ISSN : 1226-7244 (Print)
ISSN : 2288-243X (Online)
논문번호 19-02-12
70

j.inst.Korean.electr.electron.eng.Vol.23,No.2,419~424,June 2019
http://dx.doi.org/10.7471/ikeee.2019.23.2.419

빅데이터에서 개선된 TI-FCM 클러스터링 알고리즘

Improved TI-FCM Clustering Algorithm in Big Data

이 광 규*★

Kwang-Kyug Lee*★

Abstract

The FCM algorithm finds the optimal solution through iterative optimization technique. In particular, there is a difference in execution time depending on the initial center of clustering, the location of noise, the location and number of crowded densities. However, this method gradually updates the center point, and the center of the initial cluster is shifted to one side. In this paper, we propose a TI-FCM(Triangular Inequality-Fuzzy C-Means) clustering algorithm that determines the cluster center density by maximizing the distance between clusters using triangular inequality. The proposed method is an effective method to converge to real clusters compared to FCM even in large data sets. Experiments show that execution time is reduced compared to existing FCM.

요 약

FCM 알고리즘은 반복 최적화 기법을 통해 최적해를 찾는다. 특히, 클러스터링 초기 중심과 잡음의 위치, 몰려있는 밀도의 위치, 개수에 따라 실행시간 차이가 난다. 하지만 이 방법은 중심점을 점차 갱신해 나가는 방법으로 초기 클러스터 중심이 한 쪽으로 치우치게 되고 클러스터링 결과의 편차가 심해 클러스터링 대푯값의 신뢰도가 떨어진다. 따라서 본 논문에서는 삼각부등식을 이용하여 클러스터 간 거리를 최대한 멀어지게 하여 클러스터 중심 밀도를 결정하는 TI-FCM(Triangular Inequality-Fuzzy C-Means:삼각부등식-FCM)클러스터링 알고리즘을 제안한다. 제안된 방법은 대용량의 빅데이터에서도 FCM에 비해 실제 클러스터에 수렴하는 효과적인 방법이고 실험을 통해 기존 FCM보다 실행시간이 감소됨을 보였다.

Key words : Fuzzy C-Means(FCM), K-Means, Data Mining, Big Data, Clustering

Ⅰ. 서론

빅데이터처럼 방대한 양의 데이터는 기존 전통적 데이터베이스 명령어로는 빠르게 처리하기 어려워 데이터를 신속하게 분석하기 위한 새로운 기술과 분석 도구를 요구한다[1, 2]. 빅데이터 분석은 기존의 데이터베이스 분석과는 다른 문제 해결 방법에 초점을 맞춰야 한다[3]. 대용량 데이터를 처리하고 분석하는 데이터 마이닝 작업이 다소 어렵지만, 소프트웨어 분야에서 데이터 마이닝 작업은 매우 중요하고 유용한 도구이다. 데이터 마이닝 기법은 대규모 데이터 저장소에서 유용한 정보를 자동적으로 탐색하는 과정이며, 이 기법은 데이터베이스를 구석구석 뒤져서 모른 채 넘어갈 수 있는 새롭고 유용한 패턴을 탐색하기 위해 적용된다[4, 5]. 이들은 백화점에서 처음 방문한 신규 고객이 100달러

* IT Convergence Engineering, Shinhan University,
★ Corresponding author
E-mail : kklee@shinhan.ac.kr, Tel : +82-31-870-1744
※ Acknowledgment
Manuscript received Jun. 4, 2019; revised Jun. 10, 2019; accepted June. 11, 2019.

이상을 구매할 것인가를 추정하는 것과 유사하게 미래의 결과를 예측하는 기능을 제공하기도 한다. 클러스터링은 동질 그룹에 데이터를 정렬하고 묶는 기술을 사용하여 엄청난 양의 데이터를 정리하고 분류한다. 클러스터링은 데이터를 유사 패턴으로 분할하고 정렬하여 사용자의 의사 결정을 위해 대량의 데이터를 추출하여 유용한 정보를 제공할 수 있어야 한다. 일반적으로 잘 알려진 FCM은 유클리디안 거리를 이용하여 퍼지 소속도를 할당해 줌으로써 클러스터링을 수행하게 된다[5, 6]. 클러스터링은 주어진 데이터의 유사성(similarity)을 기준으로 클러스터를 형성하는 무감독 학습 전략(unsupervised learning strategy)중 하나이며, FCM은 유클리드 거리 척도를 사용해 중심점을 갱신해 나가는 방법으로 초기 클러스터 중심이 한 쪽으로 치우치게 되는 것이 문제점으로 지적되어 왔다. 그러므로 클러스터링 결과의 편차가 심해 클러스터링 대푯값의 신뢰도가 떨어진다. 실제 데이터의 경우 클러스터링의 중심을 나타내는 중심값 보다는 밀도가 몰려있는 중심이 클러스터의 중심값을 대표하는 값으로 보는 것이 타당하다고 볼 수 있으며, 클러스터의 중심이 잘못 결정되면 샘플 데이터가 다른 클러스터에 속하는 문제점이 발생한다[7]. 또한, 초기 클러스터 중심이 한 쪽으로 치우치면 클러스터링 결과가 적절하지 못한 결과가 초래돼 전체 성능을 떨어지는 원인이 될 수 있다[8]. 이에 본 논문에서는 삼각부등식을 이용하여 초기 클러스터 중심 간의 거리를 최대 멀어지게 하고 클러스터 중심을 고르게 분포 되게함으로써, 밀도 중심의 좀더 정확한 클러스터링 결과를 도출하는 개선된 FCM 알고리즘을 제안한다.

논문의 구성은 다음과 같다. 먼저 2장에서 FCM에 대해 설명한다. 3장에서는 FCM의 목적함수 수행시간을 줄이기 위한 개선된 TI-FCM 알고리즘을 제안 하고, 4장 실험 결과를 통해 제안하는 알고리즘의 효율성을 보인다. 결론 및 향후 연구 방향은 5장에서 언급한다.

II. 본론

퍼지 클러스터링에서는 모든 객체들이 모든 클러스터링들에 비해서 0과 1사이의 가중치를 가지고 속하게 된다. 다시 말하면, 클러스터링들은 퍼지 집합처럼 취급된다. FCM 알고리즘은 퍼지 클러스터링 알고리즘 중에서 가장 보편적으로 사용되는 알고리즘으로, FCM 이전의 이진 클러스터링에서 하나의 데이터 포인트는 하나의 클러스터에 소속될 수 있다. 이에 비해 FCM은 소속도를 사용하여 하나의 데이터 포인트가 여러 개의 클러스터에 서로 다른 정도로 소속될 수 있도록 하는 점에서 차이가 있다[9]. 이 알고리즘은 식(1)과 같이 각 데이터와 클러스터 중심과의 거리를 고려한 유사도를 측정한 후 유사도에 기초한 목적함수를 최소화할 수 있도록 데이터 집합을 분할하는 알고리즘이다.

$$J_m(U,V)=\sum_{k=1}^{n}\sum_{i=1}^{c}u_{ik}^{m}\ \|x_k-v_i\|^2 \tag{1}$$

여기서, 모든 데이터 i에 대해 $\sum_{i=1}^{c}u_{ik}=1$이다. n은 데이터의 갯수, c는 클러스터의 갯수 그리고 m은 퍼지 소속정도를 나타내는 가중치이다. $x=\{x_1,x_2,\cdots,x_n\}$인 데이터 벡터 집합과 $v=\{v_1,v_2,\cdots,v_c\}$인 클러스터 중심들 사이의 소속 정도를 $c\times n$인 행렬 $U(=u_{ik})$로 나타낼 수 있다. 식(2)에서 u_{ik}는 k번째 데이터가 i번째 클러스터에 속하는 소속도를 나타내고, 식(3)에서 $v_i(1\le i\le c)$는 i번째 클러스터 중심이다. 여기서 m이 1보다 큰 경우에 모든 i, k에 대해서 $v_i\neq x_k$를 만족한다면 위의 식을 만족할 때만 (U,V)가 J_m의 최소화를 가능하게 한다. 이 알고리즘은 식(2)와 식(3)의 과정을 반복하므로 J_m은 어떤 정해진 값으로 수렴한다.

$$u_{ik}=\left\{\sum_{j=1}^{c}\left(\frac{d_{ik}}{d_{jk}}\right)^{\frac{2}{(m-1)}}\right\}^{-1}, \tag{2}$$

$$v_i=\frac{\sum_{k=1}^{n}u_{ik}^{m}x_k}{\sum_{k=1}^{n}u_{ik}^{m}} \tag{3}$$

III. TI-FCM 알고리즘

FCM에서는 초기 클러스터 중심점이 갱신되면서 클러스터의 중심이 계산되므로 결정된 클러스터 중심이 대푯값을 나타내는데 적절치 못하고 할당

재계산에 소요되는 비용이 증가한다[10]. 그러므로 클러스터 중심을 데이터가 많이 몰려있는 중심으로 결정하기 위해 삼각부등식을 이용한다. 삼각 부등식은 그림 1처럼 FCM의 클러스터 K_1, K_2의 중심점인 c_1, c_2를 계산하는 것이 아닌 d_1, d_2를 중심점으로 결정해 클러스터 간의 중심을 최대한 멀리 선정하고 클러스터의 분류를 명확하게 한다. FCM은 유클리드 거리 중심점을 갱신해가면서 중심을 계산하므로 클러스터 중심이 한 쪽으로 치우치는 현상이 발생한다. 따라서 중심을 밀도 중심으로 산출하고 전체 시스템 성능 시간을 줄이기 위해 삼각부등식을 이용하여 밀도 중심의 최단거리를 계산하는 목적함수를 추가하였다. 제안하여 변형된 FCM의 목적함수는 식(4)와 같다.

$$\max(U,V)\left\{J_m(U,V)=(\alpha)\sum_{k=1}^{n}\sum_{i=1}^{c}u_{ik}^{m}\parallel x_k-v_i\parallel^2\right\} \quad (4)$$

FCM에 비해 추가된 항은 각 관측값에서 임의의 두 변의 길이의 합이 나머지 한 변의 길이보다 큰 삼각부등식의 가중치 $\alpha(>0)$를 구하는 식이다. 여기서 α는 적당한 양의 정수로 식(5)와 같이 삼각형 임의의 변의 최단거리를 선택하여 목적함수에 반영되는 비율을 나타내는 상수이다.

$$\alpha=[\delta(u,v)\le\delta(u,x)+\delta(x,v)] \quad (5)$$

(단, u,v,x는 유클리드 공간 δ에 존재하는 데이터)

α는 밀도 기반의 최단거리를 구하고 FCM 알고리즘을 개선하기 위해 추가되었으며, 각 클러스터데 대해 배타적인 성격을 갖게 되어 클러스터를 명확하게 분류하고 클러스터의 중심을 결정짓는 효과를 가져 올 수 있다. 따라서 목적함수를 최소화하는 조건을 만족하는 u_{ij}를 클러스터 밀도 중심에 따라 균일하게 산출할 수 있다.클러스터의 최단거리 값들의 최대값이 결정되면 목적함수J는 작은 값을 갖게 되므로 전체적인 수행시간이 감소하게 된다. 이때 모든 k에 대해 $\sum_{i=1}^{c}u_{ik}=1$이다. 식(4)와 식(5)의 두 갱신 식을 이용하여 제안하는 TI-FCM 알고리즘을 나타내면 그림 2와 같다.

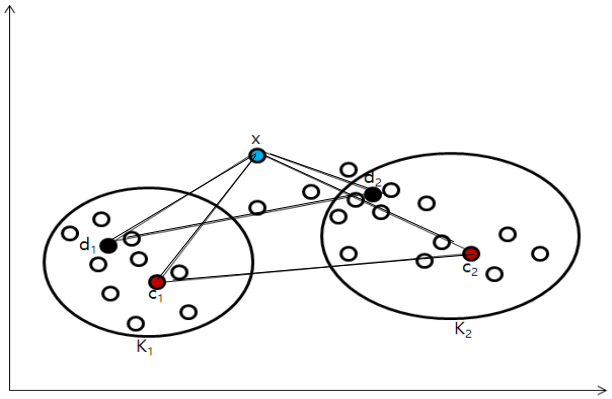

Fig. 1. The shortest distance based on the density of triangular inequality.
그림 1. 삼각부등식의 밀도 기준 최단거리

1 : U : 멤버쉽 데이터
2 : V : 클러스터 중심
3 : V를 초기화 한다
4 : **for** $J_m(U,V)$의 각 데이터 x_k에 대하여 **do**
5 : (4)를 이용하여 U, V를 계산
6 : (4), (5)를 이용하여 α를 계산
7 : **if** $\forall u,v,x \ \exists \ \delta(u,v)\le\delta(u,x)+\delta(x,v)$
8 : **then** $\max(u,v)$를 선택
9 : U,V가 수렴할 때까지
10 : **end if**
11 : **end for**
12 : **return** $\{U,V\}$

Fig. 2. TI-FCM Algorithm.
그림 2. TI-FCM 알고리즘

Ⅳ. 실험 결과

본 절에서는 클러스터링 초기값 결정에 따른 실행시간을 평가하기 위해 데이터 100개에서부터 단계적으로 100만개까지 추가하여 실행하였다. 표 1은 FCM과 제안 알고리즘인 TI-FCM 실행시간 비교이다. FCM의 경우 정확하게 클러스터링 되는 경우도 있으나, 그렇지 않은 경우도 발생하여 클러스터링 결과가 초기 클러스터 중심에 종속적임을 확인 할 수 있다. 그러나 제안 알고리즘은 삼각부등식의 최대값을 이용하므로 비교적 일관성 있게 실행시간이 감소됨을 보였다.

Table 1. FCM vs TI-FCM data execution Time.
표 1. FCM vs TI-FCM 데이터 실행 시간

(단위 : 밀리초)

알고리즘 실행시간 / 데이터 수	FCM	TI-FCM	감소 비율(%)
100	0.0156	0.0115	0.0041
1000	0.0175	0.0120	0.0055
10000	0.3551	0.3170	0.0381
100000	3.5052	3.3100	0.1952
1000000	6.1825	5.6430	0.5395

그림 3에서 보듯이 제안 알고리즘이 FCM보다 실행시간이 감소되었음을 알 수 있다.

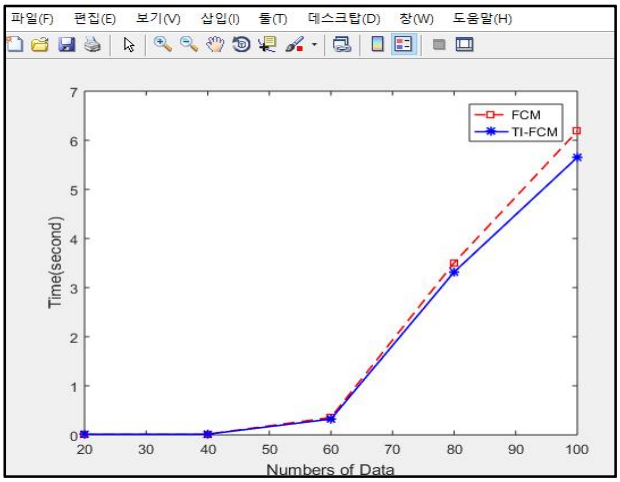

Fig. 3. Comparison of FCM vs TI-FCM execution time.
그림 3. FCM vs TI-FCM 실행 시간 비교

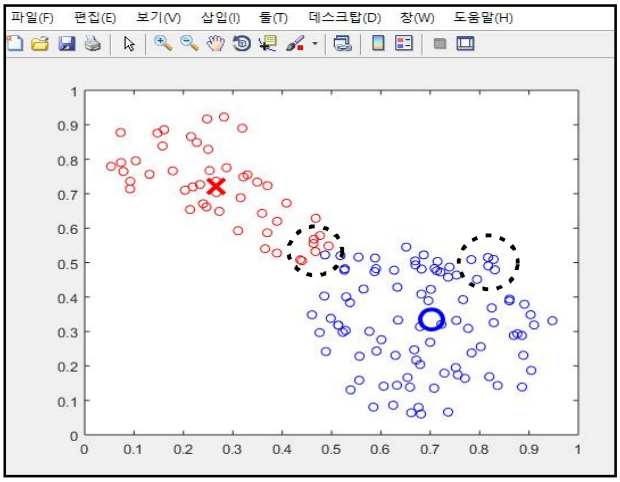

Fig. 4. FCM 100 data execution results.
그림 4. FCM 100개 데이터 실행 결과

그림 4는 초기 100개의 데이터를 클러스터링 했을 경우의 클러스터 중심 결과를 나나낸다. 그림에서 보듯이 클러스터링 중심이 일정치 않고, 클러스터 중심 주변 중심 값을 왜곡시키는 이상치들이 존재한다. 또한, 중심에서 멀리 떨어져 클러스터 분류가 산재되어 혼란스럽기까지 한다. 이는 데이터 집합이 랜덤으로 생성되어 경계가 애매한 부분이 많기 때문에 발생되는 현상이다. 이와 같은 현상은 데이터가 많아질수록 더 많이 나타날 확률이 높으므로 전체 성능이 감소되는 원인이 될 수 있다.

하지만 그림 5는 삼각부등식을 이용해서 클러스터 중심간 거리를 최대로 멀게 하여 밀집도가 높은 데이터의 응집성을 고려하여 클러스터 중심이 결정되고, 중심을 잘 찾을 뿐만 아니라 대체적으로 데이터도 명확하게 분류한다. 또한, 데이터 수를 점차 백만 개 이상의 대량의 데이터로 확대해가며 실험한 결과, 그림 6에 비해 그림 7에서 보듯이 클러스터 중심에서 멀리 떨어져 있는 이상치나 잡음 등 민감한 데이터를 제거하였다. 실험에서 보는 바와 같이 FCM의 클러스터 분류보다 제안 알고리즘이 밀도가 높은 쪽으로 중심을 찾을 확률이 높으며, 클러스터 간 거리도 최대한 멀어지게 하므로 클러스터 경계면 근처의 이상치나 잡음을 제거하여 정확한 클러스터 중심을 찾아 수렴한다. 이 결과는 클러스터 경계에 있는 데이터가 전제 데이터에서 적은 부분을 차지한다 하더라도 전처리 과정에서 사용되는 클러스터링의 결과가 전체적인 성능에 미치는 영향이 적지 않다. 또한, 유클리디안 거리 중심의 클러스터 분류보다 실제 데이터의 밀도가 높은 쪽으로 중심이 결정되므로 클러스터의 대푯값을 나타내어 신뢰도가 높게 된다. 이상에서 살펴

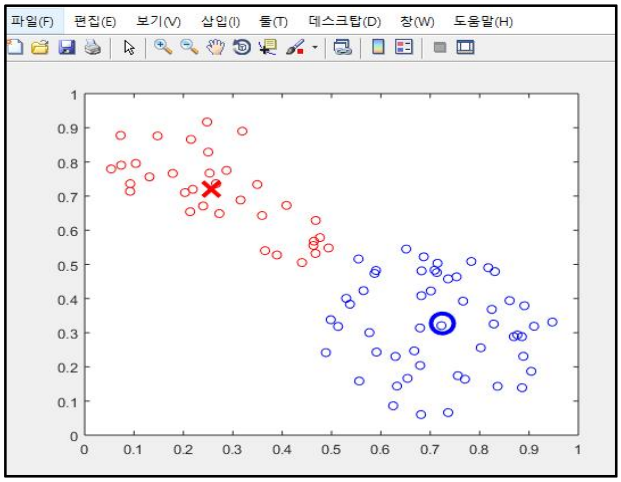

Fig. 5. TI-FCM 100 data execution results.
그림 5. TI-FCM 100개 데이터 실행 결과

본 바와 같이, FCM 알고리즘의 초기 클러스터 중심 선정에 따라 클러스터링 성능이 달라지는 현상을 본 논문에서는 제안된 방법을 통해서 개선할 수 있었으며, 클러스터링 계산시간을 단축시킴으로써 전체적인 실행시간을 줄일 수 있었다. 이는 실제 클러스터의 중심을 찾아낼 확률이 더 높으며, 클러스터 경계면 근처에 데이터가 이상 없이 클러스터에 소속시킨 가능성이 증가되는 것이다.

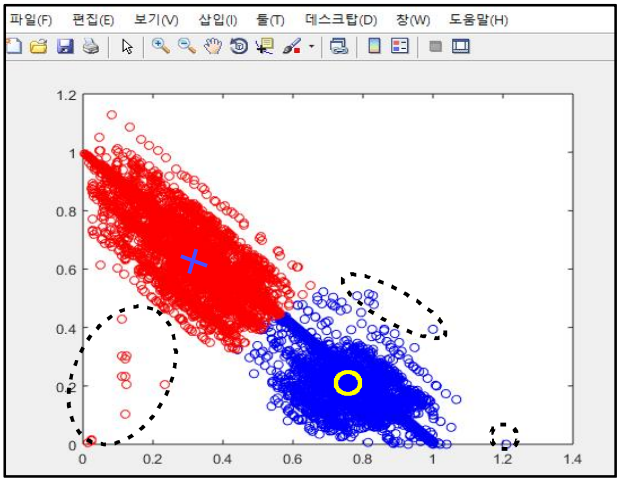

Fig. 6. FCM 1 million data execution results.
그림 6. FCM 100만개 데이터 실행 결과

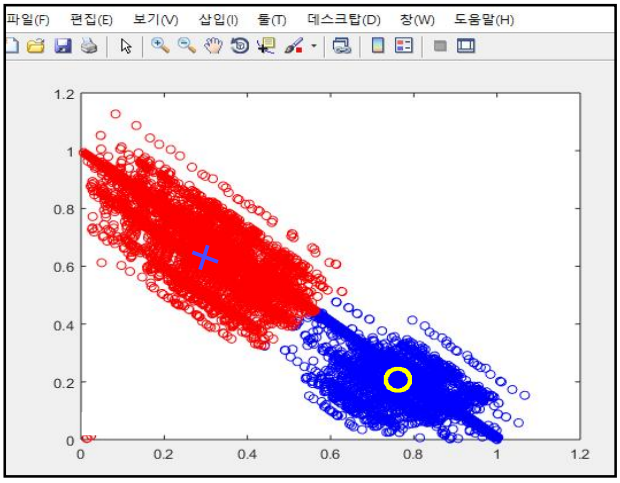

Fig. 7. TI-FCM 1 million data execution results.
그림 7. TI-FCM 100만개 데이터 실행 결과

V. 결론

본 논문에서는 FCM의 단점을 해결하고자 삼각부등식을 이용한 밀도 중심의 개선된 TI-FCM 알고리즘을 제안 하였다. FCM은 간단하면서도 효과적인 클러스터링 방법이지만, 원형 그대로 유클리디안 척도를 사용해 클러스터링 분류가 명확하지 않고 클러스터 중심 경계면 근처에서 값의 왜곡을 불러와 데이터가 잘 못된 클러스터에 소속되거나 오류로 인하여 전체적인 성능을 저하시키는 원인이 되었다. 본 논문에서는 대량의 빅데이터에서도 삼각부등식을 이용한 TI-FCM을 제안하므로서, 초기 클러스터 중심을 랜덤으로 선정하는 방식이 아닌 중심들을 최대한 멀게지게 하므로서 클러스터링 분류 성능을 향상시키고자 하였다. 향후에는 통계적 접근방법을 이용해 클러스터 중심 경계면 근처의 이상치나 잡음을 제거하여 명확한 클러스터 중심을 찾고, 성능을 향상시키는 빅데이터 기반의 이상치 제거 방법을 연구 하고자 한다.

References

[1] http://www-01.ibm.com/software/data/bigdata
[2] Mugdha Jain, Chakradhar Verma, “Adapting k-means for Clustering in Big Data,” *International Journal of Computer Applications (0975-8887),* Vol.101, No.1, 2014. DOI: 10.5120/17652-8457
[3] The Big Data Long Tail. Blog post by Bloomberg, Jason. 2013.
[4] Soumi Ghosh, Sanjay Kumar Dubey, “Comparative Analysis of K-Means and Fuzzy C-Means Algorithms,”(IJACSA) *International Journal of Advanced Computer Science and Applications*, Vol.4, No.4, 2013.
DOI: 10.14569/IJACSA.2013.040406
[5] Anwesha Barai (Deb), Lopamudra Dey, “Outlier Detection and Removal Algorithm in K-Means and Hierarchical Clustering,” *World Journal of Computer Application and Technology,* Vol.5, No.2, pp.24-29, 2017.
DOI: 10.13189/wjcat.2017.050202
[6] J. Bezdek, Pattern Recognition with fuzzy Objective Function Algorithms, New York, Springer, 1981.
[7] Christopher, T., and T. Divya. “A Study of Clustering Based Algorithm for Outlier Detection in Data streams,” *Proceedings of the UGC Sponsored National Conference on Advanced*

etworking and Applications. 2015.
[8] Fahad, A, Alshatri, N., Tari, Z., AlAmri, A., Zomaya, Y., Khalil, I., Foufou, S., Bouras, A, "A Survey of Clustering Algorithms for Big Data: Taxonomy & Empirical Analysis," *Emerging Topics in Computing, IEEE Transaction*s, vol.PP, no.99, pp.1, 1. 2014.
DOI: 10.1109/TETC.2014.2330519
[9] S. Muyamoto, Fuzzy Clustering-Basic Ideas and Overview, Handbook of Computational Intelligence, Springer, pp.293-248, 2015.
[10] J. Nayak, "Fuzzy C-means(FCM) Clustering Algorithm: A Decade Review from 2000 to 2014," *Systems and Technologies,* vol.32, no.2, pp.133-179, 2014.

BIOGRAPHY

Kwang-Kyu Lee (Members)

1985 : B.S Degree in Math, Dongguk University
1991 : M.S Degree in Math, Dongguk University
2002 : Ph.D. Degree in Electronic Computation, Chungbuk National University
1996~current : Professor, IT Convergence Engineering, Shinhan University

※ Interests: Big data, data mining, fuzzy logic, information security