TRƯỜNG ĐẠI HỌC VINH
KHOA CÔNG NGHỆ THÔNG TIN

BÀI GIẢNG MÔN HỌC

KHAI PHÁ DỮ LIỆU (Data Mining)
Chương 5: Phân cụm dữ liệu (Clustering)

Giảng viên: TS. Cao Thanh Sơn
Bộ môn các hệ thống thông tin
Email: ctsdhv@gmail.com

2017

Chương 5: Phân cụm (Clustering)

*Based on slides **Data Mining: Concepts and Techniques***
by
Jiawei Han, Micheline Kamber, and Jian Pei, 2011

*and slides **Introduction to Data Mining***
by
Tan, Steinbach, Kumar, 2005

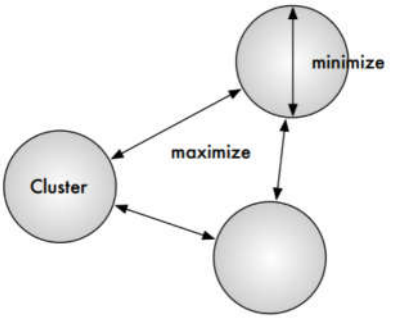
Some illustrative images are downloaded from the Internet.

Nội dung

- Phân cụm dữ liệu
- Phương pháp chia cắt
  - Thuật toán *k*-means
- Phân cụm dựa trên phân cấp
- Phân cụm dựa trên mật độ

Phân cụm dữ liệu

- Bài toán
  - $D = \{A_1, A_2, \ldots, A_m\}$: tập dữ liệu
  - Phân dữ liệu thuộc D thành các cụm sao cho
    - Các đối tượng trong cùng cụm có mức tương đồng cao
    - Các đối tượng khác cụm có mức tương đồng thấp
  - Tìm sự tương tự giữa các đối tượng dựa trên đặc điểm hay thuộc tính của chúng và tìm cách nhóm các đối tượng giống nhau vào cùng 1 cụm (nhóm)
  - Unsuppervised learning: tìm cách phân cụm các đối tượng chưa được phân cụm



Phân cụm dữ liệu

- Một số ví dụ ứng dụng của bài toán phân cụm
  - **Sinh học**: phân loại động vật và thực vật dựa vào thuộc tính của chúng
  - **Tìm kiếm thông tin**: phân nhóm văn bản, tài liệu
  - **Địa lý**: phát hiện các vùng địa lý tương tự nhau
  - **Marketing**: phân các nhóm khách hàng có sở thích mua sắm giống nhau
  - **Social networks**: phân nhóm người dùng cùng sở thích
  - **Library**: theo dõi độc giả, sách, dự đoán nhu cầu của độc giả
  - **Web mining**: phân loại tài liệu, phân loại người dùng web

Phân cụm dữ liệu

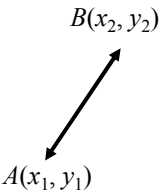
- Các bước cơ bản để phân cụm dữ liệu
  - Lựa chọn thuộc tính
    - chọn thuộc tính liên quan đến vấn đề quan tâm
    - cần tối thiểu hoá dư thừa thông tin
  - Độ đo tương tự/khác biệt
    - đánh giá sự giống nhau của hai đối tượng
  - Xây dựng thuật toán
  - Đánh giá và phân tích ý nghĩa của kết quả

Mức độ tương đồng – Similarity

- Mức độ tương đồng/khác biệt giữa 2 đối tượng o_1, o_2 thường được cụ thể hoá bằng một hàm tính khoảng cách *distance* (*dist*) dựa trên các thuộc tính của đối tượng
  - small distance = similar object
  - large distances = dissimilar objects
- Các thuộc tính của hàm tính khoảng cách *dist*
  - *dist*(o_1, o_2): khoảng cách giữa 2 điểm o_1 và o_2
  - *dist*(o_1, o_2) = $d \in \mathbb{R}^{\geq 0}$ (không âm)
  - *dist*(o_1, o_2) = 0 iff $o_1 = o_2$
  - *dist*(o_1, o_2) = *dist*(o_2, o_1) (đối xứng)

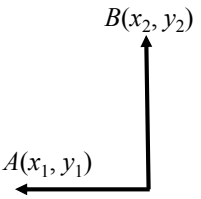
- Chất lượng của phân cụm dữ liệu thường phụ thuộc nhiều vào hàm tính khoảng cách
- Có nhiều cách tính khoảng cách
  - Khoảng cách Euclidean:

$$dist(A,B) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$



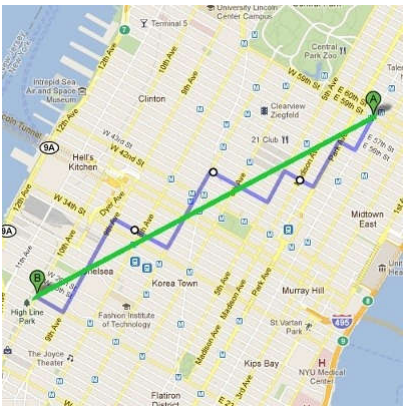
  - Khoảng cách Manhattan:

$$dist(A,B) = |x_2 - x_1| + |y_2 - y_1|$$



  - ...

❑ Manhattan vs Euclidean distance



❑ Các phương pháp tiếp cận chính

- Chia cắt (Partitioning)
  - Thực hiện chia cắt tập dữ liệu thành các tập con và đánh giá chất lượng dựa trên hàm mục tiêu
  - Các thuật toán cơ bản: *k*-means, *k*-medoids,…
- Phân cấp (Hierachical)
  - Tạo ra cây phân cấp nhóm các đối tượng bằng cách sử dụng các hàm đánh giá khoảng cách hợp lý
  - Các thuật toán cơ bản: Agnes, Diana,…
- Dựa trên mật độ (Density-based)
  - Dựa trên sự kết nối của các đối tượng và hàm đánh giá mật độ
  - Các thuật toán cơ bản: DBSCAN, OPTICS,…
- …

Phương pháp chia cắt – Partitioning

❑ Khái niệm cơ bản

- Tạo ra một phân hoạch trên tập đối tượng D thành k cụm
  $C = \{C_1, C_2, \ldots, C_k\}$
- Xét $C_i = \{x_1, x_2, \ldots, x_m\}$ (m đối tượng trong 1 cụm)
- Mỗi đối tượng có n thuộc tính $x_j = (A_{j1}, A_{j2}, \ldots, A_{jn})$, $1 \le j \le m$
- Trọng tâm cụm (centroid) là đối tượng c_i được xác định, $1 \le j \le m$

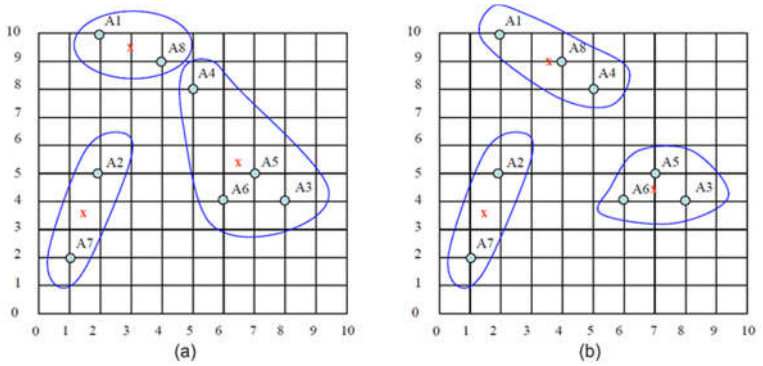
$$c_i = \left(\frac{1}{m}\sum_{i=1}^{m} A_{j1}, \frac{1}{m}\sum_{i=1}^{m} A_{j2}, \ldots, \frac{1}{m}\sum_{i=1}^{m} A_{jn}\right)$$

- Ví dụ:
  - Cho cụm $C_1 = \{x_1, x_2, x_3\}$
  - $x_1 = (1, 2, 1)$, $x_2 = (1, 3, 2)$, $x_3 = (1, 1, 3)$
  - Trọng tâm cụm c_1 được tính
  - $c_1 = \left(\frac{1+1+1}{3}, \frac{2+3+1}{3}, \frac{1+2+3}{3}\right) = (1, 2, 2)$

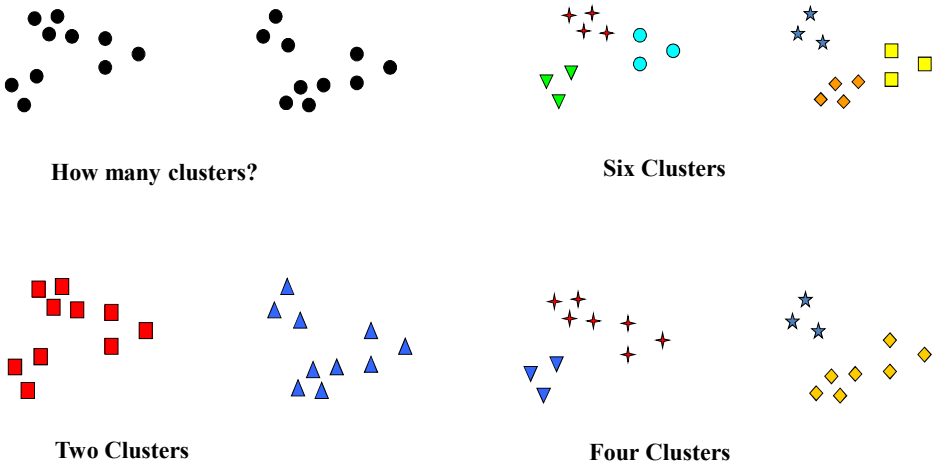
Phương pháp chia cắt – Partitioning

❑ Thuật toán điển hình

- *k*-means: mỗi cụm được đại diện bởi trọng tâm cụm (*centroid*) là vector trung bình tính từ các đối tượng trong nhóm



Phương pháp chia cắt – Partitioning



Thuật toán *k*-means [1]

Algorithm: *k*-means. The *k*-means algorithm for partitioning, where each cluster's center is represented by the mean value of the objects in the cluster.

Input:

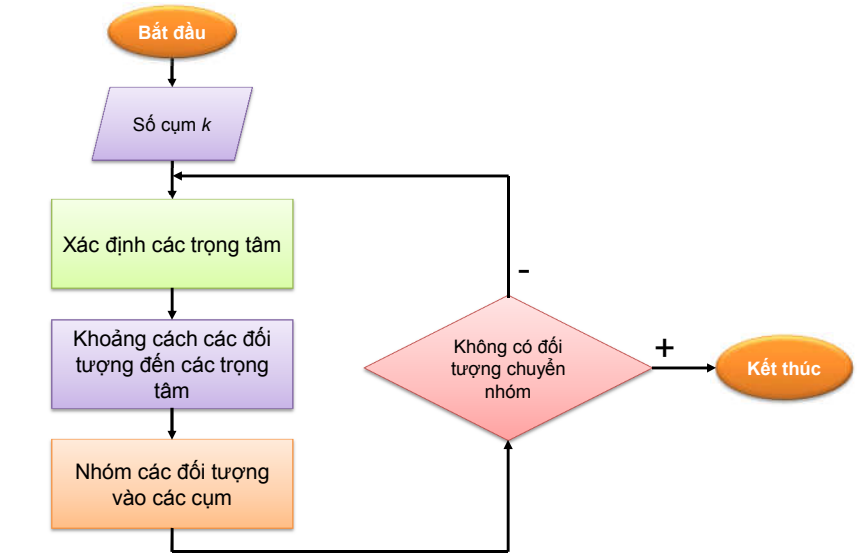
- *k*: the number of clusters,
- *D*: a data set containing *n* objects.

Output: A set of *k* clusters.

Method:

(1) arbitrarily choose *k* objects from *D* as the initial cluster centers;
(2) **repeat**
(3) (re)assign each object to the cluster to which the object is the most similar, based on the mean value of the objects in the cluster;
(4) update the cluster means, that is, calculate the mean value of the objects for each cluster;
(5) **until** no change;

Thuật toán *k*-means [1]



Thuật toán *k*-means [1]

- Thuật toán *k*-means thực hiện qua các bước sau
  1. chọn (ngẫu nhiên) *k* điểm trọng tâm (centroid) cho *k* cụm
  2. tính khoảng cách giữa các đối tượng đến điểm trọng tâm
     - khoảng cách Euclidean
     - khoảng cách Manhattan
     - …
  3. nhóm các đối tượng vào cụm gần nhất
  4. xác định lại điểm tâm mới cho các cụm
  5. thực hiện bước 2 cho đến khi không có sự thay đổi nhóm nào của các đối tượng

Thuật toán *k*-means [1]

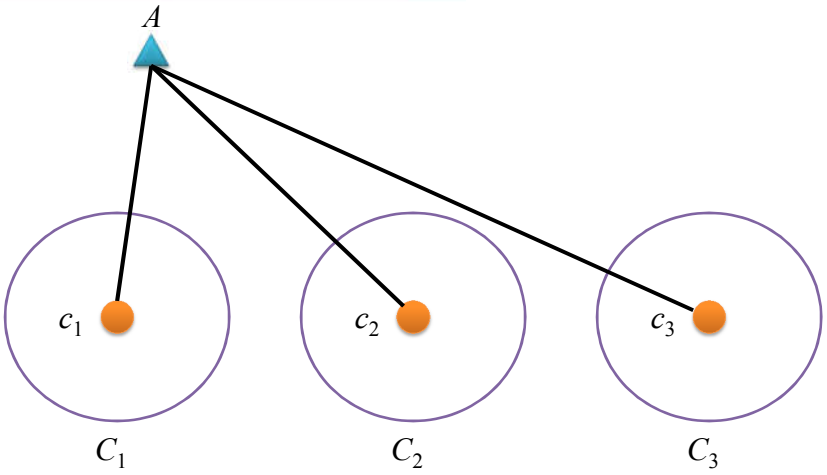
- Phương pháp phân hoạch
  - *D*: CSDL chứa *n* đối tượng
  - Phân hoạch *D* thành *k* cụm $\{C_1, C_2, \ldots, C_k\}$, sao cho tổng bình phương khoảng cách của mỗi đối tượng dữ liệu tới trọng tâm cụm chứa nó đạt giá trị cực tiểu. Giá trị được tính như sau, trong đó c_i là trọng tâm cụm (centroid, medoid)

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} dist(\boldsymbol{p}, \boldsymbol{c_i})^2,$$

Thuật toán *k*-means [1]

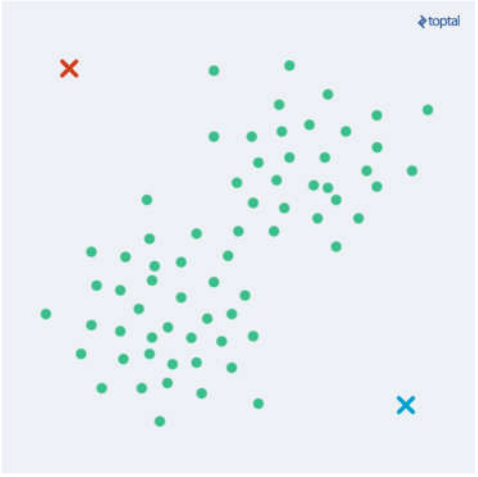
- Điều kiện dừng
  - Giải thuật hội tụ
    - không còn sự phân chia lại giữa các cụm
    - trọng tâm các cụm không thay đổi
    - lúc này tổng khoảng cách từ các đối tượng thuộc cụm đến trọng tâm cụm là cực tiểu
  - Giải thuật không hội tụ: trọng tâm các cụm thay đổi liên tục, ta dừng giải thuật khi
    - số lượng vòng lặp vượt quá ngưỡng xác định trước
    - giá trị *E* nhỏ hơn một ngưỡng nào đó xác định trước
    - hiệu giá trị của *E* trong 2 vòng lặp liên tiếp nhỏ hơn ngưỡng nào đó xác định trước $|E_{n+1} - E_n| < \varepsilon$

Thuật toán *k*-means [1]



- $dist(A, c_1) < dist(A, c_2) < dist(A, c_3) \rightarrow A \in C_1$

Thuật toán *k*-means [1]



Mô phỏng quá trình phân cụm K-Means

Nguồn: https://techmaster.vn/posts/33893/thuat-toan-phan-cum

Thuật toán *k*-means [1]

- Ưu điểm
  - Đơn giản
  - Dễ cài đặt
  - Tương đối hiệu quả
  - Các đối tượng tự động gán vào các nhóm
  - Thường đạt tối ưu cục bộ

Thuật toán *k*-means [1]

- Nhược điểm
  - Thuộc tính phi số? (biến đổi để có độ đo phù hợp)
  - *k*-means không đem lại kết quả tốt nếu các nhóm có kích thước, mật độ khác nhau
  - Cần phải xác định số nhóm (*k*) trước khi thực hiện
  - Tất cả các đối tượng đều phải gán vào các nhóm
  - Phụ thuộc vào việc chọn nhóm đầu tiên
  - Bị ảnh hưởng bởi đối tượng ngoại lai

Thuật toán *k*-means [1]

- Ví dụ áp dụng
  - cho tập dữ liệu D như sau
  - phân cụm tập dữ liệu D với $k = 2$ (áp dụng khoảng cách Manhattan)
  - $C = \{C_1, C_2\}$
- Chọn centroid
  - $c_1 = x_1 = (1, 1) \in C_1$
  - $c_2 = x_3 = (4, 3) \in C_2$

	A_1	A_2
x_1	1	1
x_2	2	1
x_3	4	3
x_4	5	4

Thuật toán *k*-means [1]

- Lần lặp 1
  - $x_2 = (2, 1)$
    - $dist(x_2, c_1) = |1-2|+|1-1| = 1$
    - $dist(x_2, c_2) = |4-2|+|3-1| = 4$
    - $\Rightarrow x_2 \in C_1$
  - $x_4 = (5, 4)$
    - $dist(x_4, c_1) = |1-5| + |1-4| = 7$
    - $dist(x_4, c_2) = |4-5| + |3-4| = 2$
    - $\Rightarrow x_4 \in C_2$
  - Ta thu được 2 cụm:
    - $C_1 = \{x_1, x_2\}$
    - $C_2 = \{x_3, x_4\}$
  - Cập nhật lại trọng tâm cụm:
    - $c_1 = \left(\frac{1+2}{2}, \frac{1+1}{2}\right) = (1.5, 1)$
    - $c_2 = \left(\frac{4+5}{2}, \frac{3+4}{2}\right) = (4.5, 3.5)$

	A_1	A_2
x_1	1	1
x_2	2	1
x_3	4	3
x_4	5	4

Thuật toán *k*-means [1]

- Lần lặp 2 (c_1 = (1.5, 1); c_2 = (4.5, 3.5))
  - x_1 = (1, 1)
    - $dist(x_1, c_1)$ = |1.5-1|+|1-1| = 0.5
    - $dist(x_1, c_2)$ = |4.5-1|+|3.5-1| = 6

    $\Rightarrow x_1 \in C_1$
  - x_2 = (2, 1)
    - $dist(x_2, c_1)$ = |1.5-2| + |1-1| = 0.5
    - $dist(x_2, c_2)$ = |4.5-2| + |3.5-1| = 5

    $\Rightarrow x_2 \in C_1$
  - x_3 = (4, 3)
    - $dist(x_3, c_1)$ = |1.5-4| + |1-3| = 4.5
    - $dist(x_3, c_2)$ = |4.5-4| + |3.5-3| = 1

    $\Rightarrow x_3 \in C_2$
  - x_4 = (5, 4)
    - $dist(x_4, c_1)$ = |1.5-5| + |1-4| = 6.5
    - $dist(x_4, c_2)$ = |4.5-5| + |3.5-4| = 1

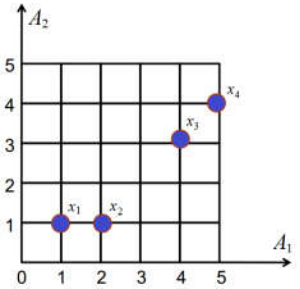
    $\Rightarrow x_4 \in C_2$

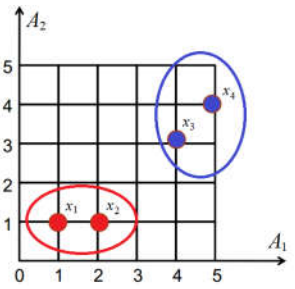
	A_1	A_2
x_1	1	1
x_2	2	1
x_3	4	3
x_4	5	4

Thuật toán *k*-means [1]

- Sau bước lặp này ta có 2 cụm
  - $C_1 = \{x_1, x_2\}$
  - $C_2 = \{x_3, x_4\}$
- Nhận xét:
  - Kết quả phân cụm giữ nguyên sau 2 lần lặp, giải thuật dừng và cho kết quả phân cụm:
    - $C_1 = \{x_1, x_2\}$
    - $C_2 = \{x_3, x_4\}$

	A_1	A_2
x_1	1	1
x_2	2	1
x_3	4	3
x_4	5	4





Thuật toán *k*-means [1]

- Bài tập tại lớp
  - Làm lại bài tập trên áp dụng khoảng cách Euclidean

Phân cụm dữ liệu

- Phân cụm dựa trên phân cấp (Hierarchical clusterings)
  - Tự đọc các tài liệu [1, 2, 3]
- Phân cụm dựa nhóm mật độ (Density-based clusterings)
  - Tự đọc các tài liệu [1, 2, 3]

Tài liệu tham khảo

[1] Jiawei Han, Micheline Kamber, Jian Pei, *Data Mining: Concepts and Techniques*, 3rd ed, Morgan-Kaufmann Publishers, 2012.

[2] Nguyễn Hà Nam, Nguyễn Trí Thành, Hà Quang Thuỵ, *Giáo trình khai phá dữ liệu*, NXB Đại học Quốc gia Hà Nội, 2013.

[3] Pang-Ning Tan, Michael Steinbach, Vipin Kumar, *Introduction to Data Mining*, Addison-Wesley, 2005

[4] WEKA, *www.cs.waikato.ac.nz/ml/weka/*