J. of Basic Sciences, Cheju Nat'l Univ.
18(2), 121~127, 2005

기초과학연구 제주대학교
18(2), 121~127, 2005

데이터마이닝에서의 군집분석 응용

강 형 창 · 김 철 수
제주대학교 전산통계학과

요 약

대량의 데이터를 분석함으로써 많은 양의 정보를 얻어낼 수 있다. 군집화는 이러한 데이터를 다루는 방법 중 하나로써 데이터를 그룹화하거나 분류한다. 일반적으로 데이터를 분류하는 분석 방법으로 군집분석이 사용된다. 군집분석은 군집 내 객체들은 유사성은 작게 하고 군집 간 유사성은 크게 분류하는 것을 목적으로 한다. 본 논문에서는 군집분석에 사용되는 여러 알고리즘에 대한 효율성을 알아보고자 한다.

1. 서 론

컴퓨터와 인터넷이 발전함에 따라 기업이나 모든 조직들은 정보 인프라로 데이터 베이스를 구축하게 되었으며, 데이터 베이스는 방대하게 커졌다. 방대한 데이터베이스로부터 새로운 지식을 얻고자 하는 과정을 KDD(Knowledge Discovery in Database)라 하며 이러한 한 분야가 데이터 마이닝이다. 데이터 마이닝 방법 중 데이터들이 비슷한 특징을 갖는 그룹으로 분류하여 그 그룹들의 특성이나 대표성을 찾는 과정을 군집분석이라 한다.

군집분석은 데이터 마이닝에서 중요한 비지도학습(Unsupervised Learning)의 한 방법으로 상호연관성에 근거하여 서로 동질적인 집단으로 분류하는 기법이며, 군집분석을 통해 생성된 군집에 관한 정보는 하나의 설명변수로 다른 분석에 사용될 수 있다.

군집분석은 크게 계층적 군집분석(Hierarchical Clustering)과 분리 군집분석(Partitioning Clustering)으로 구분되며 계층적 군집분석은 군집의 수를 미리 결정하지 않으나 한 군집 내에 다른 군집이 포함될 수 있으며, 분리 군집분석은 군집의 수를 미리 결정하여야 하고 한 군집 내에 다른 군집이 포함되지 않아야한다.

k-means 알고리즘(k-means clustering)은 분리 군집분석에서 가장 많이 알려져 있는 방법 중 하나이다. k-means 알고리즘(Hartigan, 1974)은 데이터에 있는 각 객체를 유사한 특성을 지니는 k개의 그룹으로 분할하는 방법으로 각 군집에 속하는 객체들의 평균값을 중심점으로 하여 근접한 거리에 있는 객체를 묶어서 분할하게 된다. k-means 알고리즘은 특별한 변환이 필요 없고 데이터 마이닝에 적용이 쉽다는 장점으로 많은 분야에 이용되고 있지만, 이 방법은 군집의 수 k에 대한 정보를 사전에 결정해야 하며 k개 군집의 중심인 초기값(seed) 결정이 군집 형성에 상당한 영향을 준다. 부적절한 초기값의 결정은 잘못된 군집의 생성과 군집 생성과정에서 발생

하는 많은 반복으로 인해 많은 시간이 소용되며, 또한 군집분석 성능에 상당한 영향을 미치게 된다(Bae and Roh, 2005). k-means 알고리즘은 중심계산을 평균으로 하기 때문에 이상치에 민감한 단점이 있다. 이러한 단점을 해결하기 위해 실제 관측치인 메도이드(medoid)를 중심으로 사용하는 알고리즘(k-medoids clustering)으로 PAM(Partitioning Around Medoids), CLARA(Clustering LARge Applications) 및 CLARANS(Clustering Large Applications based on RANdomized Search)등이 연구되어 왔으며 이들 알고리즘들은 k-means 알고리즘보다 이상치에 덜 민감하다고 알려졌다(Kaufman and Rousseeuw, 1990). 이러한 알고리즘들은 수치 데이터에만 적용가능하며 비수치(범주)데이터는 군집분석을 수행할 수 없다.

비수치 속성을 가진 데이터를 군집화하기 위해 k-means 알고리즘을 확장한 k-modes 알고리즘(Huang, 1998)은 비유사도 측정을 통하여 군집화 데이터를 분류하고 발생 도수를 기반으로 군집화의 중심을 결정하였다.

대량의 데이터와 다양한 종류의 변수 타입을 저장하고 있는 데이터베이스에서 데이터 마이닝을 위한 군집분석을 수행하기 위해서는 대량 처리가 가능하여야 하고, 변수 타입에 맞는 군집분석 방법을 선택하여야 하여야 한다.

2장의 1절에서는 k-means 알고리즘과 초기값 결정 방법을 설명하고, 2절에서는 k-medoids 알고리즘을 설명하고, 3절에서는 k-modes 알고리즘을 설명한다. 3장에서는 데이터 마이닝에서 군집화를 위한 방법에 대해 개선되어야 할 내용을 설명한다.

2. 군집분석 알고리즘

2.1. k-means 알고리즘과 초기값 결정 방법

k-means 알고리즘은 현재 분리 군집분석 방법 중 보편적으로 많이 쓰이는 알고리즘의 하나로 군집 내 유사성은 작게 하고, 군집 간 유사성은 크게 분류 하는 것이 목적이다. 군집의 유사성은 군집의 중심인 평균과 객체들간의 거리로 측정한다. k-means 알고리즘은 중심과 주어진 객체의 거리를 계산하여 가장 가까운 중심에 주어진 객체를 할당하는 방법이다. k-means 알고리즘은 군집의 수 k가 미리 결정되어 있어야 하며 중심을 평균으로 계산하기 때문에 평균을 구할 수 없는 비수치(범주)데이터 에는 적용할 수 없다.

〈표 1〉 k-means 알고리즘

Step 1. 데이터를 k개의 초기 군집으로 분할한다.
Step 2. 분할된 k개의 중심을 평균을 이용하여 구한다.
Step 3. 각 객체와 중심들 사이의 거리를 계산하여 객체가 현재 속해있는 군집 중심에 가까우면 현재 군집에 포함하고, 다른 군집의 중심에 가까우면 그 군집으로 재분류한다.
Step 4. 할당되는 객체가 없을 때까지 Step 2와 Step 3을 반복한다.

k-means 알고리즘에서 초기값 결정은 군집의 형성 및 군집 형성 시간에 대한 중요한 요인이 되므로 이에 대한 많은 연구가 진행되어 왔다. MA(Macqueen Approach) 방법(Macqueen, 1976)은 데이터에서 k개의 초기값을 선택하고 나머지 객체들은 초기값에 가장 가까운 군집으로 포함한 후, 군집의 중심을 다시 계산하여 군집의 중심의 변화량이 임계값

(threshold)이하가 될 때까지 반복하여 군집을 형성하게 된다. 이 방법은 랜덤하게 초기값을 선택하기 때문에 쉽고 편하게 사용할 수 있으나 초기값이 부적절하게 선택되었을 때는 잘못된 군집을 형성할 수 있다.

KA(Kaufman Approach)방법(Kaufman and Rousseeuw, 1990)은 데이터의 가장 중앙에 위치한 관측치를 첫 번째 초기값으로 설정하고, 나머지 초기값은 첫 번째 초기값과 일정한 거리 이상 떨어져 있으면서 군집이 형성되기 쉽도록 초기값을 하나 선택하게 했다. 이 방법은 MA보다는 정교하지만 초기값을 구한 후 다음 단계의 초기값을 하나씩 구하는 과정에서 주변의 모든 관측값들을 고려하기 때문에 데이터의 크기가 커지는 경우 계산량이 많아진다.

Max-Min 방법(Bae and Roh, 2005)은 단계적으로 초기값을 선택하였을 때 선택된 초기값들이 다음 초기값을 정하는데 정보를 줄 수 있도록 하되, KA방법보다는 적은 계산을 하지 않도록 고안되었다. 데이터에서 랜덤하게 하나의 관측값을 선택하여 첫 번째 초기값으로 선택하고, 첫 번째 초기값에서 나머지 관측값과의 거리를 구하여 그 거리를 최대로 하는 관측값을 두 번째 초기값으로 선택한다.

2.2. k-medoids 알고리즘

2.2.1. PAM 알고리즘

PAM 알고리즘은 k-medoids의 한 방법이며 군집의 실제 객체인 메도이드를 중심으로 사용한다. 메도이드란 군집 내에서 객체들간의 평균 비유사성이 가장 작은 객체를 말한다. 이상적인 메도이드를 찾기 위해 반복을 통하여 메도이드들을 변화시켜, 메도이드들을 변화시킬 때마다 객체들이 가까운 메도이드들을 중심으로 객체를 형성하기 위해 재분류된다. 객체들이 변화된 메도이드로 인하여 이동한 객체와 본래의 메도이드, 변화된 메도이드와 거리의 차를 비용 이라한다.

비용함수는 메도이드와 나머지 객체들간의 차이 값으로 계산된다. PAM 알고리즘에서는 객체들이 이동하면서 발생한 비용을 모두 더한 총 비용을 이용하여 이상적인 메도이드를 찾아낸다.

$$TC_{ik} = \sum_{j} C_{jih} \qquad (2\text{-}1)$$

〈표 2〉 PAM 알고리즘

Step 1. 임의로 k개의 초기값을 선택
Step 2. k개의 객체를 데이터에서 임의로 추출하여 현재 메도이드인 O_i로 설정
Step 3. 현재 메도이드인 O_i에 가까이 있는 객체들로 군집을 분류
Step 4. 분류된 k개의 군집 내에서 비유사성을 계산하여 가장 좋은 메도이드 O_k를 찾음
Step 5. 현재 O_i와 새로운 메도이드 O_k 사이의 $\min_{O_i, O_k} TC_{ik}$를 가지는 메도이들 찾는다. 만약 $\min_{O_i, O_k} TC_{ik}$가 음수이면, 현재 메도이드 O_i를 새로운 메도이드 O_k로 바꾸고 Step 3으로 되돌아간다.
Step 6. 음수가 아닐 경우가 발생할 때까지 Step 3, Step 4, Step 5를 반복하여 평균 비유사성이 가장 낮은 군집을 찾는다.

k-means 알고리즘에서는 군집을 거리의 차이로 사용하지만, PAM 알고리즘은 비용함수로 대신한다. 이 알고리즘도 군집의 수 k를 미리 결정해야 한다. PAM 알고리즘은 모든 경우의 데이터에 대해 계산을 하므로, 데이터의 크기가 커질수록 계산량이 많아 컴퓨터의 수행속도가 느려지는 단점이 있다(Han and Kamber, 2000).

2.2.3 CLARA 알고리즘

CLARA 알고리즘은 대량 데이터를 효율적으로 다루기 위한 방법이다. 군집을 나눌 때 데이터의 표본을 랜덤 추출한 후 표본에 PAM 알고리즘을 적용시켜 표본에서 k개의 최적의 메도이드를 구한다.

표본 추출과 메도이드를 찾는 과정을 반복하여 최적의 메도이드를 찾는다. 군집의 정확성을 측정할 경우 표본에서 구해진 메도이드들과 표본의 객체들간의 평균 비유사성을 구하는 것이 아니라, 표본에서 구해진 메도이드들과 전체 데이터의 모든 객체들의 평균 비유사성을 계산한다. 이 알고리즘도 군집의 수 k를 미리 결정해야 한다.

〈표 3〉 CLARA 알고리즘

Step 1. 전체 데이터에서 임의로 표본 추출하여 그 표본에 PAM 알고리즘을 적용시켜 k개의 메도이드를 찾는다.

Step 2. Step 1에서 구한 k개의 메도이드를 중심으로 전체 데이터를 이용해 메도이드에 가까운 객체들로 군집을 형성한다.

Step 3. 전체 데이터로 형성된 군집을 이용해 평균 비유사성을 계산한다. 만약 계산된 값이 현재 값보다 작으면 계산된 값을 현재 값으로 바꾼다.

Step 4. 메도이드가 수렴할 때까지 Step 1, Step 2, Step 3을 반복한다.

CLARA 알고리즘은 데이터에서 표본을 추출하기 때문에 표본 크기에 의존한다. PAM 알고리즘은 전체 데이터에서 메도이드를 추출하지만 CLARA는 표본에서 최적의 메도이드를 찾는 것이다. 만약 추출된 표본에 좋은 메도이드가 없다면 CLARA는 최적의 군집을 찾지 못할 수도 있다.

2.2.4. CLARANS 알고리즘

CLARANS 알고리즘은 그래프의 개념을 이용한다. 그래프 $G_{n,k}$는 n개의 객체, k개의 군집을 가지는 데이터에서 각각의 노드들의 집합을 말한다. 노드란 메도이드들의 집합인 $\{O_{m1}, \cdots, O_{mk}\}$을 말한다. 만약 두 개의 노드에서 한 개 메도이드만 다르고 다른 메도이드들은 동일하다면 이 두 노드는 이웃(neighbor)이라고 한다. 즉 노드 $S_1 = \{O_{m1}, \cdots, O_{mk}\}$과 $S_2 = \{O_{w1}, \cdots, O_{uk}\}$는 $|S_1 \cap S_2| = k-1$로 두 개의 노드에 $k-1$개의 공통 메도이가 있는 것이다. 각각의 노드들은 $k(n-k)$개의 이웃들을 가진다.

〈표 4〉 CLARANS 알고리즘

Step 1. CLARANS 알고리즘에 이용할 총 반복 횟수와 평가할 이웃들의 수를 설정한다.

Step 2. 반복 횟수인 i의 값을 1로 초기화한다.

Step 3. 그래프 $G_{n,k}$에서 현재 사용할 노드를 뽑아 초기 노드로 사용한다.

Step 4. 현재 노드에서 평가할 이웃들이 가지는 값의 개수만큼 표본을 뽑아 이웃들 사이의 비용을 계산하다. 비용 계산은 식 (2.1)을 사용한다.

Step 5. 만약 표본으로 사용한 이웃들간의 비용이 더 작다면 현재 비용을 작은 비용으로 갱신하고 Step 3으로 돌아가 이웃들 사이의 비용을 다시 계산한다. 이 과정을 표본의 이웃들 수 만큼 반복한다.

Step 6. 반복한 후 최소 비용을 현재 최소 비용으로 저장하고, 수렴할 때까지 Step 3과 Step 4를 반 복한다.

Step 7. 반복 횟수가 총 반복 횟수가 될 때까지 전체 과정을 반복한다.

현재 메도이드 O_i가 새로운 메도이드 O_h로 이동할 경우 발생하는 비용은 PAM 알고리즘에서 정의한 (2.1)의 TC_{ih}를 이용하여 계산한다. 메도이드를 변형시킬 때, 메도이드 O_i는 S_1에 속하는 객체이고, 새로운 메도이드 O_h는 S_2에 속하는 객체이다. O_i, $O_h \notin S_1 \cap S_2$로 메도이드 O_i와 O_h는 노드의 교집합에 속하지는 않지만, $O_i \in S_1$, $O_h \in S_2$으로 노드에 속하는 유일하게 다른 하나의 메도이드로 새로운 메도이드로 설정한다.

CLARANS는 노드의 모든 이웃들을 평가하지는 않는다(노드의 이웃의 표본을 추출하여 평가). CLARA 알로리즘은 반복할 때마다 정해진 표본의 개수만큼 표본을 추출하고, CLARANS 알고리즘은 단계를 거칠 때마다 노드의 이웃의 표본을 추출하는 것이다. 따라서 CLARANS도 표본 추출에 의존한다.

2.3. k-modes 알고리즘

$X = \{x_1, \cdots, x_n\}$의 n개의 범주형 데이터 $x_j (1 \le j \le n)$를 군집속성 $A = \{A_1, \cdots, A_p\}$에 대한 범주속성이라 정의하자.

각 범주속성 $A_l (1 \le l \le p)$은 도메인 값들이고, 이를 $DOM(A_l) = \{a_l^{(1)}, \cdots, a_l^{(n_l)}\}$으로 표시하자. n_l은 범주속성 A_l의 범주값(categorical value)들의 개수를 의미한다. 이에 따라 x_j는 $[x_{j,1}, \cdots x_{j,p}]$로 나타낸다. 따라서 x_j는 속성값들과 결합하여 논리적으로 다음과 같이 나타낼 수 있다.

$[A_l = x_{j,1}] \wedge [A_j = x_{j,2}] \wedge \cdots \wedge [A_p = x_{j,p}]$

여기서, $x_{j,l} \in DOM(A_l)$, $1 \le l \le p$이다.

k-modes 알고리즘은 범주형 데이터 X를 k개의 군집으로 군집화하기 위해 다음과 같은 함수를 최소하는 기법을 사용한다.

$$J_m(V:X) = \sum_{i=1}^{k} \sum_{j=1}^{n} (\mu_{i,j})^m d_c(v_i, x_j) \quad (2.2)$$

여기서, $\mu_{i,j}$는 k-modes 알고리즘에서 x_j가 i번째 군집에 소속되었을 때 $\mu_{i,j} = 1$이고, 다른 경우에는 $\mu_{i,j} = 0$이다. 군집의 중심을 $V = \{v_1, \cdots, v_k\}$라고 하면, 각 범주형 군집에서 중심 v_i는 p개의 후보로써 $[v_{i,1}, \cdots, v_{i,p}]$로 나타낼 수 있다. 식 (2.2)에서 모수 m은 각 데이터의 소속을 제어하기 위한 양(positive)의 개수 이다.

범주형 데이터를 군집화하기 위해 k-modes 알고리즘에서는 군집의 중심과 범주형 데이터의 거리를 측정하고 군집화의 각 단계에서 군집 중심을 갱신한다. k-modes 알고리즘에서 중심 v_i와 범주형 데이터 x_j사이의 거리값 $d_c(v_i, x_j)$는 다음과 같이 정의한다.

$$d_c(v_i, x_j) = \sum_{l=1}^{n} \delta(v_{i,l}, x_{j,l}) \quad (2.3)$$

여기서, $v_{i,l} = x_{j,l}$일 경우 $\delta(v_{i,l}, x_{j,l}) = 0$이 되며, $v_{i,l} \neq x_{j,l}$일 경우 $\delta(v_{i,l}, x_{j,l}) = 1$이 된다. i번째 mode를 i번째 군집의 중심 $v_i = [v_{i,l}, \cdots, v_{i,p}]$라고 할 때 이 중심의 갱신은 각 $v_{i,l} \in v_i (1 \le l \le p)$에 대하여 다음과 같이 갱신한다.

$$v_{i,l} = a_l^{(r)} \in DOM(A_l) \quad (2.4)$$

여기서, $a_l^{(r)}$은 다음과 같은 조건을 만족한다.

$$| \{\mu_{i,j} | x_{j,l} = a_l^{(r)}, \mu_{i,j} = 1\} | \geq | \{\mu_{i,j} | x_{j,l} = a_l^{(t)}, \mu_{i,j} = 1\} |, \ 1 \leq t \leq n_l \quad (2.5)$$

k-modes 알고리즘에서 군집 중심 v_i에 대한 속성 $v_{i,l}$의 군집은 i번째 군집에 소속된 데이터의 집합에서 속성 A_l의 군집에 대한 빈도 형태로 결정된다.

3. 향후 개선점

데이터 마이닝의 특징은 대량 데이터(giga, tera bytes)이고, 데이터들은 다양한 타입(internal, ratio, binary, ordinal, nominal,…)을 갖는다는 것이다. 데이터 마이닝에서 k-means 알고리즘은 대량 데이터를 처리할 수 있는 특징을 갖고 있지만 수치 데이터에만 제한되며, 평균을 군집의 중심으로 계산하기 때문에 잡음 또는 이상치에 민감하고, 군집의 개수를 미리 결정해야 한다. 그리고 k-means 알고리즘은 부분 최적(Local Optimum) 문제가 발생할 수 있다.

k-means 알고리즘의 부분 최적 문제 및 이상치 문제를 보완하기 위한 방법으로 PAM 알고리즘을 이용하여 실제 객체 값인 medoid를 군집의 중심으로 하는 방법이 있다. 그러나 PAM 알고리즘은 대량의 데이터를 처리하는데 비효율적이므로 이를 보완하기 위한 방법으로 CLARA 알고리즘 및 CLARANS 알고리즘을 이용할 수 있다. 이 방법들은 대량의 데이터를 처리할 수 있으나 데이터에서 표본을 추출한 다음 medoid를 찾아내어 군집을 형성하기 때문에 부적절한 표본의 선택으로 인해 최적의 medoid를 찾아낼 수 없을 수 있다. 그리고 이러한 알고리즘들은 수치 데이터만 처리하고, 군집의 수를 미리 결정해야 한다.

범주 데이터를 군집화 하기 위해서 계층적 군집분석 방법을 고려하여 수치 데이터와 범주 데이터에 대한 군집을 처리할 수 있으나 대량 데이터를 처리하기에는 제한적이다. 범주 데이터를 군집화하기 위해 범주 데이터를 이진 데이터로 변형하여 k-means 알고리즘을 이용할 수 있으나 범주의 수가 많은 경우 대량 데이터 처리를 위한 비용 함수가 증가하게 되고, 또한 이진 데이터(0, 1)의 값이 군집의 특성을 설명하기는 어렵다고 할 수 있다.

k-modes 알고리즘은 범주 데이터를 효율적으로 군집화 할 수 있지만, 단일의 중심을 사용하고 있고 간단한 거리도 측정을 하여 정확도나 대량의 범주형 데이터에 적용하기에는 문제점을 내포하고 있다.

참고문헌

[1] Bae and Roh (2005). A Study on K-Means Clustering. The Korean Communications in Statistics Vol. 12 No. 2, 2005 pp. 497-508.

[2] Han. J and Kamber. M (2000). Data Mining : Concepts and Techniques, The Morgan Kaufman Series in Data Management Systems, Morgan Kaufman Publishers.

[3] Hartigan J.A (1974). Clustering Algorithms. John Wiley & Sons, New York.

[4] Huang, Z (1998). Extensions to the K-means Algorithm for Clustering Large Data Sets with Categorical Values, Data Mining and Knowledge

Discovery, 2, 283-304.

[5] Kaufman L and Rousseeuw P.J (1990). Finding Groups in Data. An Introduction to Cluster Analysis. John Wiley & Sons, Canada.

[6] Lee and An (2003). Journal of Korean Data & Information Science Society 2003. Vol. 14, No.4 pp. 725-736.

[7] MacQueen. J (1967).. Some methods for classification and analysis of multivariate observation. Proc. 5th Berkeley Symp. Math. Statist. Prob., 1:128-297.

Clustering Applications In Data Mining

Hyung Chang Kang · Chul Soo Kim

Department of Computer Science and Statistics, Cheju National University

Abstract

We encounter a large amount of information and store or represent it as data, for further analysis and management. One of the vital means in dealing with these data is to classify or group them into a set of categories or clusters. Clustering is a popular approach to implementing the partitioning operation. Clustering methods partition a set of objects into clusters such that objects in the same cluster are more similar to each other than objects in different clusters according to some defined criteria. In this article we investigate the several methods of clustering algorithm and compare it efficiency for feature study.