.

이렇게 각 그리드별로 얻어진 대표점들로 클러스터링을 수행하며, 마지막에 얻어진 클러스터들에 대해서 대표점이 속해 있는 셀 내의 원 데이터들을 해당 클러스터에 포함시키는 라벨링 작업을 수행한다.

3.3 알고리즘의 구현

기존의 전형적인 클러스터링 기법은 클러스터링 계산 과정에 있어서 데이터 또는 데이터 개체들의 수에 의존하는 반면에 그리드 기반 표본의 클러스터링은 셀의 수에 의존하므로 기존의 방법에 비해서 빠른 처리 시간을 가진다. 본 연구에서 제시하는 그리드 기반 표본의 무게중심을 이용한 클러스터링 수행 단계는 다음과 같다. 수행 단계는 Park과 Ryu가 제시한 그리드 기반 표본의 클러스터링 수행 단계에서 세 점의 랜덤 샘플링 단계와 무게중심을 이용한 대표점 계산 단계가 수정 및 추가되었다.

[단계 1] 데이터 처리 : 클러스터링을 수행할 데이터를 얻어 새로운 데이터 셋을 만든다. 데이터수가 작다면 직접 입력을 해도 되겠으나, 일반적으로 클러스터링에서는 대량의 데이터를 다루므로 데이터베이스(Database) 또는 파일 시스템으로부터 데이터를 얻어온다. 일단 데이터가 얻어지면 개체 번호, 각 점들로 이루어진 개체들을 선연하며 이 개체들로 데이터 셋을 다시 구성한다.

[단계 2] 그리드 설정 : 새로 구성된 데이터 셋으로부터 각 차원별 그리드 간격을 얻은 후, 데이터 셋을 그리드 간격에 맞게 분할한다.

[단계 3] 세 점의 랜덤 샘플링 : 이 단계에서는 그리드별 각 셀에서 무게 중심을 구하기 위한 점들로서 최대 세 점까지 비복원으로 랜덤 샘플링을 수행한다.

[단계 4] 무게중심을 이용한 대표점 계산 : 샘플링 된 데이터로부터 무게중심을 구하며, 구해진 무게중심으로 이루어진 대표점 셋을 만든다.

[단계 5] 클러스터링 : 다음 단계에서는 얻어진 대표점으로 클러스터링을 수행하며, 본 연구에서는 유클리디안 제곱거리와 최단연결법을 이용하였다.

[단계 6] 라벨링 : 샘플링 되지 않은 원 데이터를 해당 클러스터로 합병과 동시에 라벨링을 수행하며 최종 결과를 출력한다.

4. 예제 및 모의 실험

본 절에서는 3절에서 구현한 알고리즘을 바탕으로 수행 시간 및 정확도를 비교하기 위하여 예제 및 모의 실험을 실시하였다. 본 실험의 구현환경은 다음과 같다.

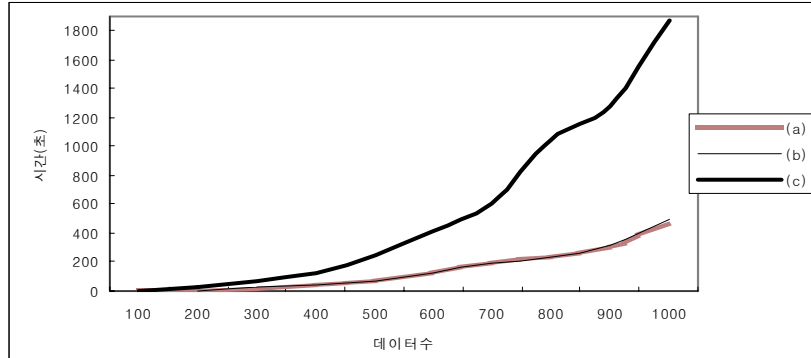
CPU : Intel Pentium4-1.8GHz Northwood
 RAM : 512MB
 O/S : Microsoft Windows XP Professional
 Language : JAVA J2SDK 1.4.0
 Database : MySQL 3.23.51 (External Linux Server)

첫 번째 실험은 클러스터링의 방법별 수행 시간을 비교하기 위해서 데이터의 특정 분포에 결과가 치우치지 않도록 데이터를 랜덤 발생시켜 고루 분포되도록 한 후 실험을 실시하였다. 데이터는 0.00부터 1000.00까지 값을 가지는 두 변수를 이용하였으며, 각 데이터의 개수별 수행시간 비교 결과는 <표 1>과 같다.

<표 1> 클러스터링 방법별 수행 시간 비교

데이터수	(a)	(b)	(c)
100	0.90	0.93	2.26
200	5.49	5.50	22.90
300	19.48	20.81	62.27
400	42.15	43.10	124.74
500	68.76	69.34	249.89
600	120.62	124.36	417.01
700	192.24	199.45	603.06
800	235.81	238.04	1043.69
900	308.58	310.31	1275.71
1000	467.00	497.00	1878.88

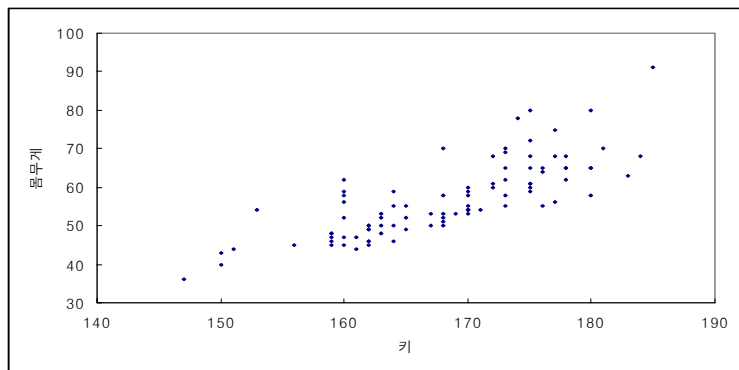
여기서 (a)는 그리드 기반 표본의 클러스터링 알고리즘, (b)는 그리드 기반 표본의 무게중심을 이용한 클러스터링 알고리즘, 그리고 (c)는 전형적인 클러스터링 알고리즘이다. 위의 결과에서, 그리드 기반 표본의 클러스터링 기법과 그리드 기반 표본의 무게중심을 이용한 클러스터링 기법은 전형적인 클러스터링 기법에 비해서 수행 시간이 훨씬 단축되었다는 것을 알 수 있으며 <표 1>에 대한 그래프는 다음의 <그림 3>과 같다.



<그림 3> 수행 시간 비교 그래프

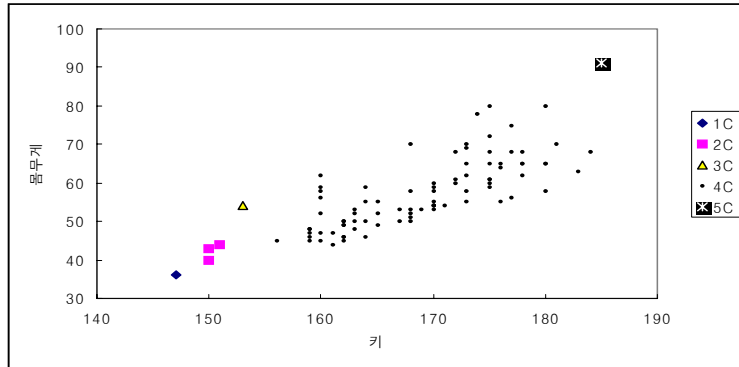
전형적인 클러스터링 기법은 <그림 3>에서 보는 바와 같이 데이터수가 1000개인 경우 수행 시간이 약 1800초로서, 실제로 방대한 양의 데이터로 클러스터링을 수행한다면 클러스터링 수행 자체가 거의 불가능하다고 할 정도로 수행시간이 급격하게 증가할 것이다. 그리고 그리드 기반 표본의 무게중심을 이용한 클러스터링 기법은 무게중심을 이용하지 않은 그리드 기반 표본의 클러스터링 기법에 비해서 약간의 수행 시간 차이가 있지만 전형적인 클러스터링 기법의 수행 시간에 비할 바는 아니다.

두 번째 실험은 클러스터링의 방법별 정확도를 비교하기 위해서 창원시에 소재하고 있는 모 중학교의 학생 키와 몸무게의 실제 데이터 ($n=92$)를 이용하였다. 다음은 데이터의 전체 분포를 나타낸 산점도이다.



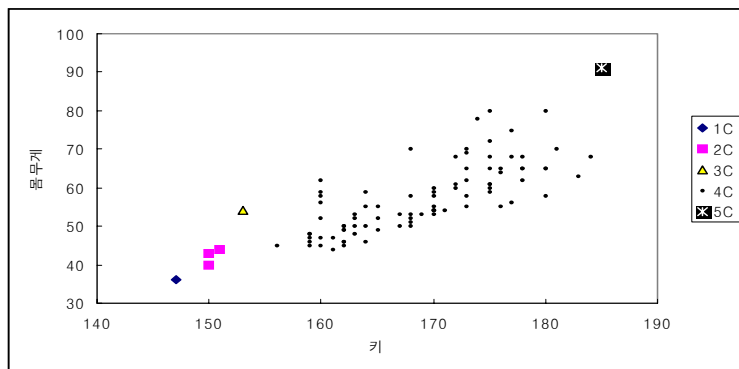
<그림 4> 전체 데이터 산포도

전형적인 클러스터링 기법에 의한 군집 결과는 다음과 같으며 5개의 군집으로 나눈 경우이다.



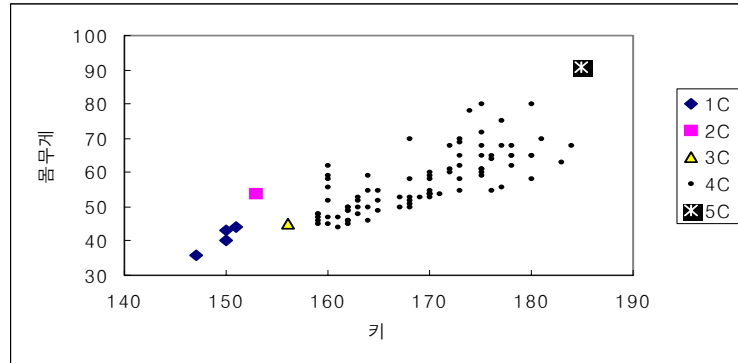
<그림 5> 전형적인 클러스터링 결과

다음의 두 그래프 <그림 6>과 <그림 7>은 그리드 기반 표본의 클러스터링 기법과 그리드 기반 표본의 무게중심을 이용한 클러스터링 기법의 군집 결과이며 5개의 군집으로 나눈 경우이다.



<그림 7> 그리드 기반 표본의 무게중심을 이용한 클러스터링 결과

실험에 의하면, 그리드 기반 표본의 무게중심을 이용한 클러스터링 기법은 전형적인 클러스터링 기법을 기준으로 군집이 5개 일 때 100%, 군집이 10개 일 때 97.752%의 정확도를 보였으며, 무게중심을 이용하지 않은 그리드 기반 표본의 클러스터링 기법은 군집이 5개 일 때 97.8%, 군집이 10개 일 때 96.6%의 정확도를 나타내었다. 무게중심을 이용하지 않은 클러스터링 기법에서는 <그림 8>에서 보는 바와 같이 키가 160 이하에서 군집이 잘못 나누어졌음을 알 수 있다.



<그림 8> 그리드 기반 표본의 클러스터링 결과

5. 결론

클러스터링은 방대한 데이터를 다루며 군집의 개수, 분포에 대해 아무런 가정을 두지도 않는다. 또한, 원시적이며 탐색적인 방법으로서 수행 시간이 길어진다는 단점에도 불구하고 여러 분야에서 데이터 마이닝의 한 기법으로 많이 쓰이고 있는 기법이다. 기존의 전형적인 클러스터링 기법은 클러스터링 계산 과정에 있어서 데이터 또는 데이터 개체들의 수에 의존하는 반면에 그리드 기반 표본의 클러스터링은 셀의 수에 의존하므로 기존의 방법에 비해서 빠른 처리 시간을 가진다.

본 논문에서는 그리드 기반으로 표본을 추출한 후, 그리드 별 각 셀 단위로 무게중심을 이용하여 클러스터링을 수행하였다. 실험 결과 전형적인 클러스터링 기법에 비해 그리드 기반 표본의 무게중심을 이용한 클러스터링 기법은 수행 시간을 단축시키는 동시에 정확도에 있어서도 만족할 만한 수준의 결과를 나타낸다는 것을 확인하였다. 또한 그리드 기반 표본의 클러스터링 기법과 비교해볼 때 수행속도 면에서는 별 차이가 없으나 더 정확한 클러스터링 결과를 얻었다.

참고문헌

1. Anderberg, M.R. (1973). *Cluster Analysis for Applications*, Academic Press.
2. Guha, S., Rastogi, R., and Shim, K. (1998). CURE: An efficient clustering algorithm for large databases, *ACM-SIGMOD Int. Conf. Management of Data(SIGMOD'98)*, 73-84, Seattle, WA.
3. Guha, S., Rastogi, R., and Shim, K. (1999). Rock: A robust clustering algorithm for categorical attributes, *In Proc. 1999 Int. Conf. Data Engineering (ICDE'99)*, 512-521, Sydney, Australia.
4. Hartigan, J.A. (1975). *Clustering Algorithms*, John Wiley & Sons.

5. Jain, A.K. and Dubes, R.C. (1988). *Algorithms for Clustering Data*, Englewood Cliffs, NJ, Prentice Hall.
6. Jardine, C.J. and Sibson, R. (1968). *The construction of hierarchic and nonhierarchic classifications*, *Comput. J.*, 11, 177-184.
7. Karypis, G., Han, E.H., and Kumar, V. (1999). Chameleon: Hierarchical Clustering using Dynamic Modeling, *IEEE Computer*, 32, 68-75.
8. Kaufman, L. and Rousseeuw, P.J. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*, John Wiley & Sons.
9. Park, H.C. and Ryu, J.H. (2003). Clustering Algorithm by Grid-based Sampling, *Journal of the Korean Data & Information Science Society*, 14(3), 535-543.
10. Tryon, R.C. (1939). *Cluster Analysis*, McGraw-Hill.
11. Tryon, R.C. and Bailey, D.E. (1970). *Cluster Analysis*, McGraw-Hill.
12. Zhang, T., Ramakrishnan, R., and Livny, M. (1996). BIRCH: An efficient data clustering method for very large databases, *ACM-DIGMOD Int. Conf. Management of Data (SIGMOD'96)*, 103-114, Montreal, Canada.

[2005년 1월 접수, 2005년 4월 채택]