**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**
**KHOA HỆ THỐNG THÔNG TIN**


# BÀI TẬP LỚN MÔN HỌC KHAI PHÁ DỮ LIỆU
# Đề tài: PHÂN CỤM DỮ LIỆU VỚI THUẬT TOÁN K - MEANS

Giáo viên hướng dẫn:

**TS. Nguyễn Đình Thuân**

Nhóm thực hiện:

1. **Đặng Thị Ngọc Loan 08520501**
2. **Lê Nguyễn Hào Hiệp 08520631**
3. **Nguyễn Thị Mỹ Dung 08520642**



**Tp. Hồ Chí Minh, tháng 11/2011**

## Mục lục

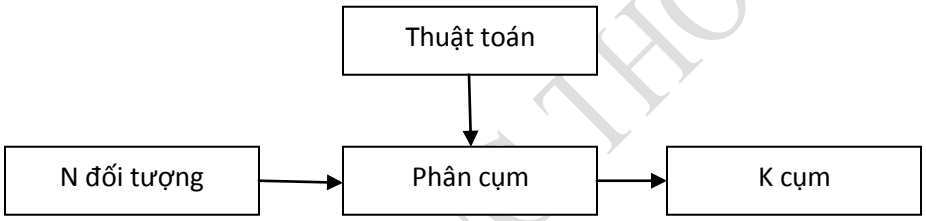
# 1 Phân cụm dữ liệu

## 1.1 Định nghĩa

Phân cụm dữ liệu là một kỹ thuật khai phá dữ liệu phổ biến nhất, phân cụm dữ liệu có liên quan đến nhiều ngành khoa học khác: CSDL, thống kê, học máy, … Phân cụm dữ liệu được ứng dụng trong rất nhiều lĩnh vực: nhận dạng, phân lớp, tài chính, ngân hàng, Internet, …

Phân cụm dữ liệu thuộc nhóm các kỹ thuật khai phá dữ liệu mô tả, thuộc loại học không giám sát (unsupervised learning) trong ngày học máy tính.

Phân cụm dữ liệu là kỹ thuật sử dụng quan sát đối tượng, mục đích để tổ chức một tập các đối tượng cụ thể hoặc trừu tượng vào các nhóm, cụm phân biệt. Những tài liệu có nội dung tương tự nhau sẽ được xếp vào cùng một cụm và những tài liệu có nội dung khác nhau được xếp vào cụm khác nhau.




Hình 1: Quy trình phân cụm

### 1.1.1 Các kiểu biểu diễn dữ liệu

Dựa trên kích thước ta có thể phân dữ liệu thành hai loại là thuộc tính liên tục và thuộc tính rời rạc. Bên cạnh đó, nếu phân loại dựa trên hệ đo thì có một số dữ liệu thông dụng như thuộc tính định danh, thuộc tính có thứ tự, thuộc tính khoảng, thuộc tính tỉ lệ. Các đơn vị đo có ảnh hưởng trực tiếp đến kết quả phân cụm. Vì thế người ta phải chuẩn hóa dữ liệu để khắc phục yếu điểm này.

- Biểu diễn dưới dạng ma trận của các biến cấu trúc hay các thuộc tính của đối tượng.

  Ví dụ: Đối tượng người có các thuộc tính là tên, tuổi, chiều cao, cân nặng, màu mắt, … Nếu ta có n đối tượng, mỗi đối tượng có p thuộc tính thì sẽ có một ma trận với n dòng, p cột

$$\begin{bmatrix} x_{11} & \dots & x_{1j} & \dots & x_{1p} \\ \dots & \dots & \dots & \dots & \dots \\ x_{i1} & \dots & x_{ij} & \dots & x_{ip} \\ \dots & \dots & \dots & \dots & \dots \\ x_{n1} & \dots & x_{nj} & \dots & x_{np} \end{bmatrix}$$

Hình 2: Ma trận dữ liệu (data matrix)

- Biểu diễn dữ liệu dưới dạng độ do khoảng cách giữa đôi một các cặp đối tượng. Nếu ta có n đối tượng, chúng ta sẽ được biểu diễn bằng một ma trận với n hàng và n cột như sau

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ . & . & & & \\ . & . & & d(i,j) & \\ d(n,1) & d(n,2) & & & 0 \end{bmatrix}$$

Hình 3: Ma trận sai biệt (dissimilary matrix)

Trong đó, d(i,j) là độ đo khoảng cách giữa hai đối tượng i và j; thể hiện sự khác biệt giữa đối tượng i và j; được tính tùy thuộc vào kiểu của các biến/ thuộc tính.

d(i,j) gần bằng 0 khi hai đối tượng i và j là gần nhau hay có nội dung gần giống nhau, và d càng tăng khi các đối tượng có nội dung khác nhau.

Hình 3 biểu diễn ma trận khoảng cách của tập dữ liệu có

d(i,j) = d(j,i)

d(i,i) = 0

$d(i,j) \geq 0$

$d(i,j) \leq d(i,k) + d(k,j)$

- Đối tượng vector (vector objects)

  Đối tượng i và j được biểu diễn tương ứng bởi vector x và y.

  Độ tương tự giữa i và j được tính bởi độ đo cosine:

$$s(\boldsymbol{x}, \boldsymbol{y}) = \frac{\boldsymbol{x}^t \cdot \boldsymbol{y}}{||\boldsymbol{x}|| ||\boldsymbol{y}||}$$

  Trong đó: $x = (x_1,..., x_p)$

  $y = (y_1,\ldots, y_p)$

  $s(x, y) = (x_1*y_1 + \ldots + x_p*y_p)/((x_1^2 + \ldots + x_p^2)^{1/2}*(y_1^2+ \ldots + y_p^2)^{1/2})$

- Kiểu dữ liệu thuộc tính
  - Kiểu định danh/ chuỗi (norminal): không có thứ tự

    Lấy các giá trị từ một tập không có thứ tự các giá trị (định danh)

    Ví dụ: Các thuộc tính như: Tên, Nghề Nghiệp,…
  - Kiểu nhị phân (binary): là một trường hợp đặc biệt của kiểu định danh

    Tập các giá trị chỉ gồm 2 giá trị (Y/N, 0/1, T/F)

- Kiểu có thứ tự (ordinal):
  Lấy giá trị từ một tập có thứ tự các giá trị
  Ví dụ 1: Các thuộc tính lấy giá trị như: tuổi, chiều cao,…
  Ví dụ 2: Thuộc tính Thu nhập lấy giá trị từ tập {thấp, trung bình, cao}
- Kiểu thuộc tính rời rạc (Discrete – valued attributes)
  Tập các giá trị là một tập hữu hạn
  Bao gồm cả các thuộc tính có kiểu giá trị là các số nguyên
  Bao gồm cả các thuộc tính nhị phân (binary attributes)
- Kiểu thuộc tính liên tục (Continuos – valued attributes)
  Các giá trị là các số thực (real numbers)

### 1.1.2 Một số độ đo trong phân cụm

#### 1.1.2.1 Phép đo khoảng cách cho dữ liệu thuộc tính khoảng

Khoảng cách giữa hai đối tượng i, j hay độ đo phí tưởng tượng giữa hai đối tượng được xác định bằng một ma trận. Một số phương pháp đo khoảng cách phổ biến là: khoảng cách Euclidean, khoảng cách Manhattan, khoảng cách Chebychev, … được định nghĩa bằng:

- Khoảng cách Minkowski:

$$d(i,j)=\sqrt[q]{(|x_{i_1}-x_{j_1}|^q+|x_{i_2}-x_{j_2}|^q+...+|x_{i_p}-x_{j_p}|^q)}$$

- Khoảng cách Manhattan

$$d(i,j)=|x_{i_1}-x_{j_1}|+|x_{i_2}-x_{j_2}|+...+|x_{i_p}-x_{j_p}|$$

- Khoảng cách Enclidean

$$d(i,j)=\sqrt{(|x_{i_1}-x_{j_1}|^2+|x_{i_2}-x_{j_2}|^2+...+|x_{i_p}-x_{j_p}|^2)}$$

#### 1.1.2.2 Phép đo khoảng cách cho dữ liệu thuộc tính nhị phân

Xác định một bảng ma trận thuộc tính nhị phân

| | j=1 | j=0 | Sum |
|---|---|---|---|
| i=1 | a | b | a+b |
| i=0 | c | d | c+d |
| Sum | a+c | b+d | p (= a+b+c+d) |

Trong đó

a: tổng số thuộc tính là 1 trong 2 mẫu *(xi,xj)* được đo

b: Tổng số thuộc tính là 1 trong *xi* và 0 trong *xj*

c: Tổng số thuộc tính là 0 trong *xj* và 1 trong *xi*

d: tổng số thuộc tính là 0 trong 2 mẫu *(xi,xj)* được đo

Hệ số đối sánh đơn giản (nếu đối xứng (symmetric)): $d(i,j)=\frac{b+c}{a+b+c+d}$

Hệ số đối sánh Jaccard (nếu không đối xứng asysmmetric): $d(i,j)=\frac{b+c}{a+b+c}$

Ví dụ:

| Name | Gender | Fever | Cough | Test-1 | Test-2 | Test-3 | Test-4 |
|---|---|---|---|---|---|---|---|
| Jack | M | Y | N | P | N | N | N |
| Mary | F | Y | N | P | N | P | N |
| Jim | M | Y | P | N | N | N | N |

- Gender: thuộc tính đối xứng có 2 giá trị M→1,F→0 (symmetric)
- Các thuộc tính còn lại bất đối xứng Y, P → 1, N → 0

$$d(jack,mary)=\frac{0+1}{2+0+1}=0.33$$

$$d(jack,jim)=\frac{1+1}{1+1+1}=0.67$$

$$d(jim,mary)=\frac{1+2}{1+1+2}=0.75$$

### 1.1.2.3 Phép đo khoảng cách cho dữ liệu thuộc tính định danh

Độ đo khoảng cách giữa hai đối tượng i và j được định nghĩa bằng hàm

$$d(i,j)=\frac{p-m}{p}$$

Trong đó:

m: tổng thuộc tính có giá tương ứng trùng nhau

p: tổng số các thuộc tính

## 1.2 Đặc điểm phân cụm dữ liệu

### 1.2.1 Mục đích của việc phân cụm

Bài toán phân cụm có mục đích tìm kiếm các tài liệu và phân chúng vào các cụm khác nhau. Tuy nhiên, tùy thuộc vào mục đích của người dùng mà người lập trình sẽ quyết định số lượng cụm, hay chất lượng cụm ở mức nào. Một cách phân chia dữ liệu với số lượng cụm linh hoạt được thực hiện bằng cách cắt cây ở mục phù hợp ví dụ như sử dụng thuật toán phân cụm cây phân cấp.

### 1.2.2 Các yêu cầu tiêu biểu

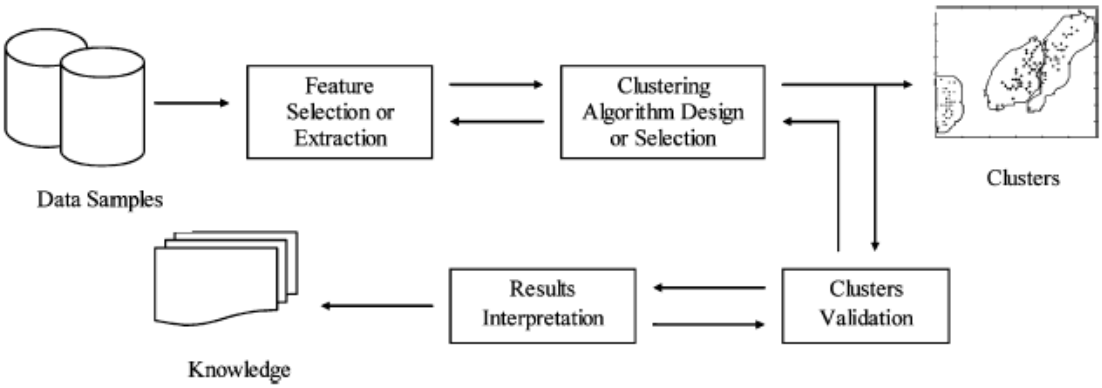
Khả năng co giãn về tập dữ liệu (scalability)
Khả năng xử lý nhiều kiểu thuộc tính khác nhau
Khả năng khám phá các cụm với hình dạng tùy ý
Tối thiểu hóa yêu cầu về tri thức miền trong việc xác định các thông tin số nhập
Khả năng xử lý dữ liệu có nhiễu
Khả năng gom cụm tăng dần và độc lập với thứ tự của dữ liệu nhập
Khả năng xử lý dữ liệu đa chiều
Khả năng gom cụm dựa trên ràng buộc
Khả diễn và khả dụng

### 1.2.3 Các bước cơ bản trong phân cụm

Bài toán phân cụm nói chung dựa theo các bước cơ bản sau:

- Chọn lựa đặc trưng: Các đặc trưng phải được chọn lựa một cách thích hợp để có thể mã hóa nhiều nhất thông tin liên quan đến công việc quan tâm. Mục tiêu chính là giảm thiểu sự dư thừa thông tin giữa các đặc trưng. Các đặc trưng cần được tiền xử lý trước khi thực hiện các bước tiếp theo.
- Chọn độ đo tương tự: Là độ đo chỉ ra mức tương tự hay khoảng cách giữa các vector đặc trưng. Phải đảm bảo các vector đặc trưng góp phần như nhau trong việc tính toán độ đo tương tự và không có đặc trưng nào lấn át đặc trưng nào.
- Tiêu chuẩn phân cụm: Tiêu chuẩn phân cụm có thể được biểu diễn bởi hàm chi phí hay một vài quy tắc khác. Nó cũng phụ thuộc vào người lập trình.
- Thuật toán phân loại: Lựa chọn một sơ đồ thuật toán riêng biệt nhằm sáng tỏ cấu trúc phân cụm của tập dữ liệu.
- Công nhận kết quả: Khi có kết quả phân loại, cần kiểm tra tính đúng đắn của nó bằng cách đánh giá độ chính xác.

- Giải thích kết quả: Trong nhiều trường hợp, chuyên gia trong lĩnh vực ứng dụng phải kết hợp kết quả phân loại với những bằng chứng thực nghiệm và phân tích để đưa ra các kết luận đúng đắn.




Hình 4: Quá trình Phân cụm dữ liệu

Các lựa chọn khác nhau của các đặc trưng, độ đo tương tự, tiêu chuẩn phân cụm có thể dẫn đến các kết quả phân cụm khác nhau.

### 1.2.4 Các phương pháp làm sạch dữ liệu

#### 1.2.4.1 Xử lý dữ liệu bị thiếu

- Định nghĩa của dữ liệu bị thiếu: Dữ liệu không có sẵn khi cần được sử dụng
- Nguyên nhân gây ra dữ liệu bị thiếu: Khách quan (không tồn tại lúc được nhập liệu, sự cố, …) và chủ quan (tác nhân con người)
- Giải pháp cho dữ liệu bị thiếu
  - Bỏ qua các bộ
  - Xử lý tay (không tự động, bán tự động)
  - Dùng giá trị thay thế (tự động): các giá trị quy ước, các thuộc tính có nghĩa, các giá trị của các bộ cùng thể loại, giá trị có tỉ lệ xuất hiện cao.
  - Ngăn chặn dữ liệu bị thiếu: thiết kế tốt CSDL và các thủ tục nhập liệu (các ràng buộc dữ liệu)

#### 1.2.4.2 Nhận diện phần tử biên (outliers) và giảm thiểu nhiễu (noisy data)

- Định nghĩa:

Outliers: Những dữ liệu (đối tượng) không tuân theo đặc tính/hành vi chung của tập dữ liệu (đối tượng).

Noisy data: là một lỗi ngẫu nhiên hay do biến động của các biến trong quá trình thực hiện hoặc ghi chép nhầm lẫn không được kiểm soát …

- Nguyên nhân
  - Khách quan (công cụ thu thập dữ liệu, lỗi trên đường truyền, giới hạn công nghệ, …)
  - Chủ quan (tác nhân con người)
- Giải pháp nhận diện phần tử biên
  - Dựa trên phân bố thống kê
  - Dựa trên khoảng cách
  - Dựa trên mật độ
  - Dựa trên độ lệch
- Giải pháp giảm thiểu nhiễu
  - Binning: làm mịn một giá trị dữ liệu được xác định thông qua các giá trị xung quanh nó.

    Khi làm mịn trung vị trong mỗi bin, các giá trị sẽ được thay thế bằng giá trị trung bình các giá trị có trong bin.

    Làm mịn biên: các giá trị nhỏ nhất và lớn nhất được xác định và dùng làm ranh giới của bin. Các giá trị còn lại của bin sẽ được thay thế bằng một trong hai giá trị trên tùy thuộc và độ lệch giữa giá trị ban đầu với các giá trị biên đó.

    Ví dụ về phương pháp làm mịn Binning

Mảng lưu giá các mặt hàng: 4, 8, 15, 21, 21, 24, 25, 28, 34
Phân thành các bin
Bin 1: 4, 8 , 15
Bin 2: 21, 21, 24
Bin 3: 25, 28, 34

Làm mịn sử dụng phương pháp trung vị
Bin 1: 9, 9 ,9
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

Làm mịn biên
Bin 1: 4, 4, 15
Bin 2: 21, 21, 24
Bin 3: 25, 25, 34

- Hồi quy: Phương pháp thường dùng là hồi quy tuyến tính, để tìm ra được mối quan hệ tốt nhất giữa hai thuộc tính (hoặc các biến), từ đó một thuộc tính có thể dùng để dự đoán thuộc tính khác. Hồi quy tuyến tính đa điểm là một sự mở rộng của phương pháp trên, trong đó có nhiều hơn hai thuộc tính được xem xét, và các dữ liệu tính ra thuộc về một miền đa chiều.
- Phân tích cụm: Các giá trị tương tự nhau được tổ chức thành các nhóm hay "cụm" trực quan. Các giá trị rơi ra bên ngoài các nhóm này sẽ được xem xét để làm mịn.

## 1.3 Một số phương pháp phân cụm

### 1.3.1 Phân cụm phân hoạch

Các thuật toán phân cụm phân hoạch: K - means, PAM, CLARA, CLARANS,...
Đặc điểm chính của các thuật toán phân cụm phân hoạch:

Đưa ra một tham số k, tìm cách phân chia tập dữ liệu thành k cụm dựa theo việc lựa chọn các tiêu chuẩn phân chia.

K – means là thuật toán cơ sở đơn giản nhất để phân cụm dữ liệu, được đưa ra lần đầu bởi (J.B.MacQueen, 1967) và cho đến nay đã có nhiều dạng biến đổi khác nhau. Trong K – means, cụm được định nghĩa bởi trung tâm của các phần tử.

K – means chỉ áp dụng với các tập dữ liệu có thuộc tính số và rất phù hợp với các cụm có dạng hình cầu. K – means nhạy cảm với dữ liệu "nhiễu" hoặc "ngoại lai", độ phức tạp là O (lkn) (l là số đối tượng, k là số cụm, n là số lần lặp, k<<l, n<<l)

PAM (Kaufman and Rousseeuw, 1987) là thuật toán mở rộng của K – means. PAM có khả năng xử lý tốt đối với dữ liệu "nhiễu" hoặc "ngoại lai" do sử dụng các đối tượng medoid để biểu diễn các cụm dữ liệu. PAM làm việc hiệu quả với tập dữ liệu nhỏ nhưng kém hiệu quả đối với tập dữ liệu lớn, độ phức tạp là $O(lk(n-k)^2)$

CLARA (C lustering LARge Applications, (Kaufmann and Rousseeuw, 1990)) nhằm khắc phục những nhược điểm của PAM khi xử lý những tập dữ liệu lớn bằng cách tiến hành trích mẫu của các tập dữ liệu.

CLARANS ((A Clustering Algorithm Based On Randomized Search, (Ng and Han, 1994)), kết hợp với thuật toán PAM với chiến lược tìm kiếm kinh nghiệm mới. Ý tưởng là thay thế ngay các đối tượng tâm medoid nếu việc thay thế này có ảnh hưởng tốt tới kết quả phân cụm chứ không cần phải xét hết các khả năng có thể xảy ra. Clarans không bị giới hạn không gian tìm kiếm như Clara.

### 1.3.2 Phân cụm phân cấp

Các thuật toán phân cụm phân cấp: Birch, Cure, Agnes, Diana, Chameleon,…
Đặc điểm chính của các thuật toán phân cụm phân cấp:

Dựa vào cấu trúc của cây phân cụm được tạo bởi sự phân chia đệ quy hoặc sự kết hợp bởi các thuật toán đã biết. Có hai hướng tiếp cận chính là hòa nhập nhóm và phân chia nhóm.

BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) phù hợp đối với tập dữ liệu lớn, tốc độ thực hiện nhanh. BIRCH có hạn chế là không xử lý tốt nếu các cụm không có dạng hình cầu, độ phức tạp là $O(n^2)$.

CURE (Clustering Using Representatives, (Guha, Rastogi & Shim, 1998)) khắc phục được vấn đề chỉ thích hợp với các cụm hình cầu và "nhạy cảm" với dữ liệu ngoại lai mà rất nhiều thuật toán phân cụm khác vướng phải. CURE sử dụng nhiều hơn một điểm làm đại diện cho mỗi cụm để cho phép khám phá ra các cụm có hình dạng bất kỳ, "co cụm lại" để làm giảm tác động của các phần tử ngoại lai.

CHAMELEON (G.Karypis, E.H Han and V.Kumar, 1999) là cách tiếp cận sử dụng mô hình động để xác định các cụm được hình thành. CHAMELEON có khả năng cao hơn CURE trong việc khám phá cụm có hình thù bất kỳ và có chất lượng phân cụm tốt hơn CURE, độ phức tạp là $O(n^2)$

### 1.3.3 Phân cụm dựa trên mật độ

Các thuật toán phân cụm dựa trên mật độ: DBSCAN, OPTICS, DENCLUE,…Các thuật toán này dựa trên mật độ điểm để phân cụm.

Đặc điểm chính của các thuật toán này:

Khám phá ra các cụm có hình dạng bất kỳ.

Khống chế được nhiễu.

Duyệt một lần.

Cần các tham số về mật độ như là các điều kiện kết thúc.

DBSCAN: thích nghi với tập dữ liệu có mật độ dày, khám phá được các cụm có hình dạng trong CSDL có nhiễu. dbscan định nghĩa cụm là tập tối đa các điểm cải tiến bằng cách giảm bớt tham số đầu vào.

OPTICS (Ordering Points To Indentify the Clustering Structure, 1999) là mở rộng của DBSCAN, có được cải tiến bằng cách giảm bớt tham số đầu vào. OPTICS không phân cụm các điểm dữ liệu mà thực hiện việc tính toán và sắp xếp trên các điểm dữ liệu theo thứ tự tăng dần nhằm tự động phân cụm dữ liệu, độ phức tạp là O(nlog(n)).

### 1.3.4 Phân cụm dựa trên lưới

Các thuật toán phân cụm dựa trên lưới: STING, CLIQUE, WAVECLUSTER,...
Đặc điểm chính của các thuật toán này:

Lượng hóa dữ liệu không gian vào một số hữu hạn các ô để tạo thành cấu trúc lưới và dựa trên đó để hoàn thành việc phân cụm.

STING (a STatistical INformation Grid approach, (Wang, Yang and Muntz, 1997)) là kỹ thuật phân cụm đa phân giải dựa trên lưới. sting: vùng không gian dữ liệu được phân rã thành một số hữu hạn các ô hình chữ nhật, các ô lưới được hình thành từ các ô lưới con để thực hiện việc phân cụm, độ phức tạp là O(n).

CLIQUE (CLustering In QUEst, (Agrawal, 1998)) là thuật toán hữu ích cho phân cụm dữ liệu không gian đa chiều trong các CSDL lớn thành không gian con. CLIQUE có thể xem xét trên cả hai kỹ thuật tiếp cận dựa trên mật độ và dựa trên lưới, độ phức tạp là O(n).

WAVECLUSTER (a multi-resolution clustering approach using wavelet method, (Sheikholeslami, 1998)) là phương pháp gần giống với sting, tuy nhiên thuật toán sử dụng phép biến đổi dạng sóng để tìm ô đặc trong không gian. WAVECLUSTER xử lý tập dữ liệu có hiệu quả, khám phá các cụm có hình dạng bất kỳ, xử lý được các phần tử ngoại lai, mẫn cảm với thứ tự vào và không phụ thuộc các tham số vào như số các cụm hoặc bán kính láng giếng, thời gian thực hiện phân cụm mảnh, độ phức tạp là O(n).

### 1.3.5 Phân cụm dựa trên mô hình

Các thuật toán phân cụm dựa trên mô hình: EM, COBWEB,…

EM phân cụm dựa trên sự phân phối xác suất, EM gán các đối tượng cho các cụm đã cho theo xác suất phân phối thành phần của các đối tượng đó. EM khám phá được nhiều dạng cụm khác nhau nhưng chi phí cho thuật toán là tương đối cao.

COBWEB (Fisher, Douglas H. (1987)) là cách tiếp cận để biểu diễn các đối tượng dữ liệu theo kiểu cặp thuộc tính – giá trị. COBWEB thực hiện bằng cách tạo cây phân lớp, tương tự như khái niệm BIRTCH, tuy nhiên cấu trúc cây khác nhau. COBWEB xây dựng cây phân lớp theo thứ tự tăng dần bằng cách chèn các đối tượng vào cây từng bước một, sau khi chèn lại duyệt lại toàn bộ cây từ gốc.

# 2 Thuật toán Kmeans

## 2.1 Giới thiệu sơ lược về K-means:

K-means (MacQueen, 1967; Anderberg, 1973) là một thuật toán khai phá dữ liệu dùng để phận cụm đối tượng mà không cần biết các kiến thức về các mối quan hệ. Thuật toán K-means là một trong những kỹ thuật phân cụm đơn giản nhất và nó thường được sử dụng trong ảnh y học , sinh học và các lĩnh vực liên quan.

Thuật toán K-means là một non-hierachahical (không có thứ bậc) clustering, sẽ phân chia dữ liệu thành k nhóm

## 2.2 Thuật toán K-means:

### 2.2.1 Ý tưởng

Thuật toán K-means sẽ phân chia dữ liệu thành k nhóm. Mỗi nhóm sẽ có một điểm trung tâm gọi là centroid. Các đối tượng trong một nhóm có khuynh hướng giống nhau nhiều hơn so với các đối tượng thuộc nhóm khác "losenest" và "similar (lose together)".

### 2.2.2 Độ đo trong thuật toán K-means:

Thuật toán k-means mặc định độ đo "losenest" là khoảng cách Euclidean.

Khoảng cách Euclidean là loại khoảng cách được sử dụng phổ biến nhất.

Khoảng cách Euclidean giữa hai điểm p và q là độ dài đoạn thẳng nối giữa chúng ($\overline{pq}$)

Nếu p = (p1,p2,..,pn) và q = (q1,q2,…,qn) là hai điểm nằm trong không gian n , khoảng cách giữa p, q được tính bởi công thức.

$$d(p,q)=d(q,p)=\sqrt{(q1-p1)^2+(q-p2)^2+\cdots+(qn-pn)^2}$$

#### 2.2.2.1 Xác định trung tâm: Điểm trung bình (mean centroid)

$$m_i = \frac{1}{|C_i|}\sum_{c\in C_i} x$$

(vectơ) $m_i$ là điểm trung tâm của nhóm $C_i$

$|C_i|$: là kích thước của nhóm $C_i$

Ví dụ: ta có 3 điểm A(1,2) B(3,2) C(1,2) khi đó ta xác định trọng tâm G=m($m_1$,$m_2$) của nhóm như sau.

$m_1=\frac{1}{3}(1+3+1)=\frac{5}{3}$, $m_2=\frac{1}{3}(2+2+2)=2$

### 2.2.2.2 Các bước của thuật toán:

**Thuật toán k-means gồm các bước:**

- ✓ Bước 0: Bắt đầu với vùng ngẫu nhiên vào K cụm
- ✓ Bước 1: Tạo một vùng mới bằng cách phân chia điểm dữ liệu đến trung tâm (centroid) gần nó nhất.
- ✓ Bước 2: Cập nhật lại giá trị trung tâm mới cho mỗi cụm
- ✓ Bước 3: Thực hiện lại bước 1 và bước 2 đến khi các cụm không còn thay đổi thành viên (các centroid không còn thay đổi giá trị) hay nói cách khác đến khi các cụm hội tụ

**Thuật toán được trình bày cụ thể như sau:**

**Input**: Kho dữ liệu D, số cụm K.

**Output**: Tập hợp các cụm C và các thành viên của mỗi cụm.

Chọn ngẫu nhiên K dữ liệu điểm từ D. Mỗi điểm K là đại diện của một cụm C

**While** Chưa hội tụ

**For each** x ∈ D

Tính khoảng cách từ x đến trung tâm mỗi cụm.

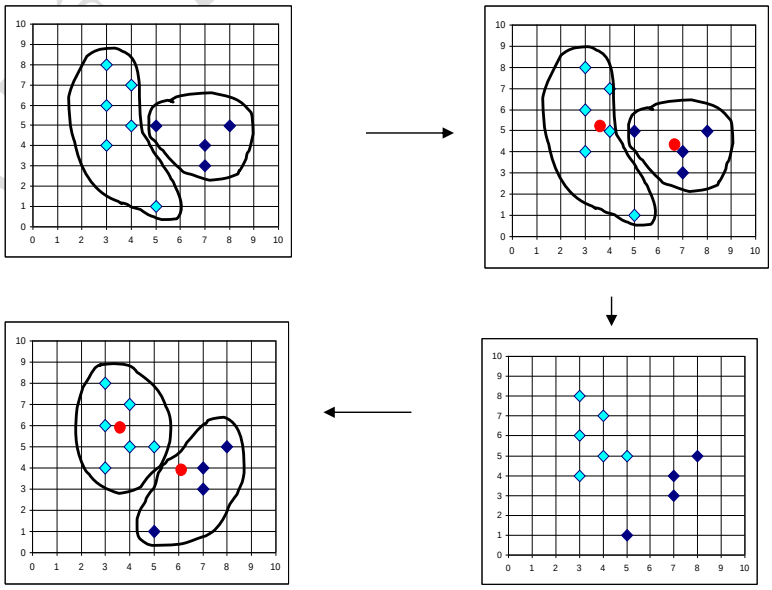
Xếp x vào cụm gần nó nhất

**End foreach**

Xác định lại trung tâm của mỗi cụm

**End while**

**Return** {k cụm}



Hình 5: Minh họa thuật toán Kmeans

## 3 Ví dụ thuật toán K-means phân cụm:

**Dùng thuật toán K-means phân 6 điểm A, B, C, D, E, F, G, H thành 3 cụm**

| Điểm | Tọa độ X | Tọa độ Y |
|---|---|---|
| **A** | **2** | **10** |
| **B** | **2** | **5** |
| **C** | **8** | **4** |
| **D** | **5** | **8** |
| **E** | **7** | **5** |
| **F** | **6** | **4** |
| **G** | **1** | **2** |
| **H** | **4** | **9** |

**Bước 0**: Khởi tạo ngẫu nhiên 3 cụm c1, c2, c3 trọng tâm của 3 cụm lần lượt là A, D, G

**Bước 1**: Tính khoảng cách từ các đối tượng đến trọng tâm của các cụm, và cập nhật lại cụm cho các đối tượng:

**Áp dụng khoảng cách: d(p,q) =** $\sqrt{(\boldsymbol{Xq} - \boldsymbol{Xp})^2 + (\boldsymbol{Yq} - \boldsymbol{Yp})^2}$

***Tính khoảng các từ điểm B đến 3 cụm:**

d(B,c1)=$\sqrt{(2-2)^2 + (5-10)^2}$ = 5

d(B,c2)=$\sqrt{(2-5)^2 + (5-8)^2}$ = 4.24

d(B,c3)=$\sqrt{(2-1)^2 + (5-2)^2}$ = 3.16

→d(B,c3)<d(B,c2)<d(B,c1) → B thuộc cụm c3.

***Tương tự với các điểm còn lại:**

d(C,c1)= 8.49

d(C,c2)=5

d(C,c3)=7.28

**→C thuộc cụm c2.**

d(E,c1)=7.07

d(E,c2)= 4.24

d(E,c3)=6.701

**→ E thuộc cụm c2**

d(F,c1)= 7.21

d(F,c2)= 4,12

d(F,c3)= 5.3

**→F thuộc cụm c2**

d(H,c1)=2.23
d(H,c2)=1.14
d(H,c3)=7.61
→ **H thuộc cụm c2**

**Bước 2:** Cập nhật vị trí trung tâm cho các cụm

- Cụm c1 có 1 điểm A(2,10)
  - ✓ Trọng tâm cụm c1≡ A(2,10)
- Cụm c2 gồm 5 điểm: D(5,8), C(8,4), E(7,5), F(6,4), H(4,9)
  - ✓ Trọng tâm cụm c2 = $\left(\frac{5+8+7+6+4}{5}, \frac{8+4+5+4+9}{5}\right)$=(6,6)
- Cụm c3 gồm 2điểm: G(1,2), B(2,5).
  - ✓ Trọng tâm cụm c2 = $\left(\frac{1+2}{2}, \frac{2+5}{2}\right)$=$(\frac{3}{2},\frac{7}{2})$

**Bước 3**: Lặp lại bước 1 và bước 2:

| | X | Y | Khoảng cách đến C1 | Khoảng cách đến C2 | Khoảng cách đến C3 | Kết luận thuộc cụm |
|---|---|---|---|---|---|---|
| A | **2** | **10** | 0 | 5.656854 | 6.519202 | C1 |
| B | **2** | **5** | 5 | 4.123106 | 1.581139 | C3 |
| C | **8** | **4** | 8.485281 | 2.828427 | 6.519202 | C2 |
| D | **5** | **8** | 3.605551 | 2.236068 | 5.700877 | C2 |
| E | **7** | **5** | 7.071068 | 1.414214 | 5.700877 | C2 |
| F | **6** | **4** | 7.211103 | 2 | 4.527693 | C2 |
| G | **1** | **2** | 8.062258 | 6.403124 | 1.581139 | C3 |
| H | **4** | **9** | 2.236068 | 3.605551 | 6.041523 | C1 |

**Tính trọng tâm cụm**

| | X | Y |
|---|---|---|
| **Trọng tâm C1** | 3 | 9.5 |
| **Trọng tâm C2** | 6.5 | 5.25 |
| **Trọng tâm C2** | 1.5 | 3.5 |

→ Trọng tâm thay đổi

**Bước 4: Lặp lại bước 1 và 2**

| | X | Y | Khoảng cách đến C1 | Khoảng cách đến C2 | Khoảng cách đến C3 | Kết luận thuộc cụm |
|---|---|---|---|---|---|---|
| A | **2** | **10** | 1.118034 | 6.543126 | 6.519202 | C1 |
| B | **2** | **5** | 4.609772 | 4.506939 | 1.581139 | C3 |
| C | **8** | **4** | 7.433034 | 1.952562 | 6.519202 | C2 |
| D | **5** | **8** | 2.5 | 3.132491 | 5.700877 | C1 |
| E | **7** | **5** | 6.020797 | 0.559017 | 5.700877 | C2 |
| F | **6** | **4** | 6.264982 | 1.346291 | 4.527693 | C2 |
| G | **1** | **2** | 7.762087 | 6.388466 | 1.581139 | C3 |
| H | **4** | **9** | 1.118034 | 4.506939 | 6.041523 | C1 |

| | X | Y |
|---|---|---|
| Trọng tâm C1 | 3.666667 | 9 |
| Trọng tâm C2 | 7 | 4.333333 |
| Trọng tâm C2 | 1.5 | 3.5 |

→Trọng tâm thay đổi, tính lại khoảng cách với trọng tâm mới

**Bước 5**: Lặp lại 1 và 2

| | X | Y | Khoảng cách đến C1 | Khoảng cách đến C2 | Khoảng cách đến C3 | Kết luận thuộc cụm |
|---|---|---|---|---|---|---|
| A | **2** | **10** | 1.943651 | 7.557189 | 6.519202 | C1 |
| B | **2** | **5** | 4.333333 | 5.044249 | 1.581139 | C3 |
| C | **8** | **4** | 6.616478 | 1.054093 | 6.519202 | C2 |
| D | **5** | **8** | 1.666667 | 4.176655 | 5.700877 | C1 |
| E | **7** | **5** | 5.206833 | 0.666667 | 5.700877 | C2 |
| F | **6** | **4** | 5.517648 | 1.054093 | 4.527693 | C2 |
| G | **1** | **2** | 7.490735 | 6.437736 | 1.581139 | C3 |
| H | **4** | **9** | 0.333333 | 5.547772 | 6.041523 | C1 |

| | X | Y |
|---|---|---|
| **Trọng tâm C1** | 3.666667 | 9 |
| **Trọng tâm C2** | 7 | 4.333333 |
| **Trọng tâm C2** | 1.5 | 3.5 |

→Trọng tâm không thay đổi kết thúc thuật toán

**Kết luận: Cụm C1 gồm A,D,H. C2 gồm C,E,F.C3 gồm B,G**

## 3.1 Biến thể của thuật toán K-means – Giải thuật k - modes

### 3.1.1 Đo lường sự không giống nhau (Dissimilarity measure):

$$\mathbf{d_1(X,Y)} = \sum_{j=1}^{m} \delta(x_j, y_j)$$

$$\delta(x_j, y_j) = \begin{cases} 0 & (x_j = y_j) \\ 1 & (x_j \neq y_j) \end{cases}$$

### 3.1.2 Thiết lập mode

Dataset X ={$X_1,X_2,\ldots,X_n$} gồm các categorical(nominal) objects,mỗi object lần lược có các thuộc tính $A_1,A_2,\ldots,A_m$.

Giả sử mỗi cụm có mode Q = [$q_1,q_2,\ldots,q_n$] thì D(X,Q) là nhỏ nhất nếu :

$f_r(A_j= q_j|X) \geq f_r(A_j=c_{k,j}|X)$ với $q_j \neq c_{k,j}$ cho tất cả j= 1,….,m

Với $f_r(A_j=c_{k,j}|X)= \frac{n_{c_{k,j}}}{n}$ : là tần số xuất hiện của $c_{k,j}$ trong X

Thuật toán k-means không thể phân cụm dữ liệu có sự đo lường khác nhau.

### 3.1.3 Thuật toán

K-mode gồm các bước

Bước 1: Chọn k mode cho k cụm

Bước 2: Phân bố các object vào cụm gần nó nhất: sử dụng đo lường không giống nhau (Dissimilarity measure)

Bước 3: Thiết lập mode của mỗi cụm.

Bước 4: Lặp lại bước 2 và bước 3 đến khi không có phần tử nào thay đổi cụm.

## 3.2 Đánh giá thuật toán K – means

### 3.2.1 Ưu điểm

- Đơn giản : dể hiểu và dễ cài đặt
- Thuật toán luôn có K cụm
- Mỗi cụm luôn có ích nhất một phần tử.
- Các cụm không phân cấp và không chồng chéo lên nhau (The clusters are non-hierarchical and they do not overlap.)
- Độ phức tạp về thời gian ~O(r.k.t) với:
    - r: kích thước của tập dữ liệu
    - k: tổng số nhóm thu được
    - t: tổng số lần lặp trong quá trình phân nhóm
    - Nếu k và t nhỏ thì thuật toán k-means được xem như có độ phức tạp ở mức tuyến tính.

### 3.2.2 Nhược điểm

- Thuật toán không có khả năng xác định chính xác số cụm dữ liệu sẽ phân ra, nên người dùng phải nhập rõ số lượng cụm muốn phân ra.
- Khó khăn trong việc so sánh chất lượng của các cụm được tạo ra vì các phân cụm có thể khác nhau khi khác giá trị K ban đầu.
- Khả năng chịu đựng nhiễu không tốt (ảnh hưởng bởi các phần tử outliers), giải pháp khắc phục outlier
    - *Giải pháp 1*: Trong quá trình phân nhóm, cần loại bỏ một số các mẫu quá khác biệt (cách xa)) các điểm trung tâm (centroids) so với các mẫu khác. Để chắc chắn (không loại nhầm), theo dõi các ví dụ ngoại lai (outliers) qua một vài bước lặp trong phân nhóm để kiểm tra chắc chắn trước khi quyết định loại bỏ
    - *Giải pháp 2*: Thực hiện lấy mẫu ngẫu nhiên, do quá trình lấy mẫu ngẫu nhiên chỉ chọn một tập con của dữ liệu nên khả năng outlier được chọn lả rất nhỏ.
    - *Giải pháp 3*: Sử dụng K-Medoids, không sử dụng giá trị trung bình, nhưng sử dụng phần tử ngay giữa.
- Không làm việc tốt đối với các cụm hình cầu,chỉ áp dụng dữ liệu của thuộc tính số

## 3.3 Các ứng dụng thực tế bằng thuật toán Kmeans:

Thuật toán k-means được sử dụng thành công trong nhiều chủ đề khác nhau:

- Phân loại thị trường (Market segmentation).
- Thiên văn học cho nông nghiệp(astronomy to agriculture)
- WWW: Phân nhóm tin tức (Internet newsgroup ar-ticles).
- Phân đoạn hình ảnh:Một số ứng dụng thực tế của phân khúc hình ảnh là:
    - Hình ảnh y tế
        - Xác định vị trí khối u và bệnh lý khác
        - Đo lường khối lượng mô
        - Hướng dẫn phẫu thuật máy tính
        - Chẩn đoán
        - Điều trị lập kế hoạch
        - Nghiên cứu cấu trúc giải phẫu
    - Xác định vị trí các đối tượng trong hình ảnh vệ tinh (đường, rừng, vv)
    - Nhận dạng khuôn mặt
    - Iris công nhận
    - Nhận dạng vân tay
    - Hệ thống điều khiển giao thông

- Phanh ánh sáng phát hiện
- Máy tầm nhìn
- Bệnh phát hiện cây trồng nông nghiệp hình ảnh

# 4 Minh họa thuật toán K - means bằng Weka

## 4.1 Giới thiệu về Weka

Weka là một công cụ phần mềm miễn phí được viết bằng Java, phục vụ lĩnh vực máy học và khai thác dữ liệu. Weka được phát triển bởi Đại học Waikato, New Zealand.

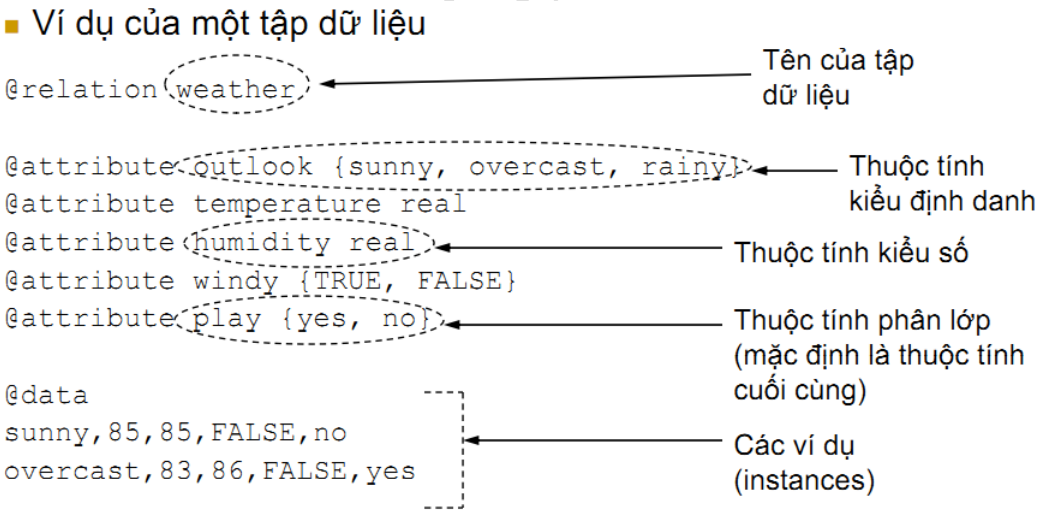
Các tính năng chính của Weka bao gồm:

- Một tập các công cụ tiền xử lý dữ liệu, các giải thuật học máy, khai phá dữ liệu và các phương pháp thí nghiệm đánh giá.
- Giao diện đồ họa, bao gồm cả tính năng hiển thị hóa dữ liệu.
- Môi trường cho phép so sánh các giải thuật máy học và khai phá dữ liệu.

Tải Weka tại địa chỉ: http://www.cs.waikato.ac.nz/ml/weka/

## 4.2 Thu thập, lưu trữ dữ liệu trong Weka (*.arrf,*.csv)

Dữ liệu được sử dụng trong Weka gồm 2 dạng ARRF, CSV




- Phần khai báo:

@relation <tên dữ liệu>

@attribute <tên thuộc tính 1><Kiểu dữ liệu>

@attribute <tên thuộc tính 2><Kiểu dữ liệu>

…

@attribute <tên thuộc tính n><Kiểu dữ liệu>

- Các kiểu dữ liệu
    - Numeric Dữ liệu dạng số
        - Ví dụ: @ATTRIBUTE name numeric
    - Nominal Dữ liệu rời rạc
        - Ví dụ: @ATTRIBUTE class {setosa, versicolor}
    - String Dữ liệu chuỗi
        - Ví dụ: @ATTRIBUTE name string
    - Date Dữ liệu kiểu ngày
        - Ví dụ: @ATTRIBUTE discovered date
- Dữ liệu thiếu được ký hiệu bằng dấu chấm hỏi "?"
- Phần dữ liệu: Mỗi mẫu dữ liệu được đặt trên một dòng, giá trị của các thuộc tính được liệt kê theo thứ tự từ trái qua phải và ngăn cách bởi dấu phẩy ","
- **Comma Separated Values (*.csv)**
- Là tập tin văn bản. Cấu trúc tương tự phần dữ liệu của tập tin arff: Các mẫu được lưu trên một dòng, các thuộc tính được ngăn cách bằng dấu phẩy. Dòng đầu tiên chứa tên các thuộc tính.

  Ví dụ: Một tập tin csv có nội dung như sau:

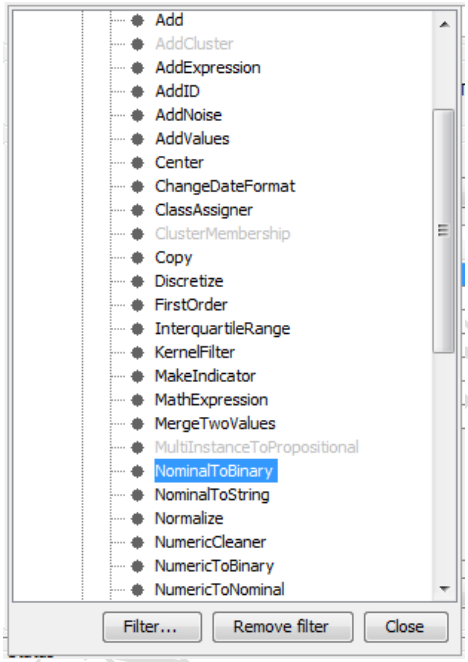
```
outlook,temperature,humidity,windy,play
sunny,hot,high,FALSE,no
sunny,hot,high,TRUE,no
overcast,hot,high,FALSE,yes
rainy,mild,high,FALSE,yes
rainy,cool,normal,FALSE,yes
rainy,cool,normal,TRUE,no
overcast,cool,normal,TRUE,yes
sunny,mild,high,FALSE,no
sunny,cool,normal,FALSE,yes
rainy,mild,normal,FALSE,yes
sunny,mild,normal,TRUE,yes
overcast,mild,high,TRUE,yes
overcast,hot,normal,FALSE,yes
rainy,mild,high,TRUE,no
```

Có nghĩa là dữ liệu này gồm có 14 mẫu và 5 thuộc tính (outlook, temperature, humidity, windy, play).

### 4.3 Bộ lọc filter trong Weka

- Weka cung cấp rất nhiều bộ lọc dùng trong tiền xử lý dữ liệu (preprocessing). Bộ lọc có nhiệm vụ thực hiện các thuật toán tiền xử lý dữ liệu để có thể làm việc với Kmeans.
- Thuật toán Kmeans sử dụng độ đo Euclidian, chỉ làm việc với dữ liệu dạng số (numeric) nên các bộ lọc có thể áp dụng cho dữ liệu gồm.

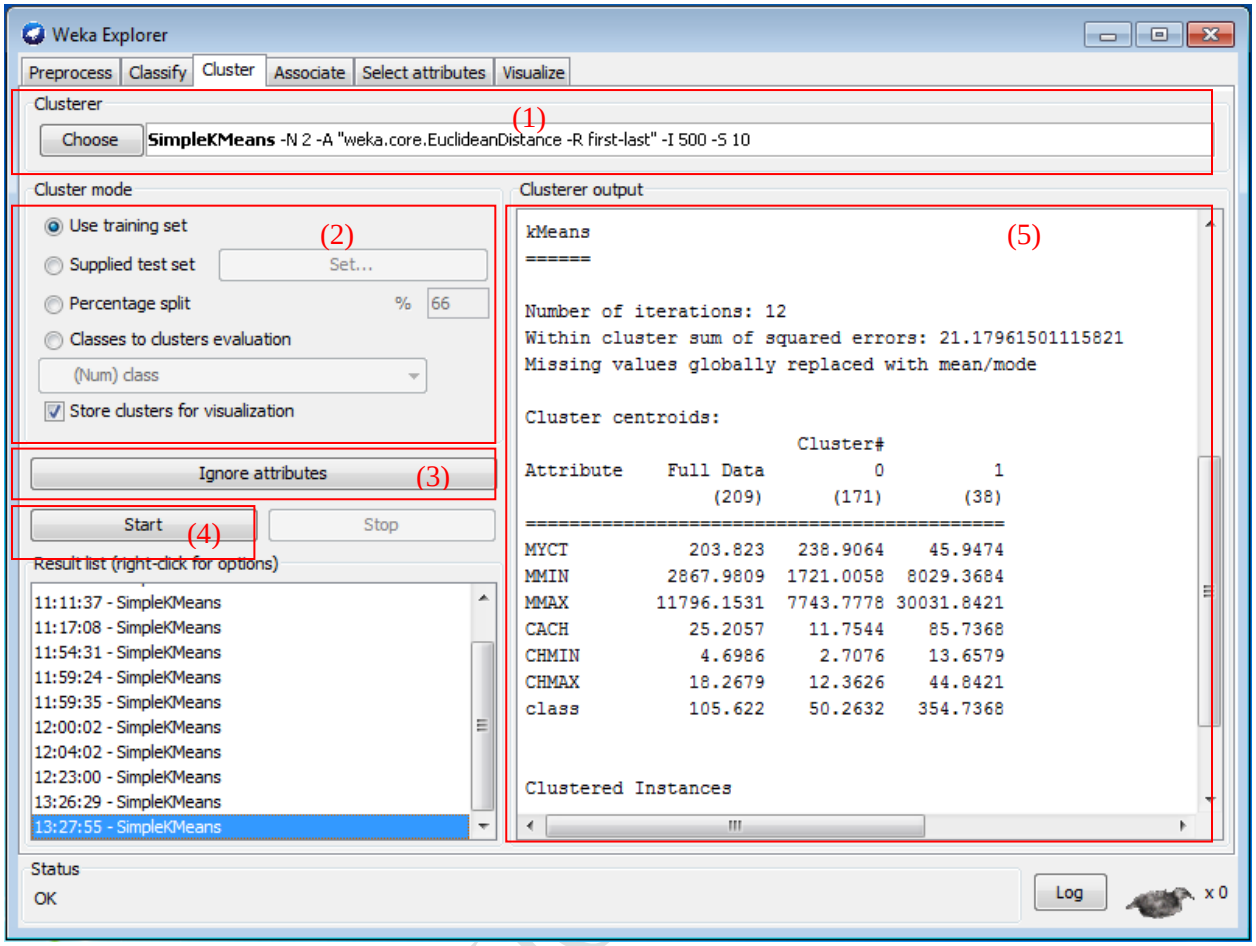
- StringtoNominal: Chuyển đổi dữ liệu từ chuỗi thành dạng định danh (Chú ý, Kmeans không làm việc với kiểu dữ liệu dạng string).
- NominaltoBinary: Chuyển đổi dữ liệu từ dạng định danh (Nominal) thành dạng số nhị phân.



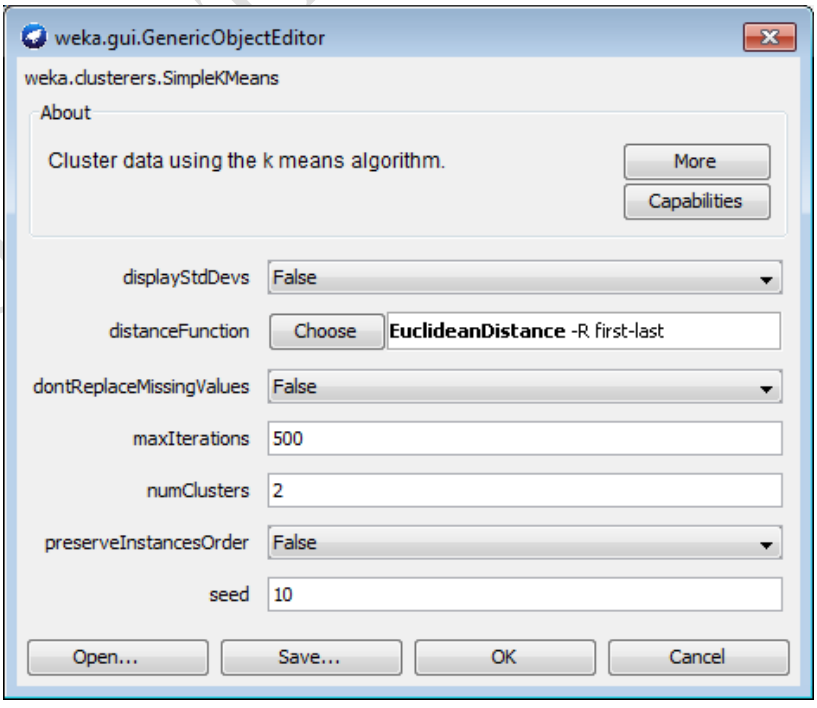

- Tuy nhiên, nếu ta bỏ qua bước tiền xử lý dữ liệu, **SimpleKmeans** trong Weka sẽ áp dụng độ đo **độ tương tự** cho thuật toán (giải thuật K-mode đã đề cập phần trên).

## 4.4 Tiến hành phân cụm trong Weka

- Chọn tab Preprocess để nạp dữ liệu, chọn đường dẫn của file .arrf hoặc .csv bằng cách chọn **Open file**. Có thể xem dữ liệu, chỉnh sửa dữ liệu bằng cách chọn **Edit**.
- Chọn bộ lọc (**Filter**) cần thiết bằng cách chọn **Choose, Weka** có rất nhiều bộ lọc nhưng ta chỉ cần quan tâm đến các bộ lọc cho **Kmeans** (StringtoNominal, NominalToBinary,…). Nếu dữ liệu mặc định kiểu số(numeric) ta không cần xử lý. Tuy nhiên SimpleKmeans sẽ chọn độ đo độ tương tự giữa các đối tượng so với điểm nằm giữa (mode)
- Chọn tab Cluster

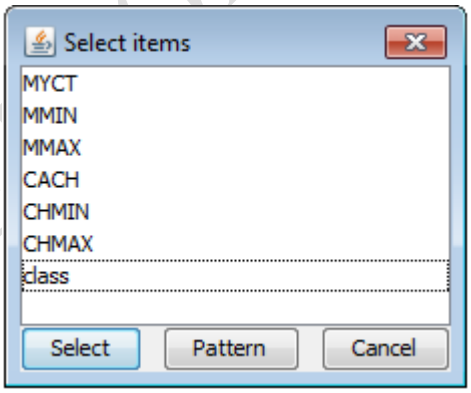



- (1)Clusterer : Lựa chọn mô hình Gom cụm SimpleKmeans

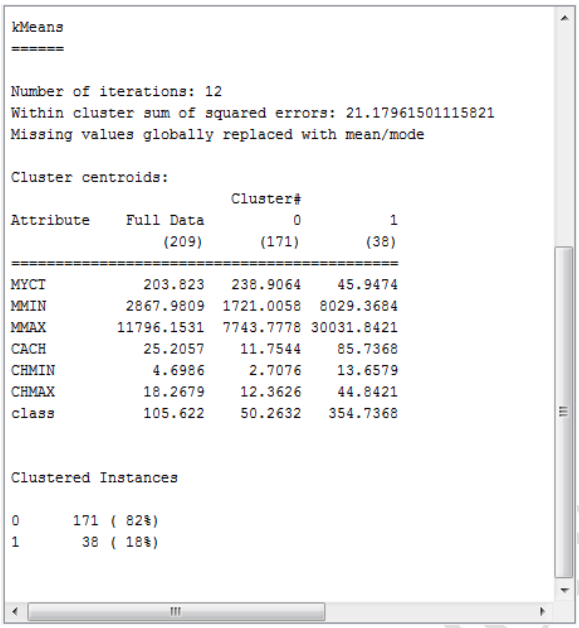


- displayStdDevs: chế độ hiển thị độ sai khác giữa các đối tượng của thuộc tính đến thuộc tính tương ứng của tâm cụm.
- distanceFunction : Chọn độ đo khoảng cách: Minskowki, Euclidean,…
- don'tReplaceMissingValues: Không thay thế giá trị khuyết bằng giá trị mặc định
- maxInterations: Số lần lặp tối đa.
- numCluster: Số nhóm phân cụm.
- preserveInstancesOrder: giữ vị trí của các instance trong dữ liệu.
- seed: tham số tùy chọn dùng để thực hiện chọn ngẫu nhiên k cụm.

- (2) Cluster mode: Các tùy chọn thực hiện với Kmeans
  - Use training set: Sử dụng chính tập dữ liệu huấn luyện để kiểm nghiệm.
  - Supplied test set: Sử dụng một tập dữ liệu khác để kiểm nghiệm
  - Percentage split : Chia dữ liệu thành 2 phần theo tỉ lệ %, một phần xây dựng mô hình, một phần dùng để kiểm thử
  - Classes to clusters evaluation: Gom cụm trên toàn bộ dữ liệu và đánh giá với tiêu chí độ lỗi là thấp nhất, Với phương pháp này, ta có thể áp dụng các phương pháp đánh giá ngoài để khảo sát chất lượng gom cụm.
- (3) Ignore attributes: Bỏ qua các thuộc tính chỉ định khi tiến hành gom cụm (ví dụ: ID, tên,…).




- (4)Tiến hành gom cụm
- (5)Cluster Output : Chứa kết quả gom cụm
  - Thông tin mô hình
  - Kết quả gom cụm

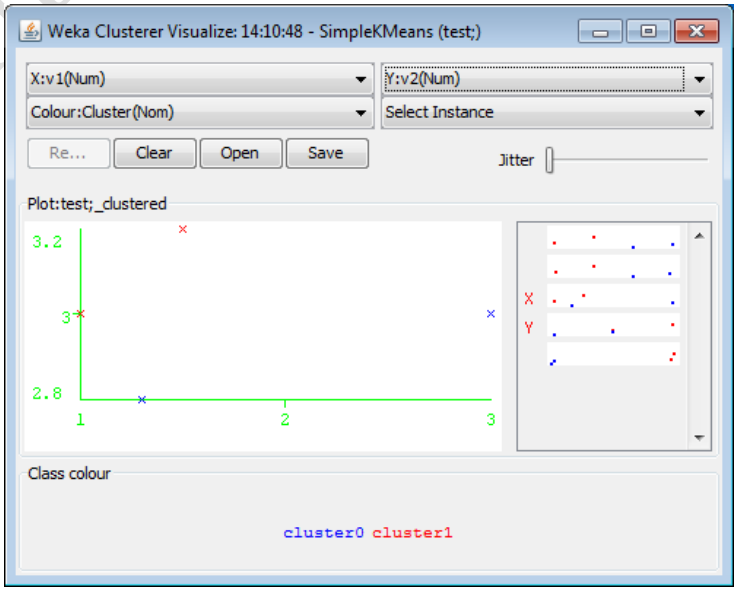


```
kMeans
======

Number of iterations: 12
Within cluster sum of squared errors: 21.17961501115821
Missing values globally replaced with mean/mode

Cluster centroids:
                         Cluster#
Attribute    Full Data          0          1
                 (209)      (171)       (38)
=============================================
MYCT            203.823   238.9064    45.9474
MMIN           2867.9809  1721.0058  8029.3684
MMAX          11796.1531  7743.7778 30031.8421
CACH             25.2057    11.7544    85.7368
CHMIN             4.6986     2.7076    13.6579
CHMAX            18.2679    12.3626    44.8421
class           105.622    50.2632   354.7368


Clustered Instances

0      171 ( 82%)
1       38 ( 18%)
```

Thông tin kết quả gom cụm bao gồm số lần lặp, các cụm và số đối tượng thuộc về các cụm đó.

## 4.5 Giải thích các kết quả khi phân cụm

- Chương trình liệt kê số lần thực hiện tính trọng tâm
- Liệt kê tâm cụm của toàn bộ dữ liệu
- Liệt kê các cụm được phân thành công dựa trên k-centroids đã nhập, số lượng các instance thuộc cụm đó và số phần trăm (%)
- Hiển thị dữ liệu (Visualize): Chương trình hiển thị dữ liệu đối với các kiểu dữ liệu 2D. Các điểm thuộc từng cụm sẽ được hiển thị với các màu khác nhau.



## TÀI LIỆU THAM KHẢO:


- **http://www.improvedoutcomes.com/docs/WebSiteDocs/Clustering/K-Means_Clustering_Overview.htm**
- **Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values**