Proceedings of Joint Conference of The Korean Data Analysis Society and Korean Data And Information Science Society, October 17-18, 2003, pp. 249-258

K-means Clustering using a Grid-based Sampling

Hee Chang Park[1], *Sun Myung Lee*[2]

Abstract

K-means clustering has been widely used in many applications, such that pattern analysis or recognition, data analysis, image processing, market research and so on. It can identify dense and sparse regions among data attributes or object attributes. But k-means algorithm requires many hours to get k clusters that we want, because it is more primitive, explorative. In this paper we propose a new method of k-means clustering using the grid-based sample. It is more fast than any traditional clustering method and maintains its accuracy.

Keywords : data mining, k-means clustering, grid-based sampling

1. 서론

현대사회가 복잡해지고 다양해짐에 따라 우리는 엄청난 양의 정보를 접하면서 살아가고 있다. 이러한 정보는 컴퓨터의 발전과 더불어 데이터베이스로 구축되어진다. 데이터 마이닝(data mining)은 이러한 방대하고 다양한 형태의 데이터로부터 의사결정에 유용한 정보 및 지식을 발견하려는 일련의 데이터 분석 및 모형선정의 과정이다. 정보를 추출하기 위한 데이터 마이닝의 기법에는 의사결정나무, 연관성 규칙, 클러스터링, 그리고 신경망 분석 등이 있다. 이들 중에서 클러스터링은 다양한 특성을 지닌 관찰대상을 유사성을 바탕으로 동질적인 집단으로 분류하는데 쓰이는 기법으로 개체들 사이의 유사성 또는 거리에 의하여 군집을 형성하고, 형성된 군집의 특성을 파악하여 군집들 사이의 관계를 분석하는 기법이다. 클러스터링에는 분할 군집법, 계층적 군집법이 있다. 그 중에서 분할 군집법은 데이터들을 임의의 부분집합으로 분할을 한 후 데이터들을 유사한 그룹으로 재배치하는 군집방법이다. 분할 군집법의 종류에는 본 논문에서 연구하고자 하는 k-means 알고리즘과 k-medoids 알고리즘, k-prototypes 알고리즘, k-modes 알고리즘 등이 있다.

k-means 알고리즘은 MacQueen(1967)에 의해 처음 소개되었으며, 데이터들을 k개의 군집으로 임

[1]Professor, Department of Statistics, Changwon National University, Changwon, Kyungnam, 641-773, Korea. E-mail : hcpark@sarim.changwon.ac.kr

[2]Graduate Student Department of Statistics, Changwon National University, Changwon, Kyungnam, 641-773, Korea

의로 분할을 하여 군집의 무게중심(평균)을 대표값으로 분할해 나가는 방법으로 데이터들을 유사성을 바탕으로 재배치를 하는 방법이다. k-means는 군집의 재배치를 통해서 계층적 군집법의 잘못 분류된 데이터를 되돌릴 수 없는 단점을 극복할 수 있는 분할 군집법의 장점을 가지고 있다. Kaufman과 Rousseeuw(1990)는 k-means 알고리즘이 이상값에 민감한 것을 보완하여 군집의 대표값을 중앙값으로 하는 k-medoids 방법인 PAM(partitioning around medoids) 알고리즘을 제안하였다. PAM은 적은 데이터 셋에서는 좋은 결과를 보였으나 많은 양의 데이터 셋에서는 효과적이지 못하다. 그래서 그들은 많은 양의 데이터를 취급하기 위해 CLARA(clustering large applications) 알고리즘을 제안하였다. 이 알고리즘은 데이터를 샘플링하여 PAM을 적용한 방법으로, 표본을 잘 뽑았다면 표본의 중앙값은 전체 데이터의 중앙값에 근사하며, 더 나은 근사값을 위해 CLARA는 다중 샘플을 사용한다. Ng 등(1995)은 CLARA를 더욱 향상시킨 CLARANS(clustering large applications based on randomized search)를 제안하였다. CLARA 알고리즘이 조사의 각 단계에서 고정된 표본을 가지는 반면에 CLARANS는 조사의 각 단계에서 어떤 임의 표본을 가지며, 이상점을 발견할 수도 있다. Huang(1997, 1998)은 k-means가 연속형 데이터에 대해 한정된 단점을 보완한 연속형과 범주형의 혼합된 데이터에 대한 k-prototypes 알고리즘을 제안하는 동시에, 범주형 데이터에 대해서 k-modes 알고리즘을 제시하였다. Chu 등(2002)은 k-medoids가 이상점에 강한 반면 수행속도가 느리다는 약점을 극복하기 위해 효과적인 샘플링을 기법을 추가하여 MCMRS(Multi-Centroid, Multi-Run Sampling Scheme) 알고리즘을 제시하였으며, 또한 이들은 MCMRS의 발전된 더 진보된 샘플링 기법인 IMCMRS(Incremental Multi-Centroid, Multi-Run Sampling Scheme) 알고리즘을 제안하였다.

앞에서도 언급한 바와 같이 데이터 마이닝은 대량의 데이터를 대상으로 하므로 그 만큼 처리 속도에 대한 발전된 많은 방법이 연구되고 있다. 특히 웹이 보편화된 현재 사용자들의 다양한 패턴을 분석하기 위한 데이터 마이닝 방법이 사용되어지고 있는데 처리 속도 문제는 더욱 중요하게 생각하고 있다. 그 이유는 아무리 정확하고 유용한 정보라 할지라도 과거의 정보는 흘러간 정보이기 때문이다. 이러한 속도 문제를 해결하기 위해 본 논문에서는 분할 군집법에서 가장 일반적으로 사용되고 있는 k-means 알고리즘에 대해 그리드를 기반으로 한 샘플링 알고리즘을 제안하고자 한다. 2절에서는 k-means에 대한 일반적인 방법을 살펴본 후, 3절에서는 그리드 기반 표본을 이용한 k-means 알고리즘을 구현하며, 4절에서는 예제 및 모의실험을 통하여 본 연구에서 제시한 기법과 기존의 기법을 비교하여 수행속도와 정확도에서 만족할 만한 수준의 결과가 얻어짐을 확인하고자 한다. 마지막으로 5절에서 본 연구의 결론을 맺고자 한다.

2. k-means 군집방법

k-means 군집방법은 데이터들을 k개의 군집으로 임의로 분할을 하여 군집의 무게중심(평균)을 대표값으로 분할해 나가는 방법으로 데이터들을 유사성을 바탕으로 재배치를 하는 방법이다. 군집분석에서 군집간의 유사성 측정은 거리로써 나타낸다. 서로 다른 개체 사이의 거리

$d_{ij} = d(X_i, X_j)$를 구하는 방법에는 유클리디안(Euclidean) 거리, 유클리디안 제곱거리, 마할라노비스(Mahalanobis) 거리, 그리고 민코우스키(Minkowski) 거리 등이 있으며, 본 논문에서는 식(2.1)의 유클리디안 제곱거리를 이용하고자 한다.

$$d_{ij} = \sum_{k=1}^{p} (t_{ik} - t_{jk})^2 \tag{2.1}$$

여기서 $d_{ij} \geq 0$, $d_{ii} = 0$, $d_{ij} = d_{ji}$ $d_{ik} + d_{jk} \geq d_{ij}$ 이다.

기본적인 k-means 군집방법의 수행단계는 다음과 같다.

[단계 1] 데이터들을 임의의 k 개의 군집으로 분할한다.

[단계 2] 각 군집의 평균을 구한다.

[단계 3] 각 데이터 점들과 각 군집의 평균과의 거리를 구하여 데이터 점들을 가장 가까운 군집으로 재할당한다.

[단계 4] 데이터 점들의 재배치가 없을 때까지 [단계 2] 및 [단계 3]의 과정을 반복한다.

3. 그리드 기반 표본의 k-means 군집방법

데이터 크기와 변수 수에 따라 처리 수행 시간에는 상당한 차이가 있다. 유용한 정보를 빠른 시간 내에 얻어야 하는 경우 속도는 큰 문제점이 된다. 이러한 속도 문제를 해결하기 위해 샘플링 기법을 도입할 수 있다. 일반적으로 이용되고 있는 샘플링 방법에는 단순임의추출법, 층화임의추출법, 집락추출법, 계통추출법 등이 있으며, 이들은 모집단을 잘 대표하는 샘플링 방법들이다. 하지만 k-means 군집분석에 적용하기에는 문제점들이 있다. 단순임의 추출법, 집락추출법, 계통추출법을 이용하는 경우 샘플링한 데이터로 k-means를 수행하지만 샘플링 되지 않은 나머지 데이터에 대한 클러스터링은 불가능하다. 그리고 층화임의추출법은 샘플링된 데이터로 k-means를 수행한 후 샘플링 되지 않은 나머지 데이터를 각층을 기준으로 각 군집에 배치를 시킬 수 있다. 하지만 층화임의추출법은 중요한 특정 변수에 의해 층을 나누므로 변수에 영향을 받는 점이 있다. 또한 변수의 중요도를 고려하지 않는 클러스터링에서의 개념과 다르므로 사용하기 어렵다.

따라서 본 논문에서는 그리드를 기반으로 한 샘플링 방법을 고려하고자 한다. 그리드 기반 샘플링은 데이터를 일정한 간격의 그리드로 나눈 후 각 그리드별로 데이터를 한 점씩 샘플링 하여 k-means를 수행하는 방법이다. 여기서 그리드는 유사한 성질의 데이터 집합이다. 그리드 기반 샘플링은 기존의 방법에 비해 속도를 크게 향상시킬 수 있다. 기존의 방법들은 데이터 수에 영향을 받지만 그리드 기반 샘플링은 그리드 수에 영향을 받으므로 데이터 수가 현저히 줄어들어 속도를 향상시킨다.

3.1 그리드의 설정

클러스터링의 수행과정을 최소화하기 위해 샘플링 이전에 그리드를 사용하여 데이터 개체들을 적당한 그리드 간격으로 분할한다. 그리드 간격이 넓으면 넓을수록 계산 과정이 줄어들며 그만큼 시간이 단축되지만 정확도는 떨어질 것이고, 그리드 간격이 좁아지면 계산 과정이 늘어나서 시간은 늘어나지만 정확도는 더욱 향상될 것이다. 따라서, 정확도 대비 속도의 적절한 균형점을 찾아야 할 것이며, 이는 곧 그리드 간격을 어떻게 설정하느냐 하는 문제로 귀착된다. 본 논문에서 제시하는 알고리즘의 그리드 간격을 GI(Grid Interval)라고 할 때 GI는 다음과 같이 설정한다.

$$GI_v = \frac{\max_v - \min_v}{n^{\frac{1}{p}}} \tag{3.1}$$

여기서 v는 v번째 변수를 나타내며, $\max_v$와 $\min_v$는 각각 v번째 변수의 최대값과 최소값을 나타낸다. n은 결측치가 없다고 가정할 때, 각 변수의 데이터가 이루는 쌍의 수가 되며, 그 쌍은 곧 거리를 위한 좌표점으로 나타낸다. p는 차원수 즉, 변수의 개수이다. 데이터 공간이 2차원인 경우, 데이터들의 분포가 정사각형으로 고루 분포되었다고 가정할 때 그 넓이는 n이고 한 변의 길이는 $\sqrt{n}$이 된다.

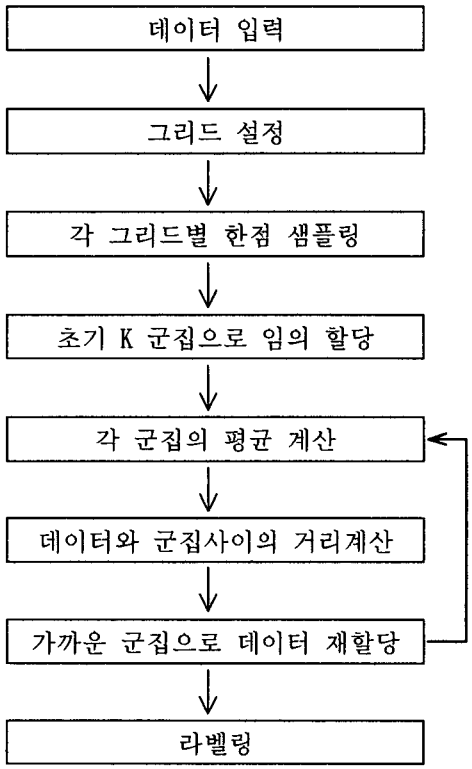
그리드 간격은 데이터들이 고루 분포되었을 경우 유사한 값을 가지는 개체들은 그리드별로 동일한 셀에 포함되도록 하였다. 먼저 각 좌표축별로 그리드 간격이 설정되면 원 데이터를 해당 그리드로 분할하며, 이는 곧 그리드의 각 셀을 하나의 군집으로 보는 첫 번째 클러스터링이 수행되는 시점이라고 볼 수 있다. 데이터가 정사각형 모양으로 고루 분포되었을 경우 그리드별 각 셀에는 한 점만을 포함할 것이며, 이 경우 클러스터의 수는 n이 될 것이다. 하지만, 셀 별로 각 한 점만을 포함할 경우는 거의 없으므로 최소한 최초 클러스터의 수는 n보다는 작게 될 것이며, 한 점도 포함하지 않는 셀은 계산에서 제외된다.

3.2 그리드 기반 샘플링

기존의 전형적인 k-means 기법은 클러스터링 계산 과정에 있어서 데이터 또는 데이터 개체들의 수에 의존하는 반면에 그리드 기반 표본의 k-means 기법은 셀의 수에 의존하므로 기존의 방법에 비해서 빠른 처리 시간을 가진다. 본 논문에서는 데이터를 동일한 간격의 그리드로 나눈 후 각 그리드별로 단순임의추출을 한다. 각 그리드별로 한점을 샘플링 하여 샘플링 된 데이터로 클러스터링을 수행한다.

3.3 알고리즘 구현

그리드 기반 표본의 k-means 군집분석을 위한 수행 과정은 다음과 같다.



<그림 1> 그리드 기반 표본의 k-means 군집 과정

이에 대한 알고리즘은 다음과 같이 구현된다.

```
clustering()
{
  int k, n, p;
  float Data;
  Data = InsertData();

  GI_v = (Max − Min) / n^(1/p);

  while(Grid_X)
  {
    while(Grid_Y)
    {
      while(i <= n)
      {
        if((GridX_min < x[i] <= GridX_max) &&
              (GridY_min < y[i] <= GridY_max))
        {
          DataGrid[i] = GridNo;
        }
        i++;
      }
    }
  }

  while(G_No <= DataGrid)
  {
    Sampling[G_No] = Rand();
    G_No++;
  }

  k_means();

  while(Grid)
  {
    Nearst = 0;
    while(remainData)
    {
      Dist = (GridCenter − Data)^2
      if(Dist < Nearst)
      {
        Nearst = Dist;
        DataGird[i] = G_No;
      }
    }
  }
}
```

4. 예제 및 모의실험

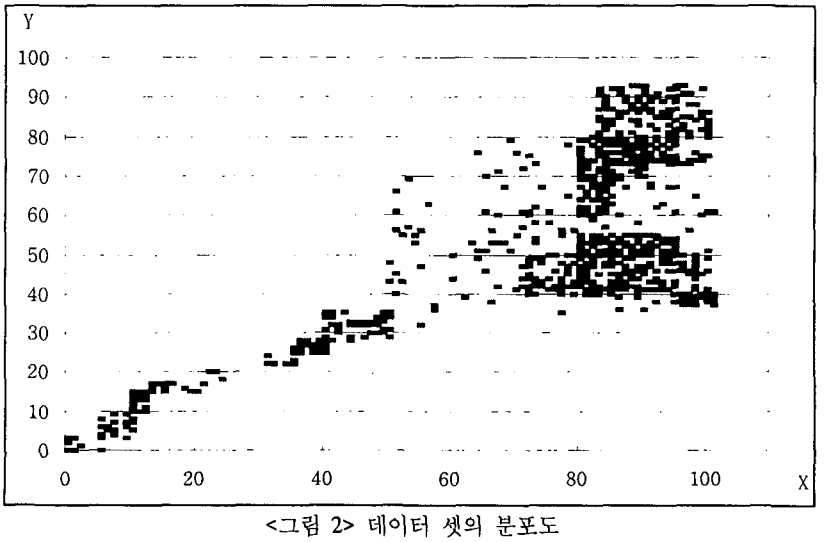
앞 절에서 구현한 그리드 기반 표본에 의한 k-means 알고리즘을 바탕으로 수행 시간 및 정확도를 비교하기 위하여 모의실험을 실시하였다. 본 실험의 구현환경은 다음과 같다.

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)

모의실험을 위해 두 개의 변수로 이루어진 1000건의 데이터를 랜덤하게 발생시켜 기본 데이터 셋으로 사용하였다. 실험은 기본 데이터 셋에서 랜덤 샘플링하여 사용을 하였다. 이들 데이터 셋에 대한 특징은 다음과 같다.

데이터 수 : 1000건
변수값의 범위 : X(0 ~ 101), Y(0 ~ 93)
$\overline{X}$ = 75.7, S_x = 24.9
$\overline{Y}$ = 53.9, S_y = 22.0

데이터 셋의 분포는 <그림 2>와 같다.



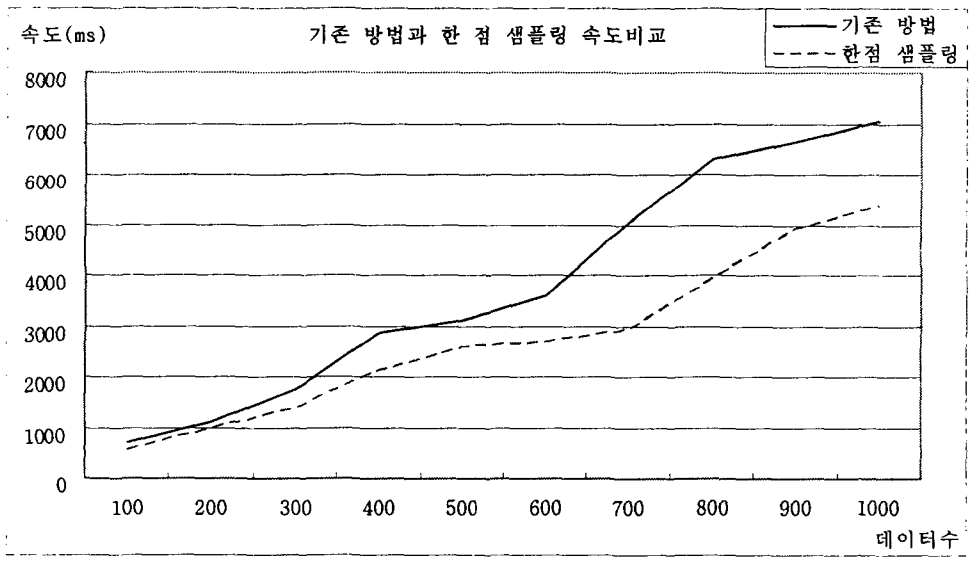
<그림 2> 데이터 셋의 분포도

그리드를 기반으로 한 점을 샘플링하여 샘플링 된 데이터로 k-means를 수행하였을 때 수행 속도와 정확도는 <표 1>과 같다.

<표 1> 그리드 기반 표본에 의한 방법의 수행 속도와 정확도

데이터 수	속도(ms)	정확도
100	541	99%
200	971	97.5%
300	1372	96.7%
400	2113	98.5%
500	2584	98.8%
600	2663	99%
700	3465	99%
800	3935	98.6%
900	4908	98.1%
1000	5348	98.2%

기존의 방법과 그리드 기반 표본에 의한 방법의 수행 속도를 비교한 결과는 <그림 3>과 같다.



<그림 3> 기존 방법과 그리드 기반 표본에 의한 방법의 수행 속도

위의 결과에서 보는 바와 같이 기존의 방법에 비해 한 점 샘플링이 속도에 있어서 효과적인 것을 알 수 있다. 반면에 정확도에 있어서는 <표 1>에서와 보는 것과 같이 다소 떨어지는 것을 알 수 있다. 이러한 실험 결과로 볼 때 수행 속도와 정확도의 적절한 균형점을 찾아 효율적인 방법을 찾는 것이 좋은 방향이 되는 것을 알 수가 있다.

위의 모의 실험에서 검증된 결과를 바탕으로 실제 데이터에 적용을 시킬 수가 있다. 창원 모 중학교 학생들의 신체검사 결과인 실제 데이터를 사용하여 속도를 비교해 보았다. 데이터의 특징은 다음과 같다.

데이터 수 : 1,734 건
사용 변수 : 키(X), 몸무게(Y)
$\overline{X}$ = 159.6, S_x = 8.4
$\overline{Y}$ = 50.9, S_y = 10.8

실험한 결과, 기존의 방식은 27,720ms, 한 점 샘플링은 10,185ms로 나타났다. 이 예제에서도 기존의 방법에 비해 그리드 기반 표본에 의한 방법이 속도 면에서 뛰어난 것을 알 수 있다.

5. 결론 및 향후 과제

아무리 정확하고 유용한 정보라 할지라도 과거의 정보는 흘러간 정보이기 때문에 데이터 마이닝에서 처리 속도 문제는 해결해야 할 과제 중의 하나이다. 이러한 속도 문제를 해결하기 위해 본 논문에서는 분할 군집법에서 가장 일반적으로 사용되고 있는 k-means 알고리즘에 대해 그리드를 기반으로 한 샘플링 알고리즘을 제안하였다. 동시에 본 연구에서 제시한 기법과 기존의 기법을 모의실험 및 예제를 통하여 비교하였으며, 수행속도와 정확도에서 만족할 만한 수준의 결과가 얻어짐을 확인하였다.

본 논문에서는 그리드를 동일한 간격으로 하였지만 정확도를 보다 높이기 위해 데이터의 밀집도에 따라 그리드 간격을 달리하는 문제가 향후 과제로 고려되어야 할 것이다.

참고문헌

1. MacQueen, J. (1967). "Some methods for classification and analysis of multivariate observations." in *5th Berkeley Symp. Math. statist, Prob.* 1, 281-297.

2. Kaufman, L. and Rousseeuw, P.J. (1990), *Finding Groups in Data: An Introduction to Cluster Analysis.* New York: John Wiley & Sons.

3. Ng, R. and Han, J. (1994). "Efficient and effective clustering method for spatial data mining." in *Very Large Data Bases (VLDB'94).* 144-155.

4. Huang, Z. (1997). "Clustering Large Data Sets with Mixed Numeric and Categorical Values." In *Proceedings of The First Pacific-Asia Conference on Knowledge Discovery and Data Mining, Singapore, World Scientific.*

5. Huang, Z. (1998). "A Fast Clustering Algorithm to Cluster Very Large Categorical Data Sets in Data Mining."

6. Chu, S.C, Roddick, J.F and Pan, J.S. (2002). " Efficient k-medoids algorithms using multi-centroids with multi-runs sampling scheme." in *Workshop on Mining Data for CRM,*

(Taipei, Taiwan), Springer, 2002.

7. Chu, S.C, Roddick, J.F and Pan, J.S. (2002). "An Incremental Multi-Centroid, Multi-Run Sampling Scheme for k-medoids-based Algorithms-Extended Report." in *Second International Conference on Knowledge Discovery and Data Mining.*

8. Han, J. (2001). "Data Mining : Concepts and Techniques".in *Academic Press.* 335-393.