새로운 K-medoids 군집방법 및 성능 비교

Performance Comparison of Some K-medoids Clustering Algorithms

박 해 상　이 상 호　전 치 혁
{shoo359, samo35, chjun}@postech.ac.kr
포항공과대학교 기계산업공학부
경상북도 포항시 남구 효자동 산31

Abstract

We propose a new algorithm for K-medoids clustering which runs like the K-means clustering algorithm and test several methods for selecting initial medoids. The proposed algorithm calculates similarity matrix once and uses it for finding new medoids at every iterative step. To evaluate the proposed algorithm, we use real and artificial data and compare with the clustering results of other algorithms in terms of three performance measures. Experimental results show that the proposed algorithm takes the reduced time in computation with comparable performance as compared to the Partitioning Around Medoids.

1. Introduction

군집분석이란 유사한 속성들을 갖는 객체들을 묶어 전체의 객체들을 몇 개의 군집으로 나누는 것을 말한다. 이를 위한 비계층적 방법으로 K-means 군집방법, K-medoids 군집방법이 많이 사용되고 있다.

사전에 정해진 군집의 수 k 를 바탕으로 K-means 군집방법은 k 개의 중심좌표(centroids)를 선정하여 각 객체와의 거리를 산출한 후 가장 가까운 중심좌표에 그 객체를 배정하는 방법이고 ([1], [2]) K-medoids 군집방법은 k 개의 군집의 대표객체(medoids)를 정하고 객체와 그가 속하는 군집의 대표 객체와의 거리의 총합을 최소로 하는 방법이다. 여기서 군집의 대표객체란 그 군집에 속하는 객체 중 다른 객체와의 거리가 최소의 총합이 최소가 되는 객체를 의미한다.

그러나 K-means 군집방법은 이상치가 있을 경우 성능이 떨어지고 어느 중심좌표에 객체가 하나도 포함이 안될 경우 더 이상 그 중심좌표가 갱신되지 않는다는 단점이 있다. 또한 K-medoids 군집방법 중 가장 성능이 좋다고 알려진 Partitioning Around Medoids (PAM) ([3])은 이상치에는 덜 민감하나 객체 수가 많은 데이터에 대해 계산시간이 오래 걸리는 단점이 있다. ([4])

이 논문은 이 두 방법의 특징을 혼합한 새로운 군집방법을 제안하고 알고리즘의 성능을 비교 분석하였다.

2. Proposed Algorithm

n 개의 객체 각각이 p 개의 변수를 갖고 객체 i 의 j 번째 변수를 X_{ij} (i =1,..., n ; j = 1,..., p) 라 하고 모든 객체를 k 개의 군집으로 분류한다고 할 때 제안하는 방법의 알고리즘은 다음과 같다.

단계 1 : (초기 대표객체를 선정)

1-1. 비유사성 척도로 유클리드 거리(Euclidean distance)를 생각하였을 때 모든 객체 사이의 거리를 다음과 같이 계산한다.

$$d_{ij} = \sqrt{\sum_{a=1}^{p}(X_{ia} - X_{ja})^2} \quad i=1,\ldots,n \quad j=1,\ldots,n \qquad (1)$$

1-2. 중심에 위치하는 객체를 선정하기 위해 다음을 계산한다.

$$p_{ij} = \frac{d_{ij}}{\sum_{l=1}^{n} d_{il}} \quad i=1,\ldots,n \quad j=1,\ldots,n \qquad (2)$$

1-3. 각 객체마다 $\sum_{i=1}^{n} p_{ij}$ 을 계산하고 정렬한 뒤 가장 작은 값을 가지는 k 개의 객체를 초기 대표객체로 선정한다.

1-4. 대표객체로 선정되지 않은 객체에 대하여 가장 가까운 대표객체에 그 객체를 배정한다.

1-5. 각 군집의 대표객체와 군집 내의 모든 객체와의 거리의 총합을 현재의 최적값으로 정한다.

단계 2 : (새로운 대표객체 산출)

같은 군집 내에서 객체 사이의 거리의 합이 최소가 되는 객체를 새로운 대표객체로 선정한다.

단계 3 : (새로운 군집 배정)

3-1. 대표객체로 선정되지 않은 객체에 대하여 가장 가까운 군집에 그 객체를 배정한다.

3-2. 대표객체로부터 그 군집에 속하는 모든 객체까지의 거리의 합을 새로운 최적값으로 정하고 이전 값과 같으면 알고리즘을 마치고 그렇지 않으면 단계 2로 돌아간다.

위에서 제안한 방법은 초기 대표객체에 따라 군집결과가 다를 수 있다. 따라서 초기 대표객체를 선정하는 몇 가지 다른 방법을 제안하고 그에 따른 알고리즘 성능을 다음 절에서 비교할 것이다.

Method 1. Random selection

k 개의 대표객체를 임의로 선정

Method 2. Systematic selection

모든 객체를 어느 한 변수에 대해 정렬한 후 k 개의 구간으로 나누어 각 구간에서 임의로 하나씩 선정

Method 3. Sampling

10% 임의 추출한 객체를 제안한 방법으로 군집분석 후 결과로 나온 대표객체를 초기 대표객체로 선정

Method 4. Outmost objects

중심으로부터 가장 멀리 떨어져 있는 k 개의 객체를 선정

Method 5. Gaussian mixture

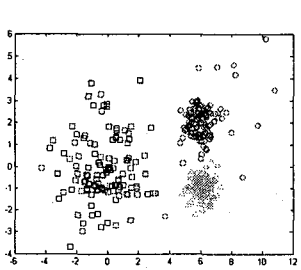
모든 객체가 k 개의 Gaussian 분포에서 나온 것이라 가정하고 각 분포의 평균을 Expectation - Maximization (EM) 알고리즘 ([5])을 이용하여 추정한 뒤 평균과 가장 가까운 객체를 선정

3. Numerical experiments

3.1. Artificial data

제안한 알고리즘의 성능을 알아보기 위하여 Artificial data를 만들어 K-means 군집방법, PAM과 비교하였다.

[그림 1]과 같이 총 세 개의 군집을 갖는 2차원 데이터를 생성하였다.



[그림 1] 알고리즘 비교를 위한 데이터

사각형으로 표시된 군집을 A, 원으로 표시된 군집을 B, 삼각형으로 표시된 군집을 C로 할 때 각 군집의 객체는 [표 1]과 같은 다변량 정규분포에서 120개씩 임의로 추출하였다. 또한 이상치가 존재할 때의 성능을 알아보기 위해 군집 B에 이상치를 추가하였다.

[표 1] 데이터 생성 시 평균과 분산

	군집A	군집B	군집C	이상치
평균	(0, 0)	(6, 2)	(6, -1)	(6, 2)
분산	1.5^2	0.5^2	0.5^2	2^2

[표 2]는 군집방법들의 성능을 각각 adjusted Rand index ([6]), Silhouette ([7]), Index I ([8]) 관점에서 비교한 결과이다. 세 지표 모두 값이 클수록 더 성능이 뛰어나다고 할 수 있다. 결과는 100번 반복 수행하여 평균 낸 수치이고 이상치 %는 군집 B의 객체 120개 중 이상치의 비율을 의미한다.

[표 2] 척도별 알고리즘의 성능 비교

(a) Adjusted Rand index

이상치 %	K-means	PAM	Proposed
0 %	0.7903	0.9679	0.9629
5 %	0.8376	0.9534	0.9335
10 %	0.7836	0.9430	0.9430
15 %	0.7957	0.9288	0.9189
20 %	0.7305	0.9150	0.9115
25 %	0.7708	0.9053	0.8904
30 %	0.7750	0.8952	0.8915
35 %	0.7595	0.8782	0.8609
40 %	0.7624	0.8667	0.8671

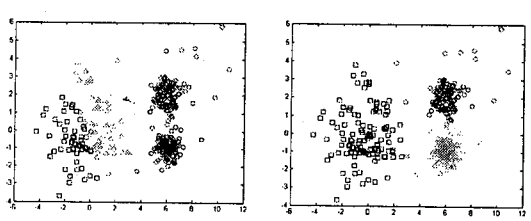
(b) Silhouette

이상치 %	K-means	PAM	Proposed
0 %	0.79757	0.82802	0.82595
5 %	0.78511	0.81524	0.8028
10 %	0.78077	0.80666	0.80487
15 %	0.77571	0.79441	0.78373
20 %	0.76394	0.78607	0.78614
25 %	0.75937	0.77632	0.77502
30 %	0.75308	0.76837	0.76609
35 %	0.73058	0.75602	0.75372
40 %	0.7242	0.75043	0.74957

(c) Index I

이상치 %	K-means	PAM	Proposed
0 %	0.80537	1.4664	1.4416
5 %	0.65205	1.3992	1.3678
10 %	0.88492	1.3601	1.3371
15 %	0.78197	1.3146	1.2886
20 %	0.61187	1.2742	1.2773
25 %	0.61646	1.1981	1.1817
30 %	0.71601	1.1833	1.1661
35 %	0.74959	1.1483	1.119
40 %	0.7411	1.1089	1.102

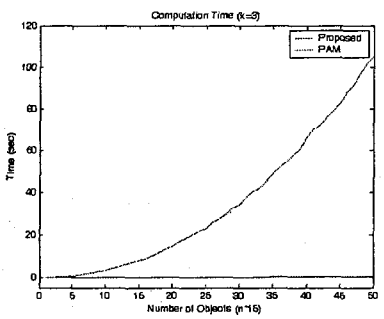
[표 2]에서 PAM과 제안한 방법의 성능이 K-means 군집방법보다 뛰어나다는 것을 알 수 있다. [그림 2]는 K-means 군집방법의 한 결과인데 군집 B와 군집 C를 구분하지 못하고 군집 A를 두 개로 나누는 것을 볼 수 있다. 이는 K-means 군집방법의 이상치에 민감한 단점 때문이라 할 수 있겠다.



(a) (b)

[그림 2] (a) K-means 군집방법의 결과
(b) PAM과 제안한 방법의 결과

제안한 방법과 PAM과의 비교를 위하여 알고리즘 수행시간을 계산하였다. [그림 3]에서 보면 PAM은 객체 수가 증가함에 따라 계산시간이 급격히 증가하고 있는데 비해 제안한 방법은 객체 수에 큰 영향을 받지 않음을 알 수 있다. 실제로 PAM의 계산 복잡도는 $O(k(n-k)^2)$ 이고 제안한 방법은 $O(nk)$ 으로 K-means 군집방법과 같다. ([9])



[그림 3] 제안한 방법과 PAM의 계산시간 비교

3.2. Performance comparison of several methods for selecting initial medoids

초기 대표객체를 선정하는 방법들을 비교하기 위해 3.1.장에서 언급한 바와 같이 10% 이상치를 가지는 데이터를 생성한 후 객체 수를 늘려가면서 adjusted Rand index와 수행시간을 계산하였다. [표 3]의 수치는 100번 반복하여 평균 낸 결과이다.

[표 3] (a) Adjusted Rand index

n	Prop.	Meth1	Meth2	Meth3	Meth4	Meth5
300	0.9393	0.8456	0.6800	0.9124	0.7153	0.9442
600	0.9289	0.8213	0.6562	0.9390	0.7844	0.9446
900	0.9283	0.8160	0.6524	0.9236	0.7075	0.9445
1200	0.9314	0.8493	0.6354	0.9259	0.7665	0.9423
1500	0.9294	0.8100	0.6368	0.9238	0.7526	0.9449
1800	0.9377	0.8396	0.6353	0.9328	0.7779	0.9431
2100	0.9274	0.7990	0.5949	0.9169	0.7258	0.9431
2400	0.9376	0.8288	0.6717	0.9273	0.7458	0.9428
2700	0.9328	0.7885	0.6512	0.9432	0.7312	0.9434
3000	0.9220	0.8091	0.6507	0.9322	0.7151	0.9427

(b) 수행시간(초)

n	Propo	Meth1	Meth2	Meth3	Meth4	Meth5
300	0.088	0.082	0.082	0.078	0.090	0.490
600	0.278	0.264	0.264	0.235	0.280	0.835
900	0.584	0.551	0.564	0.503	0.595	1.290
1200	1.052	0.969	0.993	0.897	1.030	1.921
1500	1.609	1.475	1.566	1.383	1.574	2.596
1800	2.273	2.109	2.226	1.990	2.247	3.424
2100	3.231	3.061	3.091	2.791	3.194	4.487
2400	4.753	4.201	4.391	3.942	4.410	5.848
2700	5.499	5.167	5.356	4.813	5.472	6.942
3000	6.901	6.474	6.778	6.070	6.912	8.380

[표 3] (a)를 보면 Method 5 (Gaussian mixture)의 성능이 가장 좋음을 알 수 있다. 그러나 Gaussian mixture model에서 평균을 추정할 때 많은 시간이 걸린다. 또한 군집의 개수가 늘어나면 추정해야 할 평균이 늘어나므로 계산시간은 급격히 증가한다. 그러나 제안한 방법의 성능은 Method 5만큼 좋으면서 훨씬 수행 시간이 짧은 장점이 있다.

3.3. Iris data

UCI repository [10]에 있는 'Iris' 데이터로 K-means 군집분석과 비교하였다. 'Iris' 데이터는 총 150개로 4개의 측정값을 가지며 50개씩, 3개의 종으로 구분된다. [표 4]와 [표 5]의 분석결과를 보면 K-means 군집분석의 정확도는 88.7%인데 비해 제안한 방법의 정확도는 92%로 이 데이터에 의하면 제안한 방법의 성능이 더 뛰어나다고 할 수 있다.

[표 4] Iris data의 K-means 군집분석 결과

	Setosa (predicted)	Versicolor (predicted)	Virginica (predicted)
Setosa	50	0	0
Versicolor	0	47	14
Virginica	0	3	36

[표 5] Iris data의 제안한 방법의 분석 결과

	Setosa (predicted)	Versicolor (predicted)	Virginica (predicted)
Setosa	50	0	0
Versicolor	0	41	3
Virginica	0	9	47

3.4. Soybean data

UCI repository의 'Soybean' 데이터는 총 47개로 35개의 측정값을 가지며 4개의 종으로 구분된다. [표 6]와 [표 7]의 결과를 보면 'Soybean' 데이터에 대해서도 제안한 방법의 성능이 더 뛰어나다고 할 수 있다.

[표 6] Soybean data의 K-means 군집분석 결과

	Class 1 (predicted)	Class 2 (predicted)	Class 3 (predicted)	Class 4 (predicted)
Class1	10	0	0	0
Class2	10	0	0	0
Class3	0	1	4	5
Class4	0	5	5	7

[표 7] Soybean data의 제안한 방법의 분석 결과

	Class 1 (predicted)	Class 2 (predicted)	Class 3 (predicted)	Class 4 (predicted)
Class1	10	0	0	0
Class2	0	10	0	0
Class3	0	0	8	2
Class4	0	0	7	10

4. Conclusion

본 논문은 K-means 군집분석처럼 수행되는 새로운 K-medoids 군집방법을 제안하였다. 제안한 방법은 모든 객체간의 거리를 한번만 계산하고 반복 단계에서는 이미 계산된 거리를 이용한다. 따라서 알고리즘을 수행하는 과정에서 더 이상 복잡한 계산이 필요 없는 특징이 있다.

시뮬레이션 결과 제안한 방법은 K-means 군집분석에 비해 뛰어난 성능을 보이고 PAM에 비해 적은 수행시간이 걸린다.

또한 초기 대표객체를 선정하는 여러 가지 방법을 제시하고 비교하였다. Gaussian mixture 방법이 조금 나은 성능을 보이지만 수행시간이 큰 단점이 있으므로 성능과 계산시간을 모두 고려해볼 때 제안한 방법에서의 초기 대표객체를 선정하는 방법이 비교적 좋다고 하겠다.

References

[1] MacQueen, J. B. (1967). Some Methods for Classification and Analysis of Multivariate Observations," in Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, 1, 281-297.

[2] Hartigan, J.A. (1975). Clustering Algorithms, New York, NY: Wiley

[3] Kaufman, L. and Rousseeuw, P.J. (1990). Finding Groups in Data: An Introduction to Cluster Analysis. Wiley, New York

[4] Han, J., Kamber, M. and Tung, A. (2001). Spatial clustering methods in data mining: A survey. In Miller, H., and Han, J., eds., Geographic Data Mining and Knowledge Discovery. Taylor & Francis.

[5] Vlassis, N. and Likas, A. (2002). A greedy EM algorithm for Gaussian mixture learning. Neural Processing Letters, 15(1), 77-87.

[6] Hubert, L. & P. Arabie (1985). Comparing partitions. Journal of Classification, 2, 193-218.

[7] Rousseeuw, P.J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20, 53–65.

[8] Maulik, U. and Bandyyopadhyay, S. (2002). Performance Evaluation of some Clustering Algorithms and Validity Indices, IEEE Trans. Pattern Analysis and Machine Intelligence, 24(2), 1650-1654.

[9] Ng, R. and J. Han. (1994). Efficient and Effective Clustering Methods for Spatial Data Mining. Proceedings of the 20th International Conference on Very Large Databases, Santiago, Chile.

[10] ftp://ftp.ics.uci.edu/pub/machine-learning-databases/