고차원 데이터에서 점진적 프로젝션을 이용한 클러스터링

이 혜 명[v], 박 영 배
명지대학교 컴퓨터공학과

hmlee@kmc.ac.kr, parkyb@wh.myongji.ac.kr

A Clustering using Incremental Projection for High Dimensional Data

Hye-Myung Lee[v], Young-Bae Park

Dept. of Computer Engineering, Myongji University

요 약

데이터 마이닝의 방법론 중 클러스터링은 데이터베이스 객체들의 애트리뷰트 값에 근거하여 유사한 그룹으로 식별하는 기술적인 작업이다. 그러나 대부분 알고리즘들은 데이터의 차원이 증가할수록 형성된 전체 데이터 공간은 매우 방대하므로 의미있는 클러스터의 탐색이 더욱 어렵다. 따라서 효과적인 클러스터링을 위해서는 클러스터가 포함될 데이터 공간의 예측이 필요하다. 본 논문에서는 고차원 데이터에서 각 차원에 대한 점진적 프로젝션을 이용한 클러스터링 방법을 제안한다. 제안한 방법에서는 클러스터가 포함될 가능성이 있는 데이터공간의 후보영역을 결정하여, 이 영역에서 점들의 평균값을 중심으로 클러스터를 탐색한다.

1. 서 론

데이터마이닝은 대용량 데이터베이스에 암시적으로 존재하는 숨겨진 유용한 패턴을 찾아내는 방법론을 일컫는다. 데이터마이닝 기법 중에서 클러스터링은 데이터 공간의 데이터 점들에서 유사한 특징을 가진 점들을 집단화하는데 매우 유용하다. 클러스터 분석은 데이터마이닝을 위한 우선적인 방법론이 될 수 있다. 이것은 독자적인 도구로서 심도있는 분석을 하거나 얻어진 데이터의 분포에서 지식을 얻을 수 있으며, 다른 알고리즘 수행을 위한 전처리 단계로 사용할 수 있다. 또한 데이터베이스 분야에서도 유사성 검색, 고객 분류, 경향 분석 등을 위한 도구로서 널리 연구되고 있는 실정이다.

이와 같이 클러스터링은 데이터베이스 연구의 많은 응용분야에서 널리 사용되고 있다. 그러나 잘 알려진 대부분의 클러스터링 알고리즘들은 고차원 데이터 공간에서 클러스터를 탐색하는데 실패하는 경향이 있다. 이와 같은 문제점은 데이터 점들이 갖는 자체의 희소성(sparsity)으로 인한 것으로 차원 전체가 클러스터 탐색에 관련되지 않을 수 있다는 것이다[3][4]. 이와 같은 문제를 다루는 방법으로서 연관성 있는 차원을 선택하고 대응하는 부분공간(subspace)에서 클러스터를 탐색하고자 하는 연구가 진행되고 있다. 관련성 있는 차원만을 고려하는 알고리즘으로는 CLIQUE[3], PROCLUS[4] 등이 있으며, 이들은 데이터 공간상의 점들은 전체차원의 부분집합 즉 일부차원에 대하여 보다 효과적으로 클러스터를 탐색할 수 있다는 데에 의의가 있다.

본 논문에서는 대용량의 고차원 데이터 공간에서 효과적인 클러스터링을 위한 알고리즘으로 CLIP(CLustering based on Incremental Projection)을 제안한다. CLIP은 각 차원의 점진적인 프로젝션을 통하여 클러스터 형성에 연관성이 적은 차원은 제외시켜 클러스터가 포함될 후보영역을 결정한 후, 그 후보영역에서 데이터 점들의 평균값을 이용하여 보다 정확한 클러스터 형태를 식별하고자 하는 클러스터링 기법이다. 이와 같은 점진적 프로젝션을 이용함으로써 클러스터를 탐색할 데이터 공간을 줄일 수 있으며, 클러스터가 포함될 수 있는 후보영역에서 데이터 점들의 평균 근처의 점들을 우선적으로 탐색하여 클러스터의 형태를 보다 정확히 얻고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 클러스터링 알고리즘에 대하여 설명하고 3장에서 제안하는 알고리즘을 소개하며, 마지막 4장에서는 결론 및 향후 발전방향을 고찰하고자 한다.

2. 관련연구

2.1 클러스터링 알고리즘의 요구사항

다양한 데이터마이닝 응용분야의 출현에 따라 클러스터링 기법에서도 알고리즘 개발 동기가 되는 다음과 같은 요구사항들이 제기되고 있다[3].

(1) 고차원 데이터에 관한 효과적 처리

하나의 객체 즉 데이터 레코드는 일반적으로 수십 개의 애트리뷰트로 구성되며, 각 애트리뷰트의 도메인은 매우 방대하다. 이와 같은 고차원 공간에서 점들의 평균 밀도로 클러스터를 찾는 것은 큰 의미가 없다. 이러한 문제를 복합적으로 생각하면, 애트리뷰트의 차원들 중에서 많은 차원과 이들의 조합은 잡음(noise or outlier)이나 불균형하게 분포된 값을 가질 수 있다. 그러므로 데이터의 모든 차원을 사용하는 거리 함수는 비효율적이다. 더욱이, 임의의 클러스터들은 애트리뷰트의 조합으로 구성된 다른 부분차원에 존재할지도 모른다.

(2) 결과에 대한 해석력

데이터마이닝 응용분야는 일반적으로 최종 사용자에 의해 쉽게 이해될 수 있는 명세를 요구한다. 대부분의 시각화 기법들은 고차원 공간에서는 잘 수행되기 어려우므로 특히 단순한 표현방법은 매우 중요하다고 할 수 있다.

(3) 확장성

클러스터링 기법은 속도가 빨라야 하며, 차원의 수와 입력의 크기를 확장할 수 있어야 한다. 그리고 데이터 레코드가 표현되는 순서에 무관해야 한다.

현재 클러스터링 기법들은 위의 사항들을 모두 다루고 있지는 않으나, 상당한 연구들이 각 사항별로 언급하고 있는 실정이다.

2.2 관련 차원을 고려하는 클러스터링

기존의 대부분 알고리즘들은 데이터의 모든 차원을 고려하였으나, CLIQUE[3]와 PROCLUS[4]는 알고리즘은 클러스터 형성에 연관성 있는 일부 차원만을 고려하는 알고리즘이다.

CLIQUE는 부분차원 클러스터링에 관한 첫 번째 연구로 의의가 있다. 즉 고차원 공간의 데이터 점들은 클러스터와 연관된 차원의 부분집합에 대하여 보다 효과적으로 클러스터링 할 수 있다는 것이다. CLIQUE는 밀도 및 격자 기반의 클러스터링 기법이며 좋은 확장성과 유용성을 가진다. 고차원 데이터에서 큰 밀도를 가진 영역을 찾는 효과적인 방안을 제시하는데 즉 각 차원은 일정한 간격(ξ)으로 나누고, 각 차원에서 이러한 간격의 교차-곱으로 이루어진 '단위(unit)'에 포함된 점들의 수가 기준밀도(τ)를 초과하면 밀집(dense)하다고 정의한다. 전체 k차원에서의 단위는 각 차원 간격들의 교차점이며, 클러스터는 k차원에서 연결된 밀집단위들의 집합이다. 이와 같이 CLIQUE는 프로젝션을 이용하여 고차원 데이터 공간에서 차원의 감소를 시도하여 밀도가 큰 영역을 찾는데 효과적인 방법을 제시했으나, 입력값으로 주어지는 일정하게 정해진 간격으로 인하여 클러스터의 정확한 형태를 요구하는 도메인에는 부적합하며, 탐색된 밀집영역 사이에는 큰 오버랩이 존재하는 문제점이 있다[4].

PROCLUS는 CLARANS[10]에 클러스터와 관련된 차원을 찾는 방법을 결합한 알고리즘이다. 즉 CLARANS 알고리즘과 같이 k개의 가능한 medoid를 생성한 후, 각 medoid와 관련이 높은 차원을 찾아내어 클러스터링을 한다. 이 알고리즘은 3가지 단계로 진행되는데, 첫 번째 초기화 단계에서는 CLARANS와 유사한 방법으로 알고리즘 수행 중 사용하게 될 medoid 샘플을 구하는 단계이다. 두 번째 반복 단계에서는 최상의 medoid 집합을 얻기 위하여 반복수행하며, 각 medoid에 대응하는 차원의 집합을 계산한다. 마지막으로 클러스터 정제 단계에서는 클러스터링의 질적인 향상을 위하여 데이터를 한번 더 검사하게 된다. 이와 같이 PROCLUS는 각 medoid에 대응하는 차원을 집합을 계산함으로써 차원을 감소시킬 수 있으나, 대용량의 데이터 집합의 경우 최상의 medoid를 탐색하는 시간을 예측하기 어려우며, 만약 적절한 medoid를 구하지 못했을 경우에는 데이터의 손실로 클러스터링 결과의 신뢰도를 저하시킬 가능성이 있는 단점이 있다.

3. 제안하는 클러스터링 알고리즘

본 논문에서는 고차원 데이터의 클러스터링을 위한 CLIP(CLustering based on Incremental Projection) 알고리즘을 제안한다. CLIP은 k-차원 데이터 공간에서 한 차원씩 점진적으로 프로젝션하면서 클러스터를 탐색한다. 즉 하나의 차원에서 시작하여 밀집영역을 구한 뒤, 그 차원의 밀집영역에 의존적인 그 다음 차원의 밀집영역을 찾아내는 방법으로 최종 k차원까지 반복해 나가는 것이다. 프로젝션하는 차원의 순서는 임의적일 수 있다. 즉 차원의 중요도에 따라 우선 순위를 부여하여 순서를 결정할 수 있다. CLIP 알고리즘을 설명하기 위한 데이터의 분포는 그림1과 같다고 가정한다. 단, 잡음(noise or outlier)에 해당하는 데이터 점들은 그림 1에 표시하지 않았다.

제안하는 CLIP 알고리즘은 크게 후보영역을 결정하기 위해 부분차원을 탐색하는 단계와 부분차원 내에서 클러스터를 식별하는 단계로 나누며 각각의 수행과정은 다음과 같다.

1) 클러스터를 포함하는 부분차원 탐색 (후보영역 결정)

수학적 의미에서 클러스터란, 전체 데이터 공간 R^k에서 k차원의 열린 연결 집합(open connected set)으로 정의할 수 있으며, 실제적인 데이터 집합에서 클러스터란 매우 조밀하게 분포된 데이터 집합을 포함하는 영역으로 정의할 수 있다. 클러스터를 포함하는 부분차원을 탐색하는데 있어서 중요하게 고려될 사항은 각 차원에서 밀집영역(dense region)을 찾는데 있다. 이와 같이 차원별 프로젝션을 통한 밀집영역의 조사는 다음과 같은 정리에 의거한다[3].

[정리] 만약 데이터 공간 R^k에 클러스터가 존재하면, 그 클러스터의 각 차원별 프로젝션은 밀집 영역을 포함한다. 여기서, 프로젝션 $P_l : R^k \rightarrow R$ 은 $P_l(x_1^{(1)}, x_2^{(2)}, \cdots, x_k^{(k)}) = x^{(l)}$ 로 정의한다.

위의 정리는 그 역이 항상 성립하지는 않는다. 즉, 모든 프로젝션 P_l ($l=1, \cdots, k$ 단, l :차원)이 밀집 영역을 포함하여도, 데이터 공간 R^k에 클러스터가 존재하지 않을 수도 있다. 제안하는 CLIP 알고리즘은 이러한 사실을 주목한다.

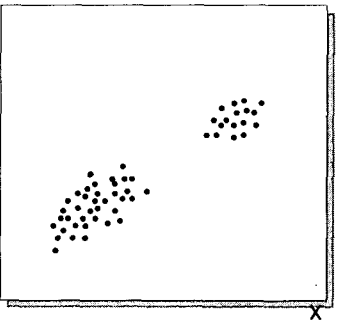

그림 1 데이터 공간

그림 1과 같이 데이터 공간의 점들은 데이터베이스에서 각각의 레코드를 의미한다. 이때 레코드의 각 애트리뷰트는 데이터 공간에서 하나의 차원을 이루게 되는데, 하나의 애트리뷰트에 해당하는 한 차원에 대하여 축-평행하게 프로젝션한다. 차원을 이루는 값들을 프로젝션한 후, 데이터의 분포를 계산하여 데이터의 밀도가 임계값 τ 이상인 밀집영역을 찾아야 한다. 그 다음 밀집영역에 해당하는 초월 사각형(hyper-rectangle) 부분에 존재하는 레코드에 대해서만, 그 다음 차원 즉 다음 애트리뷰트 값을 프로젝션시킨다. 클러스터들은 차원의 부분집합에서는 초월 사각형이다. 부분차원에서 이러한 초월 사각형 안의 데이터 점들은 평균밀도보다 매우 크다. 전체 k차원이라 할 때, 1차원에서 k차원에 이르기까지 순차적으로 각 차원에 해당하는 밀집영역을 그 다음 차원에 반영시킴으로서 최종적으로는 탐색할 데이터공간을 줄여 가는 것이다. 이 때, 각 차원에서 구해진 점들의 표준편차를 구하여 점들이 균등분포에 근사하면 밀집영역을 찾기 어려우므로 이와 같은 차원에 대해서는 일단 제외시킨다. 왜냐하면 비교적 균등하게 데이터가 분포하게 되면 클러스터를 탐색하는데 있어서는 저해요소가 되므로, 이와 같은 차원에 대해서는 후순위로 처리한다.

위 그림 1에서 클러스터에 대한 후보영역을 결정한다면 다음과 같다. 우선, X축으로 프로젝션하면 그림 2와 같이 2개의 초월 사각형 영역이 조사되는 것을 볼 수 있다. X축의 구간은 (X1~X2)와 (X3~X4)로 이루어진다. 다음으로, X축에 관하여 구해진 2개의 구간에 특정하게 Y축으로 프로젝션하면 Y축 구간은 그림 3의 (Y1~Y2)와 (Y3~Y4)로 결정된다. 이와 같이 각 차원에 대하여 점진적으로 프로젝션한 후의 결과가 되는 후보영역은 A와 B로 결정되며 이 영역의 명세는 각각 A=(X1≤X〈X2)∧(Y1≤Y〈Y2), B=(X3≤X〈X4)∧(Y3≤Y〈Y4)이다.

즉 A와 B 두 영역에 클러스터가 포함되어 있을 것이라 가정할 수 있다.

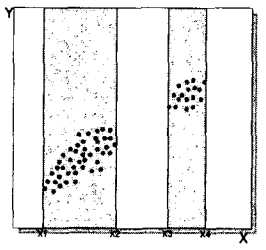

그림 2 X축 프로젝션 결과

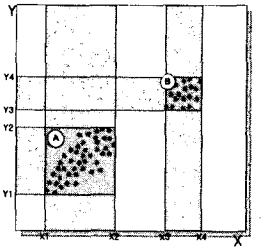

그림 3 Y축 프로젝션 결과

2) 클러스터 식별

이 단계는 앞에서 구해진 후보영역에서 클러스터의 형태를 식별하는 과정으로서, 점들의 중심부에서 탐색을 시작하기 위해 영역에 속한 데이터 점들의 대수적인 평균값을 다음과 같이 계산한다.

$$J(x^{(1)}, x^{(2)}, \cdots, x^{(k)}) = \sum_{i=1}^{n}\left[\sum_{j=1}^{k}(x^{(k)} - P_i^{(k)})^2\right]$$

(여기서, J : *functional*)

평균은 후보영역의 중심으로서 클러스터의 다른 모든 점으로부터 거리의 제곱의 합이 최소화되는 점이다.

즉 $J(x^{(1)}, x^{(2)}, \cdots, x^{(k)})$을 최소화하는 값으로서 $J(x^{(1)}, x^{(2)}, \cdots, x^{(k)}) = (\bar{x}^{(1)}, \bar{x}^{(2)}, \cdots, \bar{x}^{(k)})$을 구하는 것이다(여기서, $\bar{x}^{(l)} = \frac{1}{n}\sum_{i=1}^{n} P_i^{(l)}$ $(1 \le l \le k)$). 이와 같이 점들의 평균을 구한 다음 평균을 중심으로 2^k 개만큼 공간을 분할한다. 이것은 임의 차원에서 평균은 그 차원을 2^k개로 분할할 수 있기 때문이다. 예를 들어, 그림 4와 같이 2차원의 경우에는 2^2 = 4이므로 4개의 영역으로 데이터 공간이 분할된다. 그 다음으로 분할된 각 영역에서 그림 5와 같이 평균값을 구한다. 이는 각 분할 영역에서 평균을 중심으로 한 공간의 지역성(locality)을 고려하기 위한 것이다. 즉 데이터 점들은 그 평균을 중심으로 ε거리 내에 비교적 많이 분포한다는 것을 가정한다. 이와 같이 점들의 평균값을 중심으로 데이터를 조사하는 것은 클러스터의 형태(shape)를 효율적으로 예측할 수 있기 때문이다. 결과적으로 분할된 영역별로 평균값 즉 표준편차를 기준으로 ε거리 내의 점들을 조사하면서 클러스터의 형태를 구체화하는 것이다.

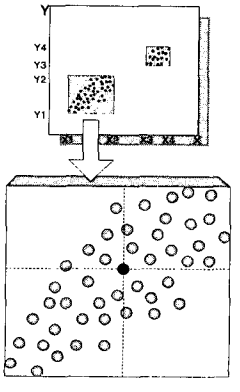

그림 4 후보영역의 중심 결정

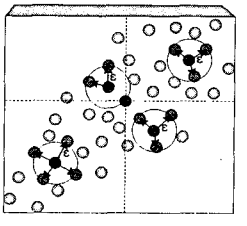

그림 5 분할영역의 중심 결정

4. 결론 및 향후 연구

본 논문에서는 고차원 데이터의 클러스터링을 위한 CLIP 알고리즘을 제안하였다. CLIP은 점진적인 프로젝션을 이용하여 클러스터가 존재할 후보영역을 제공함으로써 클러스터 탐색 공간을 크게 감소시킬 수 있으며, 데이터 점들의 평균값을 중심으로 즉 공간 지역성을 고려하여 점들을 탐색해 나감으로써 보다 효과적으로 클러스터 형태를 식별할 수 있다.

본 연구에서는 현재 CLIP 알고리즘의 효율성을 입증하기 위해 구현단계에 있으며 향후 다양한 실험을 통하여 기존의 방법들과 비교해야 할 것이다. 제안한 알고리즘에 대한 응용분야로서, 전자상거래에서 발생되는 방대한 양의 데이터를 클러스터링하여 판매전략, 수요예측, 생산계획 등에 적용할 수 있는 연구가 진행되고 있다.

참고문헌

[1] Fayyad, U. M., et al. *Advances in Knowlwdge Discovery and Data Mining*, (AAAI Press / The MIT Press), 1996.

[2] Chen, M. S., Han, J. and Yu, P.S., "Data Mining: An Overview from Database Perspective", *IEEE TKED, Vol.8, No.6* ,1996.

[3] Rakesh Agrawal, Johannes Gehrke, Dimitrios Gunopulos, and Prabhakar Raghavan, "Automatic subspace Clustering on High Dimensional Data Mining Applications," *Proc. of ACM SIGMOD Conference on Management of Data*, pp.94-105, 1998.

[4] Charu C. Agrawal, Ceilia Procopiuc, Joel L. Wolf, Philip S. Yu, and Jong Soo Prk, "Fast Algorithms for Projected Clustering," *Proc. of ACM SIGMOD Conference on Management of Data*, pp.61-72, 1999.

[5] Martin Ester, Hans-Peter Kriegel, Jorg Sander, and Xiaowei Xu, "A density-based algorithm for discovering clusters in large spatial database with noise," *Int'l Conference on Knowledge Discovery in Databases and Data Mining (KDD-96)*, 1996.

[6] Mihael Ankerst, Markus M. Breunig, Han-Peter Kriegel, and Jorg Sander, "OPTICS: Ordering points to identify the clustering structure," *the ACM SIGMOD Conference on Management of Data*, 1999.

[7] Richard O. Duda and Peter E. Hard, "Pattern Classification and Scene Analysis," A Wiley-Interscience Publication, New York, 1973.

[8] Martin Ester, Hans-Peter Kriegel, Jorg Sander, Michael Wimmer, and Xiaowei Xu, "Incremental Clustering for Mining in a Data Warehousing Environment," *the VLDB Conference*, 1998.

[9] Martin Ester, Hans-Peter Kriegel, Jorg Sander, and Xiaowei Xu, "Density-Connected Sets and their Application for Trend Detection in Spatial Databases," *Int'l Conference on Knowledge Discovery in Databases and Data Mining (KDD-97)*, 1997.

[10] R. Ng, J. Han, "Efficient and Effective Clustering Methods for Spatial Data Mining," *Proc. of the 20th VLDB Conference*, 1994.

[11] S. Guha, R. Rastogi and K. Shim, "CURE: An Efficient Clustering Algorithm for Large Databases," *Proc. of ACM SIGMOD Conference on Management of Data*, pp.73-84, 1998.

[12] Tian Zhang, Raghu Ramakrishnan, and Miron Livny, "BIRCH: An Efficient data clustering method for very large databases," *Proc. of ACM SIGMOD Conference on Management of Data*, pp.103-114, 1996.