韓國컴퓨터情報學會 論文誌
第15卷 第4號, 2010. 4.

2010-15-4-1-2

커널을 이용한 전역 클러스터링의 비선형화

허 경 용*, 김 성 훈**, 우 영 운***

A Non-linear Variant of Global Clustering Using Kernel Methods

Gyeongyong Heo*, Seong Hoon Kim**, Young Woon Woo***

요 약

Fuzzy c-means(FCM)는 퍼지 집합을 응용한 간단하지만 효율적인 클러스터링 방법 중 하나이다. FCM은 여러 응용 분야에서 성공적으로 활용되어 왔지만, 초기화와 잡음에 민감하고 볼록한 형태의 클러스터들만 다룰 수 있는 문제점이 있다. 이 논문에서는 이러한 FCM의 문제점을 해결하기 위해 전역 클러스터링(global clustering) 기법과 커널 클러스터링(kernel clustering) 기법을 결합하여 새로운 비선형 클러스터링 기법인 커널 전역 FCM(kernel global fuzzy c-means, KG-FCM)을 제안한다. 전역 클러스터링은 클러스터링의 초기화를 위한 방법 중 하나로, 순차적으로 클러스터를 하나씩 추가함으로써 초기화에 민감한 FCM의 한계를 극복할 수 있도록 해준다. FCM의 잡음 민감성과 볼록한 클러스터들만 다룰 수 있는 한계를 극복하기 위한 방법은 여러 가지가 있으며 커널 클러스터링이 그 중 하나이다. 커널 클러스터링은 사용하는 커널을 바꿈으로써 쉽게 확장이 가능하므로 이 논문에서는 커널 클러스터링을 사용하였다. 두 방법을 결합함으로써 제안한 방법은 위에서 언급한 문제점들을 해결할 수 있으며, 이는 가상 및 실제 데이터를 이용한 실험 결과를 통해 확인할 수 있다.

Abstract

Fuzzy c-means (FCM) is a simple but efficient clustering algorithm using the concept of a fuzzy set that has been proved to be useful in many areas. There are, however, several well known problems with FCM, such as sensitivity to initialization, sensitivity to outliers, and limitation to convex clusters. In this paper, global fuzzy c-means (G-FCM) and kernel fuzzy c-means (K-FCM) are combined to form a non-linear variant of G-FCM, called kernel global fuzzy c-means (KG-FCM). G-FCM is a variant of FCM that uses an incremental seed selection method and is effective in alleviating sensitivity to initialization. There are several approaches to reduce the influence of noise and accommodate non-convex clusters, and K-FCM is one of them. K-FCM is used in this paper because it can easily be extended with different kernels. By combining G-FCM and K-FCM, KG-FCM can resolve the shortcomings mentioned above. The usefulness of the proposed method is demonstrated by experiments using artificial and real world data sets.

▸ Keyword : 퍼지 클러스터링 (Fuzzy Clustering), 전역 클러스터링 (Global Clustering), 클러스터 초기화 (Cluster Initialization), 커널 클러스터링 (Kernel Clustering), 비선형 클러스터링 (Non-linear Clustering)

• 제1저자 : 허경용 교신저자 : 우영운
• 투고일 : 2010. 01. 27, 심사일 : 2010. 02. 23, 게재확정일 : 2010. 03. 11.
* Computer and Information Science and Engineering, University of Florida ** 경북대학교 컴퓨터정보학부 교수
*** 동의대학교 멀티미디어공학과 교수

Ⅰ. 서 론

클러스터링은 라벨이 주어지지 않은 데이터 집합 $X=\{x_1,\cdots,x_N\}$를 $K(1<K<N)$개의 균일한 집합 또는 클러스터로 유사도를 기준으로 나누는 작업으로 패턴 인식, 영상 처리, 그리고 최근에는 데이터 마이닝에서 중요한 부분을 차지하고 있다[1]. Fuzzy c-means(FCM)로 대표되는 퍼티 클러스터링은 클러스터링의 중요한 기법 중 하나로 이 논문에서도 FCM의 개선을 목표로 한다. FCM은 간단하면서도 효과적인 클러스터링 알고리듬이지만 몇 가지 문제점이 있다: (1) 클러스터 중심의 초기값을 정하기가 어려우며 초기값에 민감한 결과가 나온다, (2) 노이즈에 민감하다, (3) 볼록한 형태의 클러스터만 다룰 수 있다, (4) 최적의 클러스터 개수를 정하기가 어렵다. 이 논문에서는 마지막 문제를 제외한 나머지 세 가지 문제를 해결하기 위해 기존의 FCM 변형들을 이용하여 새로운 알고리듬을 제안한다. 클러스터의 개수 K는 알려진 것으로 가정한다. 제안한 알고리듬은 FCM의 문제점을 해결하기 위해 먼저 전역 클러스터링, 특히 global FCM(G-FCM)을 바탕으로 하고 있다. G-FCM은 클러스터 중심의 초기값을 결정성(deterministic) 알고리듬을 통해 증량적으로(incremental) 설정하므로 항상 동일한 초기값을 얻을 수 있고, 초기화에 따른 민감성을 줄일 수 있는 방법이다[2]. 다른 두 가지 문제를 해결하기 위해서는 커널 클러스터링, 특히 kernel FCM(K-FCM)을 이용하여 G-FCM이 커널 공간에서 수행되도록 하였다. 이 논문에서는 커널 기반 방법이 가지는 비선형성과 더불어 잡음 민감성을 줄여주는 것으로 알려진 코시(Cauchy) 커널을 사용하였다[3].

FCM이 국부 최적해만을 보장하며 이 또한 초기화(initialization)에 민감하다는 것은 널리 알려진 사실이다. 또한 전역적인 최적해를 구하는 문제는 NP-hard이므로, 여러 가지 초기화 방법을 통해 부 최적해(suboptimal solution)를 구하는 문제는 널리 연구되어 왔다. 클러스터링의 초기화 방법은 크게 샘플링 방법, 거리 최적화 (distance optimization) 방법, 밀도 추정(density estimation) 방법의 3가지로 나누어볼 수 있다[4]. 샘플링 방법은 무작위로 정한 클러스터 중심으로 클러스터링을 수행하며, 이 과정을 여러 번 반복해서 그 중 최고 또는 평균 결과를 최종 결과로 내는 방법이다. 이 방법은 반복 회수에 따라 얻어진 해의 품질이 달라지며 많은 연산량을 요구하는 단점이 있다. 거리 최적화 방법은 순차적으로 기존의 초기값들에서 가장 먼 데이터 포인트를 초기값으로 추가하는 방법이다. FCM을 포함한 대다수의 클러스터링은 클러스터 내의 변이를 최소로 하고자 한다. 따라서 거리 최적화 방법은 클러스터링 이전에 클러스터 사이의 거리를 최대로 함을 목표로 한다. 이 방법에 속하는 일부 알고리듬 역시 여러 번의 수행을 요구하며 초기값이 변두리 지역의 데이터를 고르는 경향이 있다, 즉, 잡음에 해당하는 데이터 포인트가 초기값으로 선택될 가능성이 있다. 밀도 추정 방법은 국부적인 밀도를 정의하고 이를 기준으로 초기값을 선택한다. 따라서 국부 밀도의 추정이 매우 중요하며 이를 위해서 추가적인 파라미터를 필요로 하는 단점이 있다.

전역 클러스터링 역시 밀도 추정에 속하는 방법으로, 다른 여타의 방법들이 알고리듬을 여러 번 수행해야 하거나 추가적인 파라미터를 필요로 하는 반면, 전역 클러스터링에는 그런 문제점이 없다. 전역 클러스터링은 k-means에 처음 적용되었고[5][6], 이후 FCM에 확장되어 적용되었다[2]. 이 논문에서는 이를 더 확장하여 K-FCM에 적용되도록 하였다. 제안한 방법, kernel global FCM(KG-FCM)은 G-FCM을 통해 초기화에 민감한 문제점을 완화하고, K-FCM을 통해 잡음 민감성을 줄이고 오목한 형태의 클러스터도 다룰 수 있도록 해준다.

이 논문의 구성은 다음과 같다. 2장과 3장에서는 hard clustering과 soft clustering에서의 전역 클러스터링을 설명한다. 4장에서는 향상된 클러스터링 방법을 제안하며, 5장에서는 실험을 통해 제안한 방법과 기존 방법을 비교한다. 결론 및 향후 연구 방향에 대해서는 6장에서 언급한다.

Ⅱ. Global K-Means

k-means는 주어진 데이터 집합에서 K 개의 hard cluster를 찾아내는 알고리듬으로[7] 식 (1)의 목적함수를 반복적으로 최적화한다.

$$E_{k-means}(V)=\sum_{i=1}^{N}\sum_{k=1}^{K} I(x_i\in C_k)\|x_i-v_k\|^2 \quad \cdots\cdots (1)$$

이 때 $I(\cdot)$는 지시 함수로 데이터 포인트 x_i가 클러스터 C_k에 속하는 경우 1을 그렇지 않은 경우 0의 값을 가진다. 일반적으로 K 개의 클러스터 중심 $V=\{v_1,\cdots,v_K\}$는 데이터 포인트를 이용하여 무작위로 초기화하는 방법이 많이 사용된다. k-means는 현재도 사용되는 방법이기는 하지만 초기화에 민감한 문제점이 있다. 이러한 문제점을 해결하기 위해 여러 가지 방법이 제안되었고 global k-means(GKM)가 그 중 하나이다[5][6]. GKM은 순차적으로 클러스터의 개수를 하나씩 늘려가면서 K개까지 클러스터링을 수행하며, 각 클러스터링 과정에서 결정성 알고리듬을 통해 클러스터 중심의 초기값을 하

나씩 추가한다. GKM은 $k(1 \le k \le K)$ 개의 클러스터로 이루어지는 k-clustering 문제의 최적해를 이전의 (k-1)-clustering 문제의 해와 N 개의 데이터 포인트 중 하나를 k 번째 클러스터 초기값으로 설정한 클러스터링 문제로부터 얻을 수 있다고 가정한다. 따라서 GKM은 k-clustering 문제의 최적해를 얻기 위해 N번의 k-means를 수행하여야 한다. 따라서 전체적으로는 $K \times N$번의 k-means를 수행하여야 하며, 데이터 포인트의 수가 많아지는 경우 연산량이 기하급수적으로 증가하는 문제점이 있다. 따라서 GKM의 변형인 fast GKM이 일반적으로 사용된다. Fast GKM은 N번 k-means를 수행하지 않고, 각 데이터 포인트에 대해 초기값으로의 적합도를 계산하여 그 중 최적의 데이터 포인트를 선택하여 k-means를 한 번만 수행한다. 따라서 fast GKM은 K번의 k-means 만을 수행하면 된다. Fast GKM에서 적합도는 식 (2)와 같이 계산된다.

$$b_i = \sum_{j=1}^{N} \max\left(\left(d_{k-1}^{j}\right)^2 - \|x_i - x_j\|^2, 0\right) \quad \cdots\cdots (2)$$

이 때 d_{k-1}^{j}은 x_j와 (k-1)-clustering에서 얻어진 클러스터 중심들 중 x_j에 가장 가까이에 있는 중심까지의 거리를 나타낸다. 식 (2)의 b_i는 새로운 k번째 클러스터의 중심으로 데이터 포인트 중 하나를 설정함으로써 얻을 수 있는 식 (1)의 최대 감소치를 나타낸다. 따라서 fast GKM에서는 b_i를 최대로 하는 x_i를 새로운 클러스터의 초기값으로 설정해준다. Fast GKM 알고리듬은 그림 1과 같이 요약할 수 있다.

```
Input : 클러스터 수 K, 데이터 집합 X
1 : V_1 = {(1/N) Σ_{i=1}^{N} x_i}
2 : for k = 2 to K
3 :     for l = 1 to N
4 :         J_k^l = Σ_{j=1}^{N} max((d_{k-1}^j)^2 - ||x_i - x_j||^2, 0)
5 :     end
6 :     α = argmax_{1 ≤ l ≤ N} J_k^l
7 :     V_k = V_{k-1} ∪ {x_α}
8 :     V_k ← k-means(X, V_k)
9 : end
10: return V_K
```

그림 1. Fast GKM 알고리듬
Fig. 1. Fast GKM algorithm

III. Global Fuzzy C-Means

FCM이 퍼지 소속도(membership)를 통해 k-means를 확장한 것처럼, G-FCM 역시 퍼지 소속도를 통해 GKM을 확장한 것이다. 식 (1)의 목적함수에 퍼지 소속도를 추가함으로써 식 (3)의 FCM 목적함수를 얻을 수 있다.

$$E_{FCM}(U, V) = \sum_{i=1}^{N} \sum_{k=1}^{K} u_{ki}^{m} \|x_i - v_k\|^2 \quad \cdots\cdots (3)$$

이 때 u_{ki}는 i번째 데이터 포인트가 k번째 클러스터에 소속될 정도를 나타내고 m은 퍼지화 정도를 나타내는 상수(fuzzifier constant)이다. G-FCM 알고리듬은 그림 1과 유사하지만 두 가지 측면에서 다르다. 첫 번째, 단계 8에서 k-means가 아닌 FCM을 이용한다. 식 (4)와 (5)는 FCM을 위한 소속도와 클러스터 중심의 update equation을 나타낸다.

$$u_{ki} = \frac{\|x_i - v_k\|^{-2/(m-1)}}{\sum_{j=1}^{K} \|x_i - v_j\|^{-2/(m-1)}} \quad \cdots\cdots (4)$$

$$v_k = \frac{\sum_{i=1}^{N} u_{ki}^{m} x_i}{\sum_{i=1}^{N} u_{ki}^{m}} \quad \cdots\cdots (5)$$

두 번째 차이점은 클러스터 중심의 초기값을 선택하기 위한 적합도 계산식이다. 식 (2)는 식 (1)을 최소화시키는 데이터 포인트를 찾아내는 것을 목표로 한다. 유사하게 G-FCM에서는 식 (3)을 최소화시키는 데이터 포인트를 찾아내어야 한다. 식 (3)은 식 (4)를 이용해서 식 (6)과 같이 나타낼 수 있다[8].

$$E_{FCM}(U, V) = \sum_{i=1}^{N} \left(\sum_{k=1}^{K} \|x_i - v_k\|^{2/(1-m)} \right)^{1-m} \quad \cdots\cdots (6)$$

따라서 그림 1의 단계 4에서는 식 (7)을 계산하며, 이를 최소로하는 데이터 포인트가 새로운 클러스터 중심의 초기값으로 설정된다.

$$J_k^l = E_{FCM}(U, V_{k-1} \cup \{x_l\}) \quad \cdots\cdots (7)$$

IV. Kernel Global Fuzzy C-Means

전역 클러스터링은 초기화 민감성을 줄일 수 있는 방법으

로 알려져 있다. 하지만 G-FCM 역시 잡음에 민감하며 볼록한 형태의 클러스터만 다룰 수 있는 단점이 있다. 이를 해결하기 위해 여러 가지 방법이 제안되었고 커널 클러스터링이 그 중 하나이다. 커널 클러스터링은 Girolami에 의해 kernel k-means가 처음으로 소개되었고[9], 이후 많은 클러스터링 알고리듬들이 커널 공간에서 수행되도록 변환되었다[10]. 커널을 기반으로 하는 알고리듬들은 기존의 선형 알고리듬을 커널 공간에서 동작하도록 함으로써, 피처 공간에서 비선형 알고리듬을 수행한 것과 같은 효과를 얻는다.

K-FCM의 목적함수는 식 (8)과 같이 식 (3)을 변형하여 얻을 수 있다.

$$E_{K-FCM}(U,V)=\sum_{i=1}^{N}\sum_{k=1}^{K}u_{ki}^{m}\|\phi(x_i)-\phi(v_k)\|^2 \quad \cdots\cdots (8)$$

이 때 $\phi(\cdot)$는 사상함수(mapping function)로 데이터 포인트를 고차원의 커널 공간(kernel space)으로 옮기는 역할을 한다. K-FCM의 update equation은 식 (9) 및 (10)과 같다.

$$u_{ki}=\frac{\|\phi(x_i)-\phi(v_k)\|^{-2/(m-1)}}{\sum_{j=1}^{K}\|\phi(x_i)-\phi(v_j)\|^{-2/(m-1)}} \quad \cdots\cdots (9)$$

$$v_k=\frac{\sum_{i=1}^{N}u_{ki}^{m}\phi(x_i)}{\sum_{i=1}^{N}u_{ki}^{m}} \quad \cdots\cdots (10)$$

사상함수에 의해 변환된 커널 공간은 차원이 아주 높고 때때로 무한대로 여겨지므로 실제 변환된 커널 공간에서의 데이터 포인트를 얻는 것은 불가능할 수 있다. 하지만 커널 공간에서 두 포인트의 내적이 커널함수(kernel function)로 구해지는 경우 사상함수를 통해 사상된 포인트를 구하지 않고도 알고리듬 수행이 가능하다. 이를 커널 트릭(kernel trick)이라 하며 커널 기반의 알고리듬에서 핵심적인 역할을 한다. 따라서 커널을 이용하는 알고리듬에서 커널 함수의 선택은 매우 중요하다. 이 논문에서는 일반적으로 많이 사용되는 가우시안 커널이 아닌 코시 커널을 사용하였다. 코시 커널은 비선형성과 더불어 잡음에 강건한 것으로 알려져 있다[3]. 코시 커널은 식 (11)과 같이 정의된다.

$$\kappa(x,y)=\frac{1}{1+\beta\|x-y\|^2} \quad \cdots\cdots (11)$$

이 때 β는 커널 파라미터이다. 식 (11)을 이용하여 K-FCM의 update equation은 식 (12) 및 (13)과 같이 나타낼 수 있다.

$$u_{ki}=\frac{(1-\kappa(x_i,v_k))^{-1/(m-1)}}{\sum_{j=1}^{K}(1-\kappa(x_i,v_j))^{-1/(m-1)}} \quad \cdots\cdots (12)$$

$$v_k=\frac{\sum_{i=1}^{N}u_{ki}^{m}\kappa(x_i,v_k)^2x_i}{\sum_{i=1}^{N}u_{ki}^{m}\kappa(x_i,v_k)^2} \quad \cdots\cdots (13)$$

일반적으로 커널 클러스터링에서는 피처 공간의 클러스터 중심 좌표를 구해낼 수 없고 근사화된 좌표만을 얻을 수 있다. 하지만 코시 커널을 사용하는 경우에는 식 (13)과 같이 중심의 좌표를 구할 수 있다. 식 (12)와 (13)에서 $\kappa(x_i,v_k)$는 데이터 포인트 x_i와 클러스터 중심 v_k의 유사도로 가중치의 역할을 한다. 이 가중치는 중심과 소속도를 구하는 과정에서 클러스터 중심과 상이한 점을 잡음으로 처리할 수 있도록 해줌으로써 잡음 민간성을 감소시킨다.

전역 클러스터링을 커널 클러스터링에 사용하기 위해서는 식 (6)의 적합도 함수 역시 커널 공간에서 평가되도록 식 (14)와 같이 수정되어야 한다.

$$J_k^l=\sum_{i=1}^{N}\left(\sum_{j=1}^{k}(\|\phi(x_i)-\phi(v_j)\|^2)^{1/(1-m)}\right)^{1-m} \quad \cdots\cdots (14)$$

식 (14)에서 데이터 포인트와 클러스터 중심 사이의 거리 $\|\phi(x_i)-\phi(v_j)\|$는 (k-1)-clustering 문제에서 결과로 얻어진 초기값과 데이터 포인트에서 얻어진 초기값의 두 가지로 식 (15)와 같이 나누어진다.

$$J_k^l=\sum_{i=1}^{N}\left(\begin{matrix}\sum_{j=1}^{k-1}(\|\phi(x_i)-\phi(v_j)\|^2)^{1/(1-m)}\\ +(\|\phi(x_i)-\phi(x_l)\|^2)^{1/(1-m)}\end{matrix}\right)^{1-m} \quad \cdots\cdots (15)$$

코시 커널을 사용하는 경우 식 (15)는 식 (16)과 같이 간단하게 나타낼 수 있다.

$$J_k^l=\sum_{i=1}^{N}\left(\begin{matrix}\sum_{j=1}^{k-1}(2-2\kappa(x_i,v_j))^{1/(1-m)}\\ +(2-2\kappa(x_i,x_l))^{1/(1-m)}\end{matrix}\right)^{1-m} \quad \cdots\cdots (16)$$

Input : 클러스터 수 K, 데이터 집합 X, 클러스터 중심의 초기값 V_{k-1}, 데이터 포인트 색인 α

$V_K = V_{k-1} \cup x_\alpha$
repeat
  for k = 1 to K
    for i = 1 to N
      $$u_{ki} = \frac{(1-\kappa(x_i, v_k))^{-1/(m-1)}}{\sum_{j=1}^{K}(1-\kappa(x_i, v_j))^{-1/(m-1)}}$$
    end
    $$v_k = \frac{\sum_{i=1}^{N} u_{ki}^m \kappa(x_i, v_k)^2 x_i}{\sum_{i=1}^{N} u_{ki}^m \kappa(x_i, v_k)^2}$$
  end
until U satisfies the convergence criterion
return V_K, U_K

그림 2. 초기값이 주어진 경우 K-FCM 알고리듬
Fig. 2. Algorithm of K-FCM with initialization

마지막으로 한 가지 유의할 점은 클러스터링 과정에서 update equation을 사용하는 순서이다. 일반적으로 FCM에서는 소속도와 클러스터 중심의 갱신 순서에 크게 영향을 받지 않는다. 하지만 전역 클러스터링에서는 클러스터 중심의 초기값이 결정성 알고리듬으로 정해지므로 식 (12)를 이용하여 소속도를 먼저 갱신하여야 한다. 그림 2는 초기값이 주어지는 경우 K-FCM 알고리듬을 나타낸 것이며, 이를 이용하여 KG-FCM은 그림 3과 같이 요약할 수 있다.

Input : 클러스터 수 K, 데이터 집합 X,

$V_1 = \left\{ \frac{1}{N} \sum_{i=1}^{N} x_i \right\}$
for k = 2 to K
  for l = 1 to N
    $$J_k^l = \sum_{i=1}^{N} \left(\sum_{j=1}^{k} (\|\phi(x_i) - \phi(v_j)\|^2)^{1/(1-m)} \right)^{1-m}$$
  end
  $\alpha = \mathrm{argmax}_{1 \le l \le N} J_k^l$
  $[V_k, U_k] \leftarrow K\text{-}FCM(X, V_{k-1}, \alpha, k)$
end
return V_K, U_K

그림 3. KG-FCM 알고리듬
Fig. 3. KG-FCM algorithm

V. 실험 결과 및 고찰

제안한 방법이 기존 방법에 비해 나은 결과를 보인다는 것을 확인하기 위해, 네 가지 클러스터링 알고리듬(FCM, G-FCM, K-FCM, KG-FCM)을 Matlab으로 구현하고 실험하였다. 먼저 전역 클러스터링의 유효성을 입증하기 위해 그림 4에 나타나 있는 데이터(D_7)를 이용하여 실험하였다.

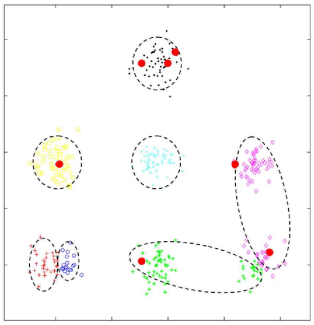

그림 4. FCM의 초기화 민감성
Fig. 4. Sensitivity to initialization in FCM

D_7은 7개의 원형 클러스터로 구성되어 있으며, 그림 4에 표시된 점은 무작위로 선택된 클러스터 중심의 초기값을 나타낸다. 대부분의 경우 FCM 역시 7개의 클러스터를 찾아낼 수 있지만, 그림 4의 경우에서처럼 국부 최적해만을 구하는 경우가 있다. 그림 4에서는 7개의 클러스터 중 3개의 클러스터만을 정확하게 찾아내고 있다. 그림 5는 G-FCM에서 결정성 알고리듬을 이용하여 클러스터 중심의 초기값을 증량적으로 결정하는 방법을 보이고 있다. KG-FCM 역시 G-FCM과 유사한 방식으로 동작하지만 피처 공간이 아닌 커널 공간에서 수행된다는 점이 다르다.

표 1은 D_7에 네 가지 알고리듬을 적용한 경우, 정확하게 찾아낸 클러스터의 개수를 요약한 것이다. 표에서 주어진 값들은 100회 반복 실험하여 평균한 것이다. 표 1에서 알 수 있듯이 FCM은 종종 국부 최적해에 빠져서 정확한 클러스터를 찾아내지 못한다. K-FCM은 FCM에 비해 나은 성능을 보이지만 역시 국부 최적해에 빠지는 경우가 있다. 하지만 전역 클러스터링은 항상 정확한 클러스터를 찾아내고 있음을 알 수 있다.

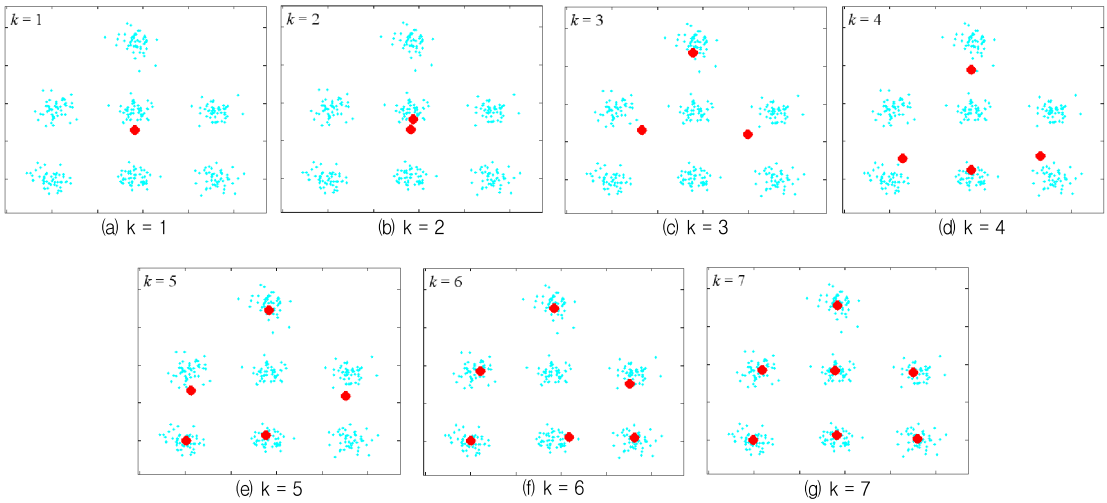

그림 5. 증량적 알고리듬에 의한 클러스터 중심의 초기값 설정
Fig. 5. Incremental initialization method in global clustering

표 1. D_7의 클러스터링 결과
Table 1. Clustering results using D_7

알고리듬	N
FCM	6.82
G-FCM	7.00
K-FCM	6.92
GK-FCM	7.00

제안한 방법의 잡은 민감성 감소를 실험하기 위해 D_7에 백색 잡음을 첨가하고 앞의 실험을 반복하였다. 이 실험에서 잡음 비율은 잡음의 수를 데이터의 개수로 나눈 값이다. 그림 6은 각 알고리듬이 찾아낸 클러스터의 수를 잡음 비율에 대한 그래프로 나타낸 것이다. 그림에서 알 수 있듯이, FCM은 잡음에 민감하여 잡음 비율이 증가함에 따라 성능이 떨어지고 있음을 알 수 있다. K-FCM은 FCM 보다 나은 성능을 보이긴 하지만, 잡음 비율이 증가함에 따라 역시 성능이 떨어진다. 반면 두 가지 전역 클러스터링 알고리듬은 잡음이 증가하여도 성능에 큰 변화가 없음을 알 수 있으며, 제안한 KG-FCM이 그 중 가장 나은 성능을 보임을 알 수 있다. D_7의 경우 피처 공간에서 볼록한 형태의 클러스터들로 구성되어 있으므로 G-FCM과 KG-FCM의 성능 차이가 크게 나지 않았다.

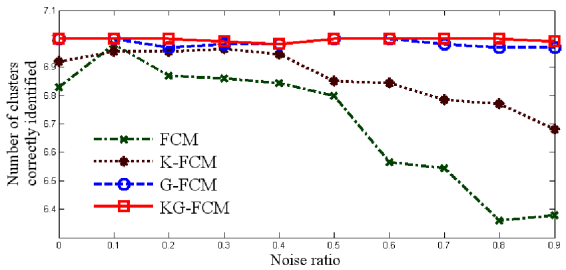

그림 6. 잡음 비율에 따라 찾아낸 클러스터의 수
Fig. 6. Number of correctly identified clusters with respect to noise ratio

제안한 방법은 UCI Machine Learning Repository[11]의 실제 데이터에 적용하여 그 유효성을 검증하였다. 서로 다른 클러스터링 방법들을 비교하기 위해 이 논문에서는 정보 변화량(variation of information)[12]을 수정해서 사용하였다. 두 개의 서로 다른 클러스터링 $C=\{C_1,\cdots,C_K\}$와$C'=\{C_1,\cdots,C_{K'}\}$을 가정해 보자. 각각의 방법은 데이터를 K와 K' 개의 클러스터로 나누며 클러스터링 C'은 알려진 클래스 라벨에 해당한다. 어떤 데이터 포인트가 클러스터 C_k에 포함될 확률은 $p(k)=N_k/N$로 나타낼 수 있으며, 이 때 N_k는 클러스터 C_k에 포함된 데이터 포인트의 수를, N은 전체 데이터 포인트의 수를 나타낸다. 클러스터링 C'에 포함된 정보량은 랜덤 변수 $p(k')$의 엔트로피로 식 (17)로 나타낼 수 있다.

$$H(C') = -\sum_{k=1}^{K'} p(k')\log p(k') \quad \cdots\cdots (17)$$

또한 클러스터링 C와 C'에 공통된 정보량은 랜덤 변수 $p(k)$와 $p(k')$의 상호 엔트로피(mutual entropy)로 나타낼 수 있다.

$$I(C,C') = \sum_{k=1}^{K}\sum_{k'=1}^{K'} p(k,k')\log\frac{p(k,k')}{p(k)p(k')} \quad \cdots\cdots (18)$$

이 때 $p(k,k')$은 어떤 데이터 포인트가 클러스터링 C에서는 C_k에 속하고, 클러스터링 C'에서는 클러스터 $C_{k'}$에 소속될 확률로 식 (19)로 계산된다.

$$p(k,k') = \frac{|C_k \cap C'_{k'}|}{N} \quad \cdots\cdots (19)$$

위의 엔트로피들을 이용하여 두 클러스터링 사이의 차이는 식 (20)으로 계산할 수 있다.

$$D_I(C,C') = H(C') - I(C,C') \quad \cdots\cdots (20)$$

이 논문에서 클러스터의 개수 K는 알려진 클래스의 개수 K'와 동일하게 설정되었다.

표 2는 UCI 데이터 집합에 클러스터링을 수행하고 알려진 클래스 라벨과의 거리를 요약한 것이다. 표 2의 값들은 식 (20)의 값을 계산한 것이며, 전역 클러스터링이 아닌 경우에는 100회 반복 실험하여 평균한 값을 나타내었다. 표 2에서 알 수 있듯이, 전역 클러스터링이 무작위 초기화 방법에 비해 나은 성능을 보임을 알 수 있다. 또한 피처 공간에서의 클러스터링에 비해 커널 공간에서의 클러스터링이 나은 성능을 보였다. 전체적으로는 제안한 커널 공간에서의 전역 클러스터링인 KG-FCM이 최고의 성능을 보임을 알 수 있다.

표 2. UCI 데이터의 클러스터링 결과
Table 2. Clustering results using UCI data sets

데이터 집합	FCM	G-FCM	K-FCM	KG-FCM
Iris	0.4042	0.4041	0.3898	0.3898
Wine	1.3088	1.2944	1.2368	1.2284
Breast Cancer	0.9339	0.9329	0.9224	0.8712
Yeast	1.6139	1.6129	1.8001	1.5855

VI. 결론

FCM은 간단하면서도 효율적인 클러스터링 알고리듬으로 퍼지 클러스터링에서 대표적인 방법 중 하나이다. FCM은 여러 영역에서 성공적으로 사용되었지만 몇 가지 문제점이 있다. 이 논문에서는 FCM의 문제점 중 초기화 민감성, 잡음 민감성, 그리고 선형 클러스터에만 적용 가능한 문제점을 극복하기 위해 기존의 G-FCM과 K-FCM을 결합하여 kernel global FCM을 제안하였다. 제안한 KG-FCM은 위의 문제점들을 완화할 수 있으며 기존 클러스터링 방법에 비해 나은 성능을 보임을 실험 결과를 통해 확인할 수 있다.

KG-FCM이 기존 방법에 비해 나은 성능을 보이지만 개선의 여지는 남아있다. 커널 클러스터링이 이 논문에서 사용된 이유는 사용되는 커널에 따라 다양한 효과를 얻을 수 있기 때문이다. 또한 이는 주어진 데이터에 적합한 커널을 사용함으로써 나은 성능을 얻을 수 있는 가능성을 시사한다. 단순화된 예로 커널 파라미터 β를 데이터에 따라 자동으로 결정하는 방법은 현재 연구 중에 있다.

참고문헌

[1] R. Xu and D. Wunsch, “Survey of clustering algorithms,” IEEE Transactions on Neural Networks, Vol. 16, No. 3, pp. 645 - 678, May 2005.

[2] W. Wang, Y. Zhang, Y. Li, and X. Zhang, “The global fuzzy c-means clustering algorithm,” Proceedings of the 6th World Congress on Intelligent Control and Automation, Dalian, China, pp. 3604 - 3607, June 2006.

[3] H.-S. Tsai, “*A study on kernel-based clustering algorithms*,” Ph.D. dissertation, Department of Applied Mathematics, Chung Yuan Christian University, Chung Li, Taiwan, 2007.

[4] J. He, M. Lan, C.-L. Tan, S.-Y. Sung, and H.-B. Low, “Initialization of cluster refinement algorithms: A review and comparative study,” Proceedings of the 2004 IEEE International Joint Conference on Neural Networks, Budapest, Hungary, pp. 297-302, July 2004.

[5] A. Likas, N. Vlassis, and J. J. Verbeek, "The global k-means clustering algorithm," Pattern Recognition, Vol. 36, No. 2, pp. 451-461, Feb. 2003.

[6] A. M. Bagirov, "Modified global k-means algorithm for minimum sum-of-squares clustering problems," Pattern Recognition, Vol. 41, No. 10, pp. 3192-3199, Oct. 2008.

[7] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, Berkeley, CA, pp. 281-297, Jan. 1966.

[8] R. J. Hathaway and J. C. Bezdek, "Optimization of clustering criteria by reformulation," IEEE Transactions on Fuzzy Systems, Vol. 3, No. 2, pp. 241-245, May 1995.

[9] M. Girolami, "Mercer kernel-based clustering in feature space," IEEE Transactions on Neural Networks, Vol. 13, No. 3, pp. 780-784, May 2002.

[10] M. Filippone, F. Camastra, F. Masulli, and S. Rovetta, "A survey of kernel and spectral methods for clustering," Pattern Recognition, Vol. 41, No. 1, pp. 176-190, Jan. 2008.

[11] A. Asuncion and D. Newman, UCI Machine Learning Repository : *http://www.ics.uci.edu/~ mlearn/MLRepository.html*, 2007.

[12] M. Meila, "Comparing clusterings-an information based distance," Journal of Multivariate Analysis, Vol. 98, No. 5, pp. 873-895, May 2007.

저 자 소 개

허 경 용

1996년 8월 : 연세대학교 본대학원 전자공학과 (공학석사)

2009년 12월 : Department of Computer and Information Science and Engineering, University of Florida (공학박사)

관심분야 : Machine Learning, Pattern Recognition, Image Processing

김 성 훈

1996년 2월 : 연세대학교 본대학원 전자공학과 (공학박사)

1996년 3월~2006년 2월 : 영동대학교 컴퓨터공학과 부교수

2006년 3월~현재 : 경북대학교 컴퓨터정보학부 부교수

관심분야 : 인공지능, 패턴인식, 지능형콘텐츠

우 영 운

1991년 8월 : 연세대학교 본대학원 전자공학과(공학석사)

1997년 8월 : 연세대학교 본대학원 전자공학과(공학박사)

1997년 9월~현재 : 동의대학교 멀티미디어공학과 교수

관심분야 : 인공지능, 패턴인식, 퍼지이론, 의료정보