ĐẠI HỌC THÁI NGUYÊN

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

ĐỖ THỊ KIM DUNG

TẠO LẬP HỆ LUẬT MỜ SỬ DỤNG PHÂN CỤM TRỪ MỜ DỮ LIỆU

LUẬN VĂN THẠC SĨ KHOA HỌC MÁY TÍNH

THÁI NGUYÊN, 2017

MỤC LỤC

DANH SÁCH CÁC HÌNH VẼ

DANH SÁCH CÁC BẢNG BIỂU

Lời đầu tiên cho em xin kính gửi các các thầy cô bộ muôn khoa Công nghệ. Cùng toàn thể lãnh đạo thầy cô đang giảng dạy và làm việc tại trường Đại Học Công Nghệ và Truyền Thông Thái Nguyên, lời chúc sức khỏe. Em xin chúc tất cả các thầy cô giáo luôn thành công trong sự nghiệp giáo dục đào tạo cũng như mọi lĩnh vực trong cuộc sống.

Em xin chân thành Cảm ơn Thầy PGS. TS Lê Bá Dũng, người đã trực tiếp hướng dẫn và nhiệt tình chỉ bảo để em có thể hoàn thành luận văn tốt nghiệp này.

Em xin cảm ơn Ban Giám hiệu, Quý thầy cô trường Đại Đại Học Công Nghệ và Truyền Thông Thái Nguyên đã trang bị cho em một lượng kiến thức bổ ích trong quá trình tôi học tập và thực hiện đề tài. Cảm ơn bạn bè đồng nghiệp đã động viên, giúp đỡ cho em trong suốt quá trình học tập và nghiên cứu.

Cuối cùng em xin chân thành cảm ơn các thành viên trong gia đình, những người luôn dành cho tôi những tình cảm nồng ấm và chia sẻ những lúc khó khăn trong cuộc sống, luôn động viên giúp đỡ tôi trong quá trình học tập và nghiên cứu.

Do kiến thức còn hạn hẹp nên không tránh khỏi những thiếu sót trong cách hiểu, lỗi trình bày. Em rất mong nhận được sự đóng góp ý kiến của quý thầy cô và Ban lãnh đao

Em xin trân trọng cảm ơn!

DANH SÁCH CÁC CHỮ VIẾT TẮT

PCDL	Phân Cụm Dữ Liệu
KPDL	Khai Phá Dữ Liệu
CSDL	Cơ Sở Dữ Liệu

LỜI CAM ĐOAN

Em xin cam đoan đây là luận văn do em nghiên cứu và thực hiện.

Các thông số, Hình ảnh và kết quả sử dụng trong luận văn là hoàn toàn có thật và chưa từng được công bố ở bất kỳ luận văn nào khác.

Thái Nguyên, ngày 16 tháng 5 năm 2017

Tác giả luận văn:

Đỗ Thị Kim Dung

MỞ ĐẦU

Sự phát triển nhanh chóng của các hệ thống thông tin như hiện nay, thì hệ mờ được áp dụng thành công trong nhiều lĩnh vực như điều khiển tự động, phân lớp dữ liệu, phân tích việc ra quyết định, các hệ chuyên gia. Hệ luật mờ xây dựng từ tri thức nói chung hay hệ suy luận mờ nói riêng được xây dựng theo suy diễn của con người, là một phần quan trọng trong ứng dụng logic mờ cũng như trong lý thuyết tập mờ vào thực tế. Trong nhiều ứng dụng cho thiết kế các hệ thống thông minh cũng như trong xây dựng các hệ trợ giúp quyết định, hệ mờ được xây dựng theo phân lớp dữ liệu, phân cụm dữ liệu, xây dựng cây quyết định.... Hệ mờ được thực hiện từ các luật mờ, các luật mờ được xây dựng từ các tri thức của các chuyên gia trong một lĩnh vực cụ thể.

Phân cụm dữ liệu đang là một vấn đề quan tâm nghiên cứu của các tác giả trong và ngoài nước và có nhiều thuật toán phân cụm được đề xuất. Trong đó, không ít thuật toán phân cụm kết hợp với việc sử dụng giải thuật di truyền trong quá trình thực hiện. Tuy nhiên các thuật toán được đưa ra mới chỉ xét đến khía cạnh phân chia dữ liệu thành các cụm với độ chính xác cao mà chưa để tâm đến sự tối ưu các luật sử dụng.

Trong các yêu cầu đặt ra cho quá trình phân cụm thì yêu cầu về độ chính xác luôn được đặt lên hàng đầu, ngoài ra với sự kết hợp các thuật toán phân cụm và giải thuật di truyền còn thỏa mãn được tính chất tối ưu của các luật được sử dụng. Vì vậy một cách tiếp cận khác mà luận văn nêu ra đó là xây dựng hệ luật mờ cho hệ mờ từ dữ liệu là một thực tế.

Phân cụm dữ liệu là quá trình phân chia một tập dữ liệu ban đầu thành các cụm dữ liệu sao cho các phần tử trong một cụm "tương tự" (similar) với nhau và các phần tử trong các cụm khác nhau sẽ "phi tương tự" (dissimilar) với nhau. Phân cụm dữ liệu là một phương pháp học không giám sát [7][8][9].

Hiện nay, các phương pháp phân cụm đã và đang được phát triển [6] và áp dụng nhiều trong các lĩnh vực khác nhau, bao gồm: nhận dạng, phân tích dữ liệu, nghiên cứu thị trường, xử lý ảnh, [1]… Các thuật toán phân cụm cũng rất đa dạng

như K-means, Pam, C-means, C-means mờ, thuật toán phân cụm trừ,… Để tăng tính ổn định và chính xác của kết quả phân cụm, ngày càng có các tiếp cận mới. Một trong những cách tiếp cận đang được nghiên cứu đó là ứng dụng lý thuyết mờ vào bài toán phân cụm dữ liệu.

Được sự gợi ý của giáo viên hướng dẫn và dựa trên những tìm hiểu của tôi trên đây, tôi quyết định chọn đề tài: "**Tạo lập hệ luật mờ sử dụng phân cụm trừ mờ dữ liệu**"

Phương pháp giúp cho chúng ta có cái nhìn nhiều chiều hơn đa dạng hơn, nhiều góc cạnh hơn về vấn đề cần giải quyết. Giúp cho các hệ tri thức hoạt động đảm bảo hơn có ý nghĩa khoa học và thực tiễn hơn.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT VỀ TẬP MỜ

1.1 **Trình bày tập Mờ.**

1.1.1 Định nghĩa tập mờ

Tập mờ A xác định trên tập vũ trụ X là một tập mà mỗi phần tử của nó là một cặp các giá trị $(x, \mu_A(x))$, trong đó $x \in X$ và μ_A là ánh xạ:

$$\mu_A: X \rightarrow [0,1]$$

Ánh xạ μ_A được gọi là hàm thuộc hoặc hàm liên thuộc (hoặc hàm thành viên - membership function) của tập mờ A. Tập X được gọi là cơ sở của tập mờ A.

$\mu_A(x)$ là độ phụ thuộc, sử dụng hàm thuộc để tính độ phụ thuộc của một phần tử x nào đó, có hai cách:

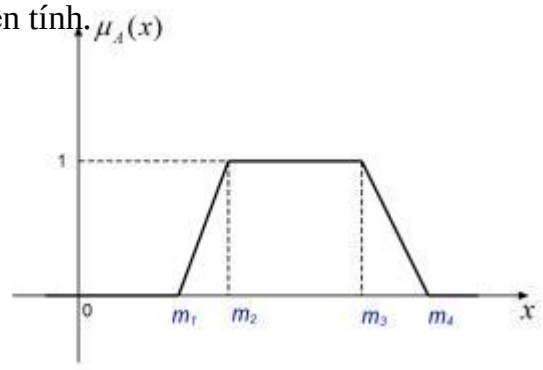
- Tính trực tiếp nếu $\mu_A(x)$ ở dạng công thức tường minh.
- Tra bảng nếu $\mu_A(x)$ ở dạng bảng.

Kí hiệu:

$$A = \{(\mu A(x) \mid x): x \in X\}$$

Các hàm thuộc $\mu_A(x)$ có dạng "trơn" được gọi là hàm thuộc kiểu S. Đối với hàm thuộc kiểu S, do các công thức biểu diễn $\mu_A(x)$ có độ phức tạp lớn nên thời gian tính độ phụ thuộc cho một phần tử lớn. Trong kỹ thuật điều khiển mờ thông thường, các hàm thuộc kiểu S thường được thay gần đúng bằng một hàm tuyến tính từng đoạn.

Một hàm thuộc có dạng tuyến tính từng đoạn được gọi là hàm thuộc có mức chuyển đổi tuyến tính.

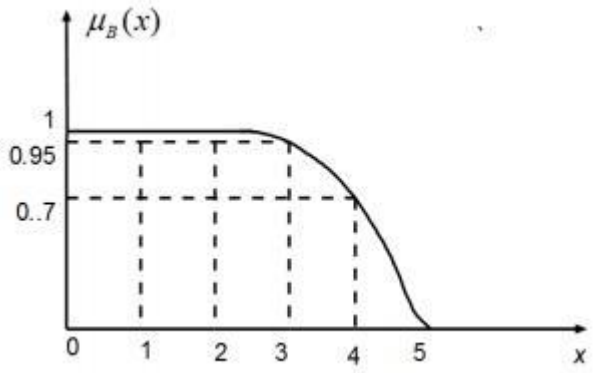


Hình 1. 1 Hàm Thuộc có mức chuyển đổi tuyến tính.

Hàm thuộc như trên với $m_1 = m_2$ và $m_3 = m_4$ chính là hàm thuộc của một tập vũ trụ.

Ví dụ 1: Một tập mờ B của các số tự nhiên nhỏ hơn 5 với hàm thuộc $\mu_B(x)$ có dạng như hình 1.2 định nghĩa trên tập vũ trụ *X* sẽ chứa các phần tử sau:

B = {(1,1), (2,1), (3,0.95), (4,0.7) }



Hình 1. 2 Hàm thuộc của tập B

Ví dụ 2: Xét X là tập các giá trị trong thang điểm 10 đánh giá kết quả học tập của học sinh về môn Toán, $X = \{1, 2, \ldots, 10\}$. Khi đó khái niệm mờ về năng lực học môn toán giỏi có thể được hiển thị bằng tập mờ *A* sau:

$A = 0.1/4 + 0.3/5 + 0.5/6 + 0.7/7 + 0.9/8 + 1.0/9 + 1.0/10$

Trong trường hợp tập mờ rời rạc ta có thể biểu diễn tập mờ ở dạng . Chẳng hạn, đối với tập mờ *A* ở trên ta có bảng như sau:

X	1	2	3	4	5	6	7	8	9	10
A	0	0	0	0.1	0.3	0.5	0.7	0.9	1.0	1.0

Bảng 1. 1 Bảng biểu tập mờ A

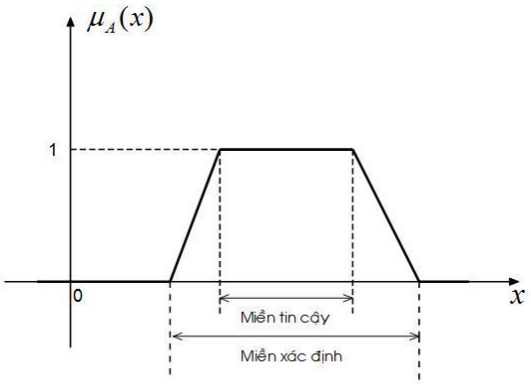
1.1.2. Một số khái niệm cơ bản của tập mờ

- **Miền xác định**: Biên giới tập mờ A, ký hiệu là *supp(A)*, là tập rõ gồm các phần tử của *X* có mức độ phụ thuộc của *x* vào tập mờ *A* lớn hơn 0.

  $supp(A) = \{ x \mid \mu_A(x) > 0 \}$

- **Miền tin cậy**: Lõi tập mờ *A*, ký hiệu là *core*(*A*), là tập rõ gồm các phần tử của *X* có mức độ phụ thuộc của *x* vào tập mờ *A* bằng 1.

core(*A*) = { *x* | *μA*(*x*) = 1}



Hình 1. 3 Miền xác định và miền tin cậy của tập mờ A

Độ cao tập mờ: Độ cao tập mờ *A*, ký hiệu: *h*(*A*), là mức độ phụ thuộc cao nhất của *x* vào tập mờ *A*.

$$h(A) = \sup_{x \in X} \mu_A(x)$$

Một tập mờ có ít nhất một phần tử có độ phụ thuộc bằng 1 được gọi là *tập mờ chính tắc*, tức là *h*(*A*) = 1, ngược lại một tập mờ *A* với *h*(*A*) < 1 được gọi là *tập mờ không chính tắc*.

1.1.3. Biểu diễn tập mờ

Tập mờ *A* trên tập vũ trụ *X* là tập mà các phần tử $x \in X$ với mức độ phụ thuộc của *x* vào tập mờ *A* tương ứng. Có ba phương pháp biểu diễn tập mờ: phương pháp ký hiệu, phương pháp tích phân và phương pháp đồ thị:

- *Phương pháp ký hiệu*: Liệt kê các phần tử và các thành viên tương ứng theo ký hiệu.

Cho X = {x1, x2, …,xn} là tập hữu hạn:

$$A = \sum_{i=1}^{n} \frac{\mu_A(x)}{x_i}$$

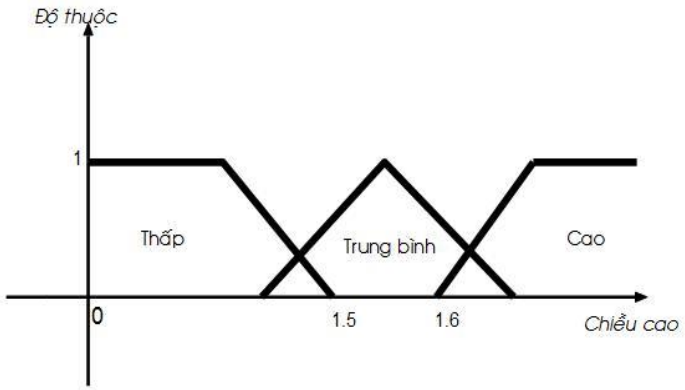
- *Phương pháp tích phân*: với *X* là tập vô hạn ta thường dùng ký hiệu sau:

$$A = \int_x \frac{\mu_A(x)}{x}$$

Lưu ý rằng các biểu thức trên chỉ có tính hình thức, các phép cộng +, phép tổng $\sum$ và phép lấy tích phân $\int$ đều không có nghĩa theo quy ước thông thường.

Tuy nhiên cách biểu diễn như vậy sẽ rất tiện dụng khi định nghĩa và thao tác các phép tính trên các tập mờ sau này.

Phương pháp đồ thị:



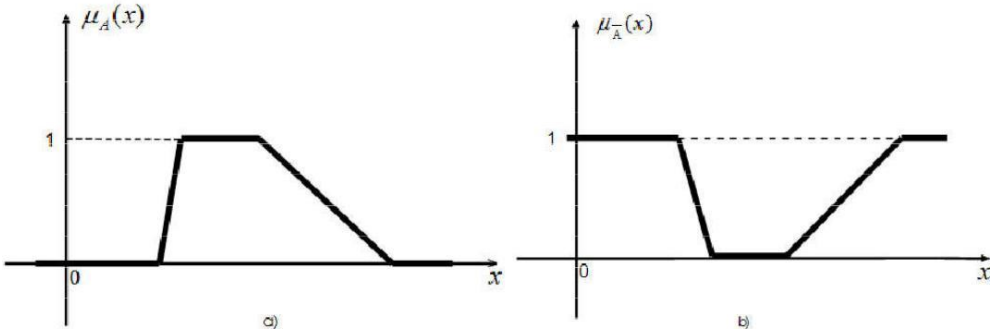
Hình 1. 4 Biểu diễn tập mờ chiều cao

1.2 Các phép toán trên tập mờ và hệ luật mờ

1.2.1 Phần bù của một tập mờ

Cho tập mờ A trên tập vũ trụ X, tập mờ bù của A là tập mờ $\bar{A}$, hàm thuộc $\mu_{\bar{A}}(x)$ được tính từ hàm thuộc $\mu_A(x)$:

$\mu_{\bar{A}}(x) = 1 - \mu_A$



Hình 1. 5Tập bù $\bar{A}$ của tập mờ A

Hàm thuộc của tập mờ $\bar{A}$

Một cách tổng quát để tìm $\mu_{\bar{A}}(x)$ từ $\mu_A(x)$, ta dùng hàm bù c,

c: [0,1] → [0,1] như sau:

$\mu_{\bar{A}}(x) = c(\mu_A(x))$

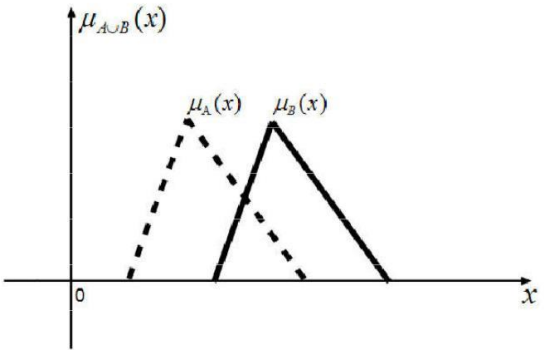
1.2.2 Phép hợp của các tập mờ

Cho tập mờ *A*, *B* trên tập vũ trụ *X*, tập mờ hợp của *A* và *B* là một tập mờ, ký hiệu là $C = A \cup B$.

Theo phép hợp chuẩn ta có $\mu_C(x)$ từ các hàm thành viên $\mu_A(x)$, $\mu_B(x)$

như sau:

$$\mu_C(x) = \mu_A \cup_B (x) = \max[\mu_A(x), \mu_B(x)], x \in X$$



Hình 1. 6 Hợp hai tập mờ có cùng tập nền

Một cách tổng quát ta dùng hàm hợp $u : [0,1] \times [0,1] \rightarrow [0,1]$. Hàm thành viên $\mu_C(x)$ có thể được suy từ hàm thành viên $\mu_A(x)$, $\mu_B(x)$ như sau:

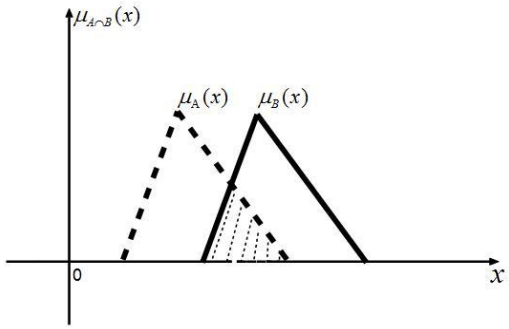
$$\mu_C(x) = u(\mu_A(x),\mu_B(x))$$

1.2.3 Phép giao của các tập mờ

Cho *A*, *B* là hai tập mờ trên tập vũ trụ *X*, tập mờ giao của *A* và *B* cũng là một tập mờ, ký hiệu: $I = A \cap B$.

Theo phép giao chuẩn ta có $\mu_I(x)$ từ các hàm thành viên $\mu_A(x)$, $\mu_B(x)$ như sau:

$$\mu_I(x) = \mu_A \cap_B (x) = min[\mu_A(x),\mu_B(x)], x \in X$$



Hình 1. 7 Giao hai tập mờ có cùng tập vũ trụ

Một cách tổng quát ta dùng hàm giao $i : [0,1] \times [0,1] \rightarrow [0,1]$. Hàm thành viên $\mu_I(x)$ có thể được suy từ hàm thành viên $\mu_A(x)$, $\mu_B(x)$như sau:

$$\mu_I(x) = i(\mu_A(x), \mu_B(x))$$

1.2.4 Tích Descartes các tập mờ

Cho A_i là các tập mờ trên tập vũ trụ X_i, $i = 1, 2, \ldots, n$. Tích Descartes của các tập mờ A_i, ký hiệu là $A_1 \times A_2 \times \ldots \times A_n$ hay $\prod_{i-1}^{n} A_i$, là một tập mờ trên tập vũ trụ $X_1 \times X_2 \times \ldots \times X_n$ được định nghĩa như sau:

$$A_1 \times A_2 \times \ldots \times A_n = \int_{x_1 \times x_2 \times x_n} \mu_{A_1}(x_1) \cap \ldots \cap \mu_{A_n}(x_n) / (x_1, \ldots, x_n)$$

Ví dụ 3: Cho $X1 = X2 = \{1, 2, 3\}$ và 2 tập mờ

$A = 0,5/1 + 1,0/2 + 0,6/3$ và $B = 1,0/1 + 0,6/2$

Khi đó:

$A \times B = 0,5/(1,1) + 1,0/(2,1) + 0,6/(3,1) + 0,5/(1,2) + 0,6/(2,2) + 0,6/(2,3)$

Một ví dụ ứng dụng của tích Descartes là kết nhập *(aggregation)* các thông tin mờ về các thuộc tính khác nhau của một đối tượng. Ví dụ trong các hệ luật của các hệ trợ giúp quyết định hay hệ chuyên gia, hệ luật trong điều khiển thường có các luật dạng sau đây:

Nếu x_1 *là* A_1 *và* x_2 *là* A_2 *và… và* x_n *là* A_n *thì y là B*

Trong đó, các x_i là các biến ngôn ngữ (vì giá trị của nó là các ngôn ngữ được xem như là nhãn của các tập mờ) và A_i là các tập mờ trên tập vũ trụ X_i của biến x_i. Hầu hết các phương pháp giải liên quan đến các luật "nếu - thì" trên đều đòi hỏi việc

tích hợp các dữ liệu trong phần tiền tố "nếu" nhờ toán tử kết nhập, một trong những toán tử như vậy là lấy tích Descartes $A_1 \times A_2 \times \ldots \times A_n$.

1.2.5 Tính chất của các phép toán trên tập mờ

Như các phép toán trên tập rõ, các phép toán trên tập mờ cũng có một số tính chất sau đối với các tập mờ *A, B, C* trên tập vũ trụ *X*:

- Giao hoán:

  $A \cap B = B \cap A$

  $A \cup B = B \cup A$

- Kết hợp:

  $A \cap (B \cap C) = (A \cap B) \cap C$

  $A \cup (B \cup C) = (A \cup B) \cup C$

- Phân bố:

  $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

  $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

- Đẳng trị:

  $A \cap A = A$

  $A \cup A = A$

- Đồng nhất:

  $A \cap X = A$

  $A \cup \emptyset = A$

  $A \cup \emptyset = \emptyset$

  $A \cup X = X$

- Bắc cầu:

  $A \subseteq B, B \subseteq C \Rightarrow A \subseteq C$

1.2.6 Hệ luật mờ

Gồm nhiều mệnh đề dạng:

IF< tập các điều kiện được thoả mãn>THEN<tập các hệ quả >

Giả sử hệ luật gồm M luật $R_j (j=\overline{1,M})$ dạng

R^j: IF x_1 is A_1 and x_2 is A_2 and… x_n is A_n^j THEN y is B^j

Trong đó x_i (i = $\overline{1,n}$) là các biến đầu vào hệ mờ, *y* là biến đầu ra của hệ mờ - các biến ngôn ngữ, A_i^j là các tập mờ trong các tập đầu vào *X* và B^j là các tập mờ trong các tập đầu ra *Y* – các giá trị của biến ngôn ngữ (ví dụ: "Rất Nhỏ", "Nhỏ", "Trung bình", "Lớn", "Rất lớn") đặc trưng bởi các hàm thuộc $\mu_{A_j^i}$ và μ_{Bj}. Khi đó R^j là một quan hệ mờ từ các tập mờ đầu vào $X = X_1 \times X_2 \times ..... \times X_n$ tới các tập mờ đầu ra *Y*.

1.3 Lập luân xấp xỉ trong hệ mờ.

1.3.1 Logic mờ

Logic mờ dùng một công cụ chính là lý thuyết tập mờ. Logic mờ tập trung trên biến ngôn ngữ trong ngôn ngữ tự nhiên nhằm cung cấp nền tảng cho lập luận xấp xỉ với những vấn đề không chính xác, nó phản ánh cả tính đúng đắn lẫn sự mơ hồ của ngôn ngữ tự nhiên trong lập luận theo cảm tính.

1.3.2 Quan hệ mờ

1.3.2.1. Khái niệm về quan hệ rõ

- ***Định nghĩa 1***: Cho $X \neq \emptyset$, $Y \neq \emptyset$, $R \subset X \times Y$ là một quan hệ (quan hệ nhị nguyên rõ), khi đó:

$$R(x,y) = \begin{cases} 1 & if(x,y)\ (x,y) \in R\ (\Leftrightarrow xRy) \\ 0 & if(x,y) \notin Ry)(\Leftrightarrow \neg xR) \end{cases}$$

Khi $X = Y$ thì $R \subset X \times Y$ là quan hệ trên *X*

Quan hệ *R* trên *X* được gọi là:

- Phản xạ nếu: $R(x,x) = 1$ với $\forall \quad \forall x \in X$
- Đối xứng nếu: R(x,y) = R(y,x) với $\forall$x, y$\in X$
- Bắc cầu nếu: $(xRy) \wedge (yRz) \Longrightarrow (xRz)$ *với* $\forall x,y,z \in X$

- ***Định nghĩa 2***: *R* là quan hệ tương đương nếu *R* là quan hệ nhị nguyên trên *X* có tính chất phản xạ, đối xứng và bắc cầu.

1.3.2.2. Các quan hệ mờ

Các quan hệ mờ là cơ sở dùng để tính toán và suy diễn (suy luận xấp xỉ) mờ. Đây là một trong những vấn đề quan trọng trong các ứng dụng mờ đem lại hiệu quả lớn trong thực tế, mô phỏng được một phần suy nghĩ của con người. Chính vì vậy, mà các phương pháp mờ được nghiên cứu và phát triển mạnh mẽ. Một trong số đó là logic mờ mở. Tuy nhiên logic mờ mở rộng từ logic đa trị, do đó nảy sinh ra rất nhiều các quan hệ mờ, nhiều cách định nghĩa các toán tử T-chuẩn, T-đối chuẩn, cũng như

các phương pháp mờ hoá, khử mờ khác nhau,… Sự đa dạng này đòi hỏi người ứng dụng phải tìm hiểu để lựa chọn phương pháp thích hợp nhất cho ứng dụng của mình.

- ***Định nghĩa 3***: Cho $U \neq \emptyset$; $V \neq \emptyset$ là hai không gian nền; R là một tập mờ trên $U \times V$ gọi là một quan hệ mờ (quan hệ hai ngôi).

  $0 \leq R(x,y) = \mu_R(x,y) \leq 1$

  Tổng quát: $R \subset U_1 \times U_2 \times \ldots\ldots \times U_n$ là quan hệ n ngôi $0 \leq$

  $R(u_1, u_{2,\ldots\ldots}u_n) = \mu_R(u_1, u_{2,\ldots\ldots}u_n) \leq 1$

1.3.2.3. Các phép toán của quan hệ mờ

- ***Định nghĩa 4***: Cho R là quan hệ mờ trên $X \times Y$, S là quan hệ mờ trên $Y \times Z$, lập phép hợp thành SoR là quan hệ mờ trên $X \times Z$

Có $R(x,y)$ với $(x,y) \in X \times Y$, $S(y,z)$ với $(y,z) \in Y \times Z$. Định nghĩa phép hợp thành:
Phép hợp thành max – min xác định bởi:

$$(S\ O\ R)(x,z) = \quad Sup\ (min(R(x,y),S(y,z))) \quad \forall (x,z) \in X \times Z y \in Y$$

Phép hợp thành max – prod xác định bởi:

$$(So\ R)(x,z) = Sup\ (\min(R(x,y) \times S(y,z)))$$

$$\forall (x,z) \in X \times Z\ y \in Y$$

Phép hợp thành max – T (với T là T - chuẩn) xác định bởi:

$$(So\ TR)(x,z) = Sup\ (T(R(x,y), S(y,z))) \quad \forall (x,z) \in X \times Z\ y \in Y$$

1.3.3. Suy luận xấp xỉ và suy diễn mờ

Suy luận xấp xỉ hay còn gọi là suy luận mờ - đó là quá trình suy ra những kết luận dưới dạng các mệnh đề trong điều kiện các quy tắc, các luật, các dữ liệu đầu vào cho trước cũng không hoàn toàn xác định.

Trong giải tích toán học chúng ta sử dụng mô hình sau để lập luận:

Định lý: “Nếu một hàm số là khả vi thì nó liên tục”

Sự kiện: Hàm f khả vi

Kết luận: Hàm f là liên tục

Đây là dạng suy luận dựa vào luật logic cổ điển Modus Ponens. Căn cứ vào mô hình này chúng ta sẽ diễn đạt cách suy luận trên dưới dạng sao cho nó có thể suy rộng cho logic mờ.

Gọi Ω là không gian tất cả các hàm số, ví dụ Ω ={g:R→R}. *A* là các tập các hàm khả vi, *B* là tập các hàm liên tục. Xét hai mệnh đề sau: *P='g∈A'* và Q='g∈B'. Khi đó ta có:

Luật (tri thức):	$P \Longrightarrow Q$
Sự kiện:	*P* đúng (True)
Kết luận:	*Q* đúng (True)

Xét bài toán suy luận trong hệ mờ

Hệ mờ *n* biến vào $x_1, \ldots x_n$ và một biến ra *y*

Cho U_n, $i = 1..n$ là các không gian nền của các biến vào, *V* là không gian nền của biến ra.

Hệ được xác định bởi *m* luật mờ:

R_1: Nếu x_1 là A_{11}và x_2 là A_{12} và ….x_n là A_{1n} thì *y* là B_1

R_2: Nếu x_1 là A_{21} và x_2 là A_{22} và…x_n là A_{2n} thì *y* là B_2

.....

.....

R_m: Nếu x_1 là A_{m1} và x_2 là A_{m2} và ……x_n là A_{mn} thì *y* là B_m

Thông tin đầu vào:

x_1 là A_{01} và x_2 là A_{02} và….x_{0n} là A_{0n}

Tính: y là B_0

Trong đó biến mờ j_i, $i=\overline{1,n}$, $j = \overline{1,m}$ xác định trên không gian nền *U*, biến mờ *Bj*, ($j=\overline{1,n}$) xác định trên không gian nền *V*.

Để giải bài toán này chúng ta phải thực hiện qua các bước sau:

1. Xác định các tập mờ của các biến đầu vào.
2. Xác định độ liên thuộc tại các tập mờ tương ứng.
3. Xác định các quan hệ mờ $R_{(A.B)}(u,v)$.
4. Xác định phép hợp thành.

Tính B' theo công thức: $B' = A' \circ R_{(A,B)}(u,v)$.

CHƯƠNG 2. PHÂN CỤM DỮ LIỆU TRỪ MỜ.

2.1 Các Phương pháp phân cụm dữ liệu nói chung

Trong thực tế, phân cụm dữ liệu (PCDL) nhằm mục đích khám phá cấu trúc của mỗi dữ liệu để thành lập các nhóm dữ liệu từ tập dữ liệu lớn, từ đó nó cho phép người ta đi sâu vào phân tích và nghiên cứu cho từng cụm dữ liệu này nhằm khám phá và tìm kiếm các thông tin tiềm ẩn, hữu ích phục vụ cho việc ra quyết định [6,7,8,9]. Vì vậy, PCDL là một phương pháp xử lý thông tin quan trọng và phổ biến, nó nhằm khám phá mối liên hệ giữa các mẫu dữ liệu bằng cách tổ chức chúng thành các cụm.

Từ đó ta có thể khái quát hóa khái niệm PCDL: PCDL là một kĩ thuật trong khai phá dữ liệu (KPDL), nhằm tìm kiếm, phát hiện các cụm, các mẫu dữ liệu tự nhiên, tiềm ẩn, quan trọng trong tập dữ liệu lớn từ đó cung cấp thông tin, tri thức hữu ích cho việc ra quyết định.

Như vậy, PCDL là quá trình phân chia một tập dữ liệu ban đầu thành các cụm dữ liệu sao cho các phần tử trong một cụm "tương tự" với nhau và các phần tử trong các cụm khác nhau sẽ "phi tương tự" với nhau. Số các cụm dữ liệu được phân ở đây có thể được xác định trước theo kinh nghiệm hoặc có thể được tự động xác định của phương pháp phân cụm [1,3]

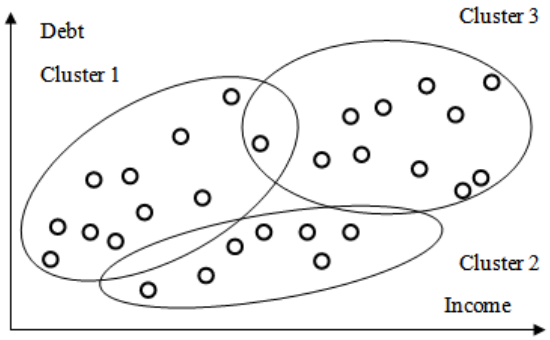
Trong PCDL khái niệm hai hoặc nhiều đối tượng cùng được xếp vào một cụm nếu chúng có chung một định nghĩa về khái niệm hoặc chúng xấp xỉ với các khái niệm mô tả cho trước.

Trong học máy, PCDL được xem là vấn đề học không có giám sát, vì nó phải giải quyết vấn đề tìm một cấu trúc trong tập hợp dữ liệu chưa biết trước các thông tin về lớp hay các thông tin về tập huấn luyện. Trong nhiều trường hợp, nếu phân lớp được xem là vấn đề học có giám sát thì PCDL là một bước trong phân lớp dữ liệu, PCDL sẽ khởi tạo các lớp cho phân lớp bằng cách xác định các nhãn cho các nhóm dữ liệu

Trong KPDL, người ta có thể nghiên cứu các phương pháp phân tích cụm có hiệu quả và hiệu suất cao trong cơ sở dữ liệu (CSDL) lớn. Những mục tiêu trước tiên

của nghiên cứu là tập trung vào khả năng mở rộng của các phương pháp phân cụm, tính hiệu quả của các phương pháp phân cụm với các hình dạng phức tạp, những kĩ thuật cho phân cụm với nhiều kiểu dữ liệu có kích cỡ lớn và những phương pháp cho PCDL tường minh và những dữ liệu dạng số hỗn hợp trong CSDL lớn. PCDL được sử dụng rộng rãi trong nhiều ứng dụng, bao gồm nhận dạng mẫu, phân tích dữ liệu, xử lý ảnh, nghiên cứu thị trường...

Hình 2.1 mô tả thực hiện phân cụm của tập dữ liệu giám sát nhiệt độ lò thành 3 cụm.



Hinh 2. 1 Ví dụ phân cụm của tập dữ liệu giám sát nhiệt độ lò thành 3 cụm

Vấn đề thường gặp trong PCDL là hầu hết các dữ liệu cần cho phân cụm đều có chứa dữ liệu "nhiễu" do quá trình thu thập thiếu chính xác hoặc thiếu đầy đủ, vì cần phải xây dựng chiến lược cho bước tiền xử lý dữ liệu nhằm khắc phục hoặc loại bỏ "nhiễu" trước khi bước vào giai đoạn phân tích PCDL. "nhiễu" ở đây có thể là các đối tượng dữ liệu không chính xác hoặc các đối tượng dữ liệu khuyết thiếu thông tin về một số thuộc tính. Một trong các kỹ thuật xử lý nhiễu phổ biến là việc thay thế giá trị của các thuộc tính của đối tượng "nhiễu" bằng giá trị thuộc tính tương ứng của đối tượng dữ liệu gần nhất.

Ngoài ra, dò tìm phần tử ngoại lai là một trong những hướng nghiên cứu quan trọng trong PCDL, chức năng của nó là xác định một nhóm nhỏ các đối tượng dữ liệu "khác thường" so với các dữ liệu khác trong CSDL – tức là đối tượng dữ liệu không tuân theo các hành vi hoặc mô hình dữ liệu – nhằm tránh sự ảnh hưởng của chúng tới

quá trình và kết quả của PCDL. Khám phá các phần tử ngoại lai đã được phát triển và ứng dụng trong viễn thông, dò tìm gian lận thương mại…

Tóm lại, PCDL là một vấn đề khó vì người ta phải đi giải quyết các vấn đề con cơ bản như sau:

- Biểu diễn dữ liệu.
- Xây dựng hàm tính độ tượng tự.
- Xây dựng các tiêu chuẩn phân cụm.
- Xây dựng mô hình cho cấu trúc cụm dữ liệu.
- Xây dựng thuật toán phân cụm và xác lập các điều kiện khởi tạo.
- Xây dựng các thủ tục biểu diễn và đánh giá kết quả phân cụm.

Theo các nghiên cứu thì đến nay chưa có một phương pháp phân cụm tổng quát nào có thể giải quyết trọn vẹn cho tất cả các dạng cấu trúc cụm dữ liệu. Hơn nữa, các phương pháp phân cụm cần có cách thức biểu diễn cấu trúc các cụm dữ liệu khác nhau, với mỗi cách thức biểu diễn khác nhau sẽ có một thuật toán phân cụm phù hợp. PCDL đang là vấn đề mở và khó vì người ta cần phải đi giải quyết nhiều vấn đề cơ bản như đã đề cập ở trên một cách trọn vẹn và phù hợp với nhiều dạng dữ liệu khác nhau. Đặc biệt đối tượng với dữ liệu hỗn hợp, đang ngày càng tăng trưởng không ngừng trong các hệ quản trị dữ liệu, đây cũng là một trong những thách thức lớn trong lĩnh vực KPDL trong những thập kỷ tiếp theo và đặc biệt trong lĩnh vực KPDL bằng phương pháp phân cụm dữ liệu.

Mục tiêu của phân cụm dữ liệu là xác định được bản chất nhóm trong tập dữ liệu chưa có nhãn. Nhưng để có thể quyết định được cái gì tạo thành một cụm tốt. Nó có thể được chỉ ra rằng không có tiêu chuẩn tuyệt đối "tốt" mà có thể không phụ thuộc vào kết quả phân cụm. Vì vậy, nó đòi hỏi người sử dụng phải cung cấp tiêu chuẩn này, theo các kết quả phân cụm sẽ đáp ứng được yêu cầu. Ví dụ, có thể quan tâm đến việc tìm đại diện cho các nhóm đồng nhất (rút gọn dữ liệu), trong tìm kiếm "các cụm tự nhiên" và mô tả các thuộc tính chưa biết (kiểu dữ liệu tự nhiên) hoặc tìm kiếm các đối tượng khác thường (dò tìm phần tử ngoại lai).

Phân cụm dữ liệu là một công cụ quan trọng trong một số ứng dụng. Sau đây là một số ứng dụng của nó:

- Giảm dữ liệu: Giả sử ta có một lượng lớn dữ liệu *(N)*. Phân cụm sẽ nhóm các dữ liệu này thành *m* cụm dữ liệu dễ nhận thấy và $m << N$. Sau đó xử lý mỗi cụm như một đối tượng đơn.
- Rút ra các giả thuyết: Các giả thuyết này có liên quan đến tính tự nhiên của dữ liệu và phải được kiểm tra bởi việc dùng một số tập dữ liệu khác.
- Kiểm định giả thuyết: Ta sẽ phân cụm để xét xem có tồn tại một tập dữ liệu nào đó trong tập dữ liệu thoả mãn các giả thuyết đã cho hay không. Chẳng hạn xem xét giả thuyết sau đây: *"Các công ty lớn đầu tư ra nước ngoài"*. Để kiểm tra, ta áp dụng kỹ thuật phân cụm với một tập đại diện lớn các công ty. Giả sử rằng mỗi công ty được đặc trưng bởi tầm vóc, các hoạt động ở nước ngoài và khả năng hoàn thành các dự án. Nếu sau khi phân cụm, một cụm các công ty được hình thành gồm các công ty lớn và có vốn đầu tư ra nước ngoài (không quan tâm đến khả năng hoàn thành các dự án) thì giả thuyết đó được củng cố bởi kỹ thuật phân cụm đã thực hiện.
- Dự đoán dựa trên các cụm: Đầu tiên ta sẽ phân cụm một tập dữ liệu thành các cụm mang đặc điểm của các dạng mà nó chứa. Sau đó, khi có một dạng mới chưa biết ta sẽ xác định xem nó sẽ có khả năng thuộc về cụm nào nhất và dự đoán được một số đặc điểm của dạng này nhờ các đặc trưng chung của cả cụm.

Cụ thể hơn, phân cụm dữ liệu đã được áp dụng cho một số ứng dụng điển hình trong các lĩnh vực sau [18]:

Thương mại: Trong thương mại, phân cụm có thể giúp các thương nhân khám phá ra các nhóm khách hàng quan trọng có các đặc trưng tương đồng nhau và đặc tả họ từ các mẫu mua bán trong cơ sở dữ liệu khách hàng.

Sinh học: Trong sinh học, phân cụm được sử dụng để xác định các loại sinh vật, phân loại các Gen với chức năng tương đồng và thu được các cấu trúc trong các mẫu.

Phân tích dữ liệu không gian: Do sự đồ sộ của dữ liệu không gian như dữ liệu thu được từ các hình ảnh chụp từ vệ tinh các thiết bị y học hoặc hệ thống thông tin địa lý

(GIS), …làm cho người dùng rất khó để kiểm tra các dữ liệu không gian một cách chi tiết. Phân cụm có thể trợ giúp người dùng tự động phân tích và xử lý các dữ liệu không gian như nhận dạng và chiết xuất các đặc tính hoặc các mẫu dữ liệu quan tâm có thể tồn tại trong cơ sở dữ liệu không gian.

Lập quy hoạch đô thị: Nhận dạng các nhóm nhà theo kiểu và vị trí địa lý, … nhằm cung cấp thông tin cho quy hoạch đô thị.

Nghiên cứu trái đất: Phân cụm để theo dõi các tâm động đất nhằm cung cấp thông tin cho nhận dạng các vùng nguy hiểm.

Địa lý: Phân lớp các động vật và thực vật và đưa ra đặc trưng của chúngWeb Mining: Phân cụm có thể khám phá các nhóm tài liệu quan trọng, có nhiều ý nghĩa trong môi trường Web. Các lớp tài liệu này trợ giúp cho việc khám phá tri thức từ dữ liệu,…

2. 2 Phân cụm dữ liệu trừ mờ

Phương pháp phân cụm trừ (subtractive clustering - SC) xác định các tâm cụm dựa trên mật độ (potential) các điểm lân cận. Xét một tập hợp dữ liệu gồm *n* điểm:

$$X = \{x_1, x_2, ..., x_n\}$$

Hàm tính mật độ cho một điểm dữ liệu là:

$$P_i = \sum_{j=1}^{n} e^{-\frac{4}{r_a^2}\|x_i - x_j\|^2} \qquad (2.1)$$

Trong đó:

P_i : Mật độ các điểm bao quanh điểm dữ liệu thứ i.

r_a : là một hằng số dương hay còn gọi là bán kính cụm.

$\|.\|$: khoảng cách Euclid giữa điểm dữ liệu thứ i với các điểm bao quanh

Khi mật độ của tất cả các điểm dữ liệu đã được tính, lựa chọn điểm có mật độ lớn nhất làm tâm cụm thứ nhất. Gọi x_1^* là vị trí tâm cụm đầu tiên, có mật độ là P_1^* thì $P_1^* = \max_{i=1}^{n} P_i$.

Tính lại mật độ cho các điểm dữ liệu theo công thức:

$$P_i = P_i - P_1^* e^{-\frac{4}{r_b^2}\|x_i - x_1^*\|^2}; i = 1,...,n \qquad (2.2)$$

Và r_b thường được chọn là $r_b = 1.5 r_a$ và tiếp tục chọn điểm có mật độ lớn nhất làm tâm cụm thứ 2.

Trong trường hợp tổng quát khi đã có k tâm cụm thì mật độ của các điểm dữ liệu còn lại được tính theo công thức:

$$P_i = P_i - P_k^* e^{-\frac{4}{r_b^2}\|x_i - x_k^*\|^2}; i = 1,...,n \qquad (2.3)$$

Sử dụng 2 điểm cận với cận dưới $\underline{e} * P^{ref}$ và cận trên $\overline{e} * P^{ref}$, với P^{ref} là mật độ của tâm cụm thứ *k*, trong đó $\overline{e}$ và $\underline{e}$ lần lượt được gọi là hằng số chấp nhận và hằng số từ chối, thường được chọn lần lượt là 0.5 và 0.15. Một tâm cụm mới được chọn nếu điểm đó có mật độ lớn hơn cận trên. Nếu điểm có mật độ lớn nhất nhỏ hơn cận dưới thì thuật toán dừng.

Nếu điểm có mật độ lớn nhất nằm giữa hai cận thì khoảng cách giữa điểm đó với các tâm cụm đã được xác định trước đó sẽ quyết định xem điểm đó có trở thành tâm cụm mới hay không.

Các bước thực hiện thuật toán phân cụm trừ mờ như sau:

Bước 1: Khởi tạo r_a, h với $h = \frac{r_b}{r_a}$, $\overline{e}$ và $\underline{e}$.

Bước 2: Tính mật độ cho các điểm dữ liệu theo công thức (2.1). Chọn điểm có mật độ lớn nhất làm tâm cụm đầu tiên: $P_k^* = \max_{i=1}^{n} P_i$ với $k = 1$ và P_k^* là mật độ của tâm cụm thứ nhất .

Bước 3: Tính toán lại mật độ cho các điểm dữ liệu còn lại theo công thức (2.2).

Bước 4: Gọi x^* là điểm có mật độ lớn nhất là P^*.

- Nếu $P^* > \overline{e} P^{ref}$: x^* là một tâm cụm mới và tiếp tục bước 3.

- Ngược lại nếu $P^* < \underline{\varepsilon} P^{ref}$: chuyển sang bước 5

- Gọi d_{min} là khoảng cách nhỏ nhất giữa x^* và các tâm cụm trước đó.

+ Nếu $\frac{d_{\min}}{r_a} + \frac{P^*}{P^{ref}} \geq 1$: x^* là một tâm cụm mới và tiếp tục bước 3.

+ Ngược lại:

Thiết lập $P(x^*) = 0$.

Chọn x^* có mật độ P^* lớn nhất và tiếp tục bước 4.

Bước 5: Đưa ra các cụm kết quả.

Khi đó bậc hay độ thuộc của một điểm đối với một tâm cụm được xác định theo công thức:

$$\mu_{ik} = e^{-\frac{4}{r_a^2}\|x_i - x_k\|^2} \qquad (2.4)$$

2.2.1.Các thuật toán phân cụm phân hoạch

Ý tưởng chính của kỹ thuật này là phân hoạch một tập hợp dữ liệu có n phần tử cho trước thành k nhóm dữ liệu sao mỗi phần tử dữ liệu chỉ thuộc về một nhóm dữ liệu có tối thiểu ít nhất một phần tử dữ liệu. Số các cụm được thiết lập là các đặc trưng được lựa chọn trước. Phương pháp này là tốt cho việc tìm các cụm hình cầu trong không gian Euclidean. Ngoài ra, phương pháp này cũng phụ thuộc vào khoảng cách cơ bản giữa các điểm để lựa chọn các điểm dữ liệu nào có quan hệ là gần nhau với mỗi điểm khác và các điểm dữ liệu nào không có quan hệ hoặc có quan hệ là xa nhau so với mỗi điểm khác. Tuy nhiên, phương pháp này không thể xử lý các cụm có hình dạng kỳ quặc hoặc các cụm có mật độ các điểm dầy đặc. Các thuật toán phân hoạch dữ liệu có độ phức tạp rất lớn khi xác định nghiệm tối ưu toán cục cho vấn đề PCDL, do nó phải tìm kiếm tất cả các cách phân hoạch có thể được. Chính vì vậy, trên thực tế thường đi tìm giải pháp tối ưu cục bộ cho vấn đề này bằng cách sử dụng một hàm tiêu chuẩn để đánh giá chất lượng của cụm cũng như để hướng dẫn cho quá trình tìm kiếm phân hoạch dữ liệu. Với chiến lược này, thông thường bắt đầu khởi tạo một

phân hoạch ban đầu cho tập dữ liệu theo phép ngẫu nhiên hoặc Heuristic và liên tục tinh chỉnh nó cho đến khi thu được một phân hoạch mong muốn, thỏa mãn ràng buộc cho trước. Các thuật toán phân cụm phân hoạch cố gắng cải tiến tiêu chuẩn phân cụm, bằng cách tính các giá trị đo độ tương tự giữa các đối tượng dữ liệu và sắp xếp các giá trị này, sau đó thuật toán lựa chọn một giá trị trong dẫy sắp xếp sao cho hàm tiêu chuẩn đạt giá trị tối thiểu. Như vậy, ý tưởng chính của thuật toán phân cụm phân hoạch tối ưu cục bộ là sử dụng chiến lược ăn tham (Greedy) để tìm kiểm nghiệm.

Lớp các thuật toán phân cụm phân hoạch bao gồm các thuật toán đề xuất đầu tiên trong lĩnh vực KPDL cũng là thuật toán được áp dụng nhiều trong thực tế như k-means, k-medoids, PAM, CLARA, CLARANS, ...

Thuật toán K-means là một trong những thuật toán phổ biến nhất. Nó căn cứ vào khoảng cách giữa các đối tượng để phân cụm. Các đối tượng được xếp vào một cụm dựa trên khoảng cách từ chúng tới tâm cụm. Trong thuật toán này, chúng ta chọn một giá trị cho k (số các cụm mong muốn), sau đó chọn ngẫu nhiên k đối tượng làm k cụm ban đầu. Tiếp theo ta tính toán khoảng cách giữa từng đối tượng với k cụm này. Căn cứ vào khoảng cách tính được để xếp từng đối tượng vào cụm thích hợp. Sau khi phân cụm, ta lại tìm tâm mới cho từng cụm. Quá trình này được lặp lại cho đến khi tâm các cụm ổn định. Thuật toán này có một vài phiên bản, phân biệt với nhau bằng hàm tính khoảng cách. Thuật toán K-means thích hợp với các cụm dữ liệu có dạng hình cầu và tròn. Tuy nhiên, K-means tỏ ra rất nhạy cảm với nhiễu và các phần tử ngoại lai.

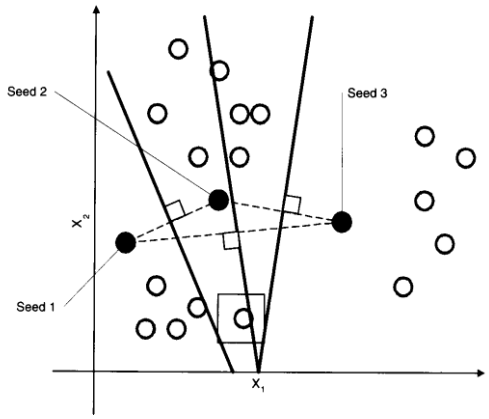
Thuật toán tiếp theo là K-medoids. Thuật toán này sử dụng phương pháp khác so với thuật toán K-means để tính trọng tâm của cụm, nhằm khắc phục ảnh hưởng của nhiễu và các phần tử ngoại lai. Thuật toán này dùng đối tượng nằm ở vị trí trung tâm nhất của cụm làm trung tâm. Phần tử này gọi là medoid của cụm dod. Mỗi khi một cụm được bổ sung một phần tử mới, một medoid được lựa chọn dựa trên các hàm chi phí để đảm bảo rằng chất lượng phân cụm luôn được cải thiện. Cách tiếp cận này giúp K-medoid giảm nhẹ ảnh hưởng của nhiễu và các phần tử ngoại lai, nhưng cũng làm tăng thời gian tính toán so với K-means.

Một biến thể khác của K-medoids là PAM (Partitioning Around Medoids), trong đó việc lựa chọn phần tử medoid phải thỏa mãn điều kiện sai số bình phương là nhỏ nhất. Chất lượng phân cụm của PAM khá tốt, nhưng thời gian thực hiện lâu hơn so với K-means và K-medoids. Tuy nhiên, PAM tỏ ra không thích hợp đối với tập dữ liệu lớn.

Do các thuật toán trên không xử lý được các tập dữ liệu lớn, người ta đã đề xuất thuật toán CLARA (Clustering LARge Applications) và CLARANS (Clustering LARge Applications based upon RANdomize Search). Lý do để các thuật toán này xử lý được tập dữ liệu lớn đó là chúng chỉ lấy một phần dữ liệu (gọi là trích mẫu) để xử lý. Những mẫu này sẽ đại diện cho cả tập dữ liệu lớn cần xét. Việc xử lý trên tập mẫu gần giống với PAM. CLARANS có điểm khác với CLARA là nó không phụ thuộc hoàn toàn vào một mẫu như CLARA. CLARANS trích mẫu sau mỗi lần lặp trong suốt quá trình thực hiện. Một vấn đề đặt ra là làm thế nào để đảm bảo việc trích mẫu thỏa mãn điều kiện các phần tử mẫu là đại diện cho toàn bộ tập dữ liệu. Đến nay đây vẫn là vấn đề được nhiều nhà khoa học máy tính tìm hiểu.

- Thuật toán k-means

Thuật toán phân cụm K-means do MacQueen đề xuất lĩnh vực thống kê năm 1967, K-means là thuật toán phân cụm trong đó các cụm được định nghĩa bởi trọng tâm của các phần tử. Phương pháp này dựa trên độ đo khoảng cách tới giá trị trung bình của các đối tượng dữ liệu trong cụm, nó được xem như là trung tâm của cụm. Như vậy, nó cần khởi tạo một tập trung tâm các trung tâm cụm ban đầu, và thông qua đó nó lặp lại các bước gồm gán mỗi đối tượng tới cụm mà trung tâm gần, và tính toán tại trung tâm của mỗi cụm trên cơ sở gán mới cho các đối tượng. Quá trình lặp này dừng khi các trung tâm hội tụ.

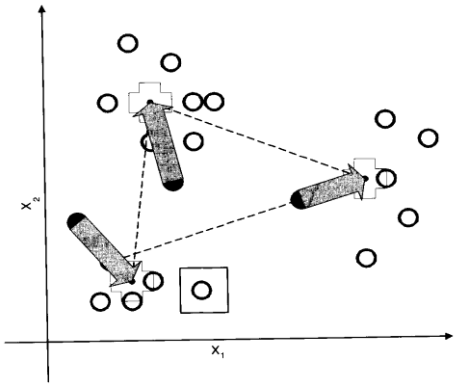


Hinh 2. 2 Các thiết lập để xác định các ranh giới các cụm ban đầu

Trong phương pháp K-means, chọn một giá trị *k* là số cụm cần xác định và sau đó chọn ngẫu nhiên *k* trung tâm của các đối tượng dữ liệu. Tính toán khoảng cách giữa đối tượng dữ liệu và trung bình mỗi cụm để tìm kiếm phần tử nào là tương tự và thêm vào cụm đó. Từ khoảng cách này có thể tính toán trung bình mới của cụm và lặp lại quá trình cho đến khi mỗi các đối tượng dữ liệu là một bộ phận của cụm nào đó.

Mục đích của thuật toán K-means là sinh *k* cụm dữ liệu $\{C_1, C_2,\ldots,C_k\}$ từ một tập dữ liệu ban đầu chứa *n* đối tượng trong không gian d chiều $X_i=\{X_{i1}, X_{i2},\ldots, X_{in}\}$, $i = \overline{1,n}$, sao cho hàm tiêu chuẩn: $E=\sum_{i=1}^{k} \sum_{x\in Ci} D^2(x-m_i)$ đạt giá trị tối thiểu.

Trong đó: m_i là trọng tâm của cụm C_i, D là khoảng cách giữa hai đối tượng.



Hinh 2. 3 Tính toán trọng tâm của các cụm mới

Trọng tâm của một cụm là một vectơ, trong đó giá trị của mỗi phần tử của nó là trung bình cộng của các thành phần tương ứng của các đối tượng vectơ dữ liệu

trong cụm đang xét. Tham số đầu vào của thuật toán là số cụm *k*, và tham số đầu ra của thuật toán là các trọng tâm của cụm dữ liệu. Độ đo khoảng cách D giữa các đối tượng dữ liệu thường được sử dụng là khoảng cách Euclide vì đây là mô hình khoảng cách nên dễ lấy đạo hàm và xác định các cực trị tối thiểu. Hàm tiêu chuẩn và độ đo khoảng cách có thể được xác định cụ thể hơn tùy vào ứng dụng hoặc quan điểm của người dùng. Thuật toán K-means bao gồm các bước cơ bản trong Hình 2.4

Input: Tập dữ liệu S và số cụm mong muốn *k*

Output: Tập các cụm $C_i (1 \leq i \leq k)$ và hàm tiêu chẩn E đạt giá trị tối thiểu.

Begin

Bước 1: Khởi tạo

Chọn *k* trọng tâm $\{m_j\} (1 \leq i \leq k)$ ban đầu trong không gian Rd (d là số chiều của dữ liệu). Việc lựa chọn nay có thể là ngẫu nhiên hoặc theo kinh nghiệm.

Bước 2: Tính toán khoảng cách

Đối với một điểm X_i $(1 \leq i \leq n)$, tính toán khoảng cách của nó tới mỗi trọng tâm m_j $(1 \leq j \leq k)$. Sau đó tìm trọng tâm gần nhất đối với mỗi đối tượng

Bước 3: Cập nhật lại trọng tâm

Đối với mỗi $1 \leq j$ k, cập nhật trọng tâm cụm m_j bằng cách xác định trung bình cộng các vectơ đối tượng dữ liệu.

Bước 4: Điều kiện dừng

Lặp các bước 2 và 3 cho đến khi các trọng tâm của cụm không thay đổi.

End.

Hinh 2. 4 Các bước thực hiện thuật toán K- means

Thuật toán K-means biểu diễn các cụm bởi các trọng tâm của các đối tượng trong cụm đó. Thuật toán K-means chi tiết được trình bày trong hình 2.5:

BEGIN

Nhập n đối tượng dữ liệu

Nhập k cụm dữ liệu

MSE $= +\infty$

For i = 1 to k do $m_i = X_{i+(i-1)*[n/k]}$*; // khởi tạo k trọng tâm*

Do {

OldMSE = MSE;

MSE' = 0;

For j = 1 to k do

{m'[j] = 0; n'[j] =0}

Endfor

For i = 1 to n do

For j =1 to k do

Tính khoảng cách Euclidean bình phương: $D^2(x[i]; m[j])$

Endfor

Tìm trọng tâm gần nhất m[h] tới X[i]

m'[h] = m'[h] + X[i]; n'[h] = n'[h] +1;

MSE' = MSE' + D^2(X[i]; m[j];)

Endfor

n[j] = max(n'[j], 1); m[j] = m' [j]/n[j] ;

MSE = MSE'

} While(MSE < OldMSE)

END.

Hinh 2. 5 Thuật toán K-means chi tiết

Các khái niệm biến và hàm sử dụng trong thuật toán *K*- means trong hình 2.5 như sau:

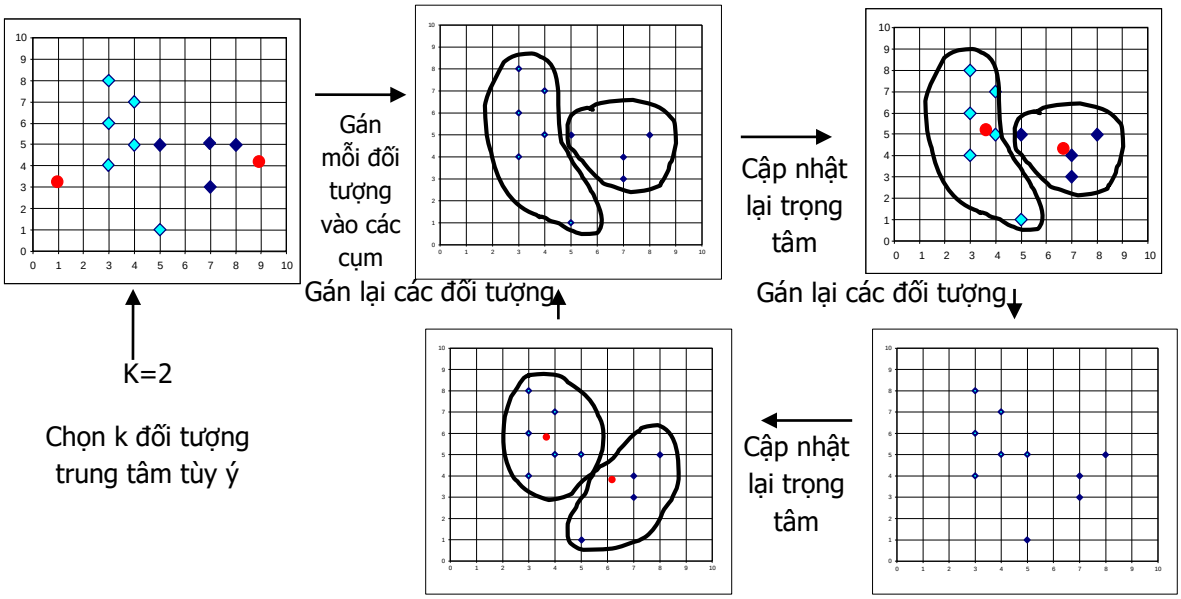
MSE (Mean Squared Error); được gọi là sai số bình phương trung bình hay còn gọi là hàm tiêu chuẩn. MSE dùng để lưu giá trị của hàm tiêu chuẩn và được cập nhật qua mỗi lần lặp. Thuật toán dừng ngay khi giá trị MSE tăng lên so với giá trị MSE cũ của vòng lặp trước đó;

$D^2(x_i, m_j)$; là khoảng cách Euclide từ đối tượng dữ liệu thứ *i* tới trọng tâm *j*;

OldMSE, m'[*j*], n'[*j*]; Là các biến tạm lưu giá trị cho trạng thái trung gian cho các biến tương ứng: giá trị hàm tiêu chuẩn, giá trị của vectơ tổng của các đối tượng trong cụm thứ *j*, số các đối tượng của cụm thứ *j*.

Thuật toán K-means tuần tự trên được chứng minh là hội tụ và có độ phức tạp tính toán là $O((3nkd)\tau T^{flop})$ [10][16][20]. Trong đó, *n* là số đối tượng dữ liệu, *k* là số cụm dữ liệu, *d* là số chiều, τ là số vòng lặp, T^{flop} là thời gian để thực hiện một phép tính cơ sở như phép tính nhân, chia…Trong khi tiến hành, một vấn đề làm sao gỡ các nút thắt trong các trường hợp ở đó có nhiều trung tâm với cùng khoảng cách tới một đối tượng. Trong trường hợp này, có thể gán các đối tượng ngẫu nhiên cho một trong các cụm thích hợp hoặc xáo trộn các đối tượng để vị trí mới của nó không gây ra các nút thắt. Như vậy, do K- means phân tích phân cụm đơn giản nên có thể áp dụng đối với tập dữ liệu lớn. Tuy nhiên, nhược điểm của K-means là chỉ áp dụng với dữ liệu có thuộc tinh số và khám phá các cụm có dạng hình cầu, K-means còn rất nhạy cảm với nhiễu và các phần tử ngoại lai trong dữ liệu.

Hình 2.6 dưới đây mô phỏng về một số hình dạng cụm dữ liệu được khám phá bởi K-means:



Hinh 2. 6 Ví dụ về một số hình dạng cụm dữ liệu được khám phá bởi K-means

Hơn nữa, chất lượng PCDL của thuật toán K-means phụ thuộc nhiều vào các tham số đầu vào như: số cụm *k* và *k* trong tâm khởi tạo ban đầu. Trong trường hợp các trọng tâm khởi tạo ban đầu mà quá chênh lệch so với trong tâm của cụm tự nhiên thì kết quả phân cụm của K-means là rất thấp, nghĩa là các cụm dữ liệu được khám phá rất lệch so với các cụm thực tế. Trên thực tế chưa có một giải pháp tối ưu nào để chọn các tham số đầu vào, giải pháp thường được sử dụng nhất là thử nghiệm với các giá trị đầu vào *k* khác nhau rồi sau đó chon giải pháp tốt nhất.

- **Ngoài thuật toán K-means ra,** phân cụm phân hoạch còn bao gồm một số các thuật toán khác như: Thuật toán PAM; Thuật toán CLARA; Thuật toán CLARANS.

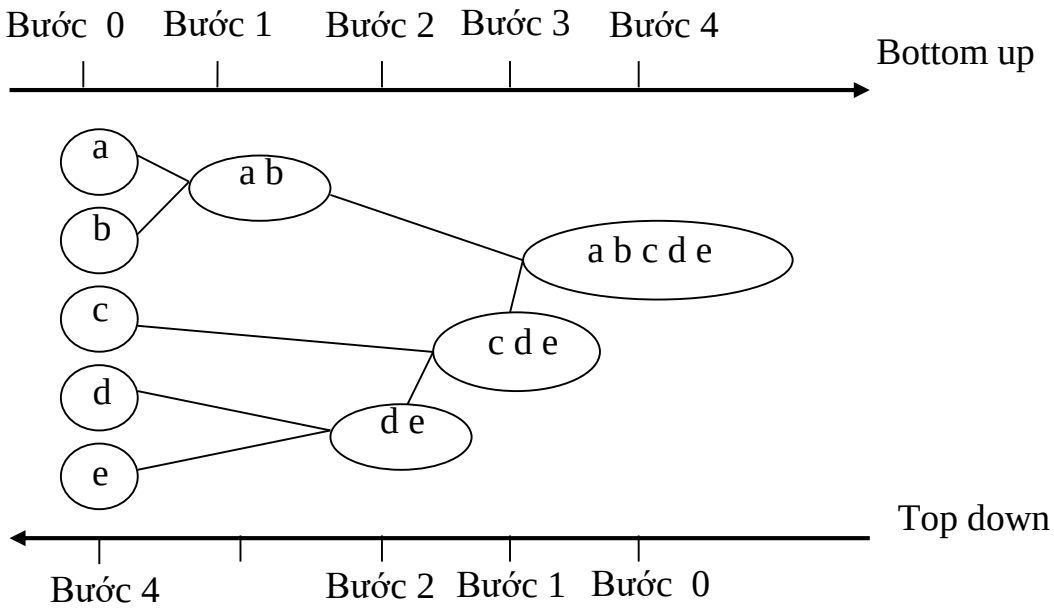
2.2.2 Các thuật toán phân cụm phân cấp

Phương pháp này xây dựng một phân cấp trên cơ sở các đối tượng dữ liệu đang xem xét. Nghĩa là sắp xếp một tập dữ liệu đã cho thành một cấu trúc có dạng hình cây, cây phân cấp này được xây dựng theo kỹ thuật đệ quy. Cây phân cụm có thể được xây dựng theo hai phương pháp sau: hòa nhập nhóm, thường được gọi là tiếp cận Bottom-Up và phân chia nhóm, thường được gọi là tiếp cận Top-Down.

Phương pháp Bottom-Up: phương pháp này bắt đầu xuất phát với mỗi đối tượng dữ liệu được khởi tạo tương ứng với các cụm riêng biệt và sau đó tiến hành nhóm các đối tượng theo một độ đo tương tự (như khoảng cách giữa hai trung tâm của hai nhóm), quá trình này được thực hiện cho đến khi tất cả các nhóm được hòa nhập vào một nhóm (mức cao nhất của cây phân cấp) hoặc cho đến khi các điều kiện kết thúc thỏa mãn. Như vậy, cách tiếp cận này sử dụng chiến lược ăn tham trong quá trình phân cụm.

Phương pháp Top-Down: Bắt đầu với trạng thái là tất cả các đối tượng dữ liệu được sắp xếp trong cùng một cụm và phương pháp này tiến hành chia nhỏ các cụm. Mỗi vòng lặp thành công, một cụm được tách ra thành các cụm nhỏ hơn theo giá trị của một phép đo tương tự nào đó cho đến khi mỗi đối tượng dữ liệu là một cụm riêng biệt hoặc cho đến khi điều kiện dừng thỏa mãn. Cách tiếp cận này sử dụng chiến lược chia để trị trong quá trình phân cụm.

Sau đây là minh họa chiến lược phân cụm phân cấp Bottom up và Top down:

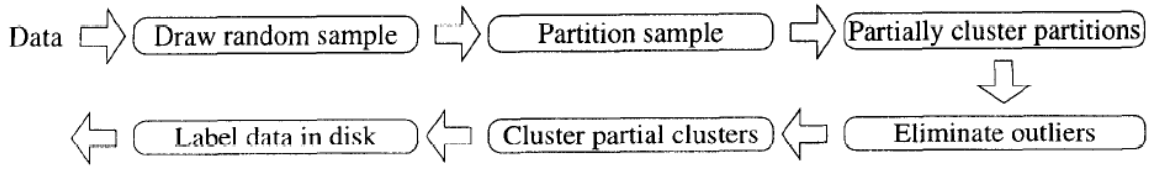


Hinh 2. 7Các chiến lược phân cụm phân cấp

Trong thực tê áp dụng, có nhiêu trường hợp người ta kêt hợp cả hai phương pháp phân cụm phân hoạch và phân cụm phân cấp, nghĩa là kết quả thu được của phương pháp phân cấp có thể cải tiến thông qua bước phân cụm phân hoạch. Phân cụm phân hoạch và phân cụm phân cấp là hai phương pháp PCDL cổ điển, hiện đã có rất nhiều thuật toán cải tiến dựa trên hai phương pháp này đã được áp dụng phổ biến trong KPDL. Phương pháp này bao gồm các thuật toán AGNES, DIANA, BIRCH, CURE, ROCK, Chemeleon,...

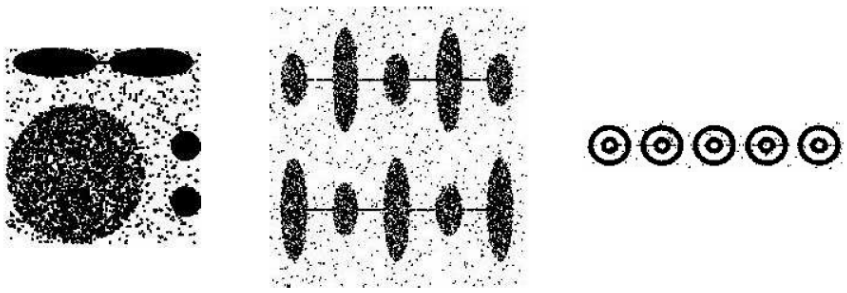
Thuật toán CURE

Thuật toán CURE (**C**lustering **U**sing **RE**presentatives) là thuật toán sử dụng chiến lược Bottom up của kỹ thuật phân cụm phân cấp. Trong khi hầu hết các thuật toán thực hiện phân cụm với các cụm hình cầu và kích thước tương tự, như vậy là không hiệu quả khi xuất hiện các phần tử ngoại lai. Thuật toán CURE khắc phục được vấn đề này và tốt hơn với các phân tử ngoại lai. Thuật toán này định nghĩa một số cố định các điểm đại diện nằm rải rác trong toàn bộ không gian dữ liệu và được chọn để mô tả các cụm được hình thành. Các điểm này được tạo ra bởi trước hết lựa chọn các đối tượng nằm rải rác cho cụm và sau đó "co lại" hoặc di chuyển chúng về trung tâm cụm bằng nhân tố co cụm. Quá trình này được lặp lại và như vậy trong quá trình này, có thể đo tỉ lệ gia tăng của cụm. Tại mỗi bước của thuật toán, hai cụm có cặp các điểm đại diện gần nhau (mỗi điểm trong cặp thuộc về mỗi cụm khác nhau) được hoà nhập.



Hinh 2. 8 Khái quát thuật toán CURE

Như vậy, có nhiều hơn một điểm đại diện mỗi cụm cho phép CURE khám phá được các cụm có hình dạng không phải hình cầu. Việc co lại các cụm có tác dụng làm giảm tác động của các phần tử ngoại lai. Như vậy, thuật toán này có khả năng xử lý tốt trong các trường hợp có các phân tử ngoại lai và làm cho nó hiệu quả với những hình dạng không phải là hình cầu và kích thước độ rộng biến đổi. Hơn nữa, nó tỉ lệ tốt với CSDL lớn mà không làm giảm chất lượng phân cụm.
Hình 2.9 dưới đây là ví dụ về quá trình xử lý của CURE.



Hinh 2. 9 Các cụm dữ liệu được khám phá bởi CURE

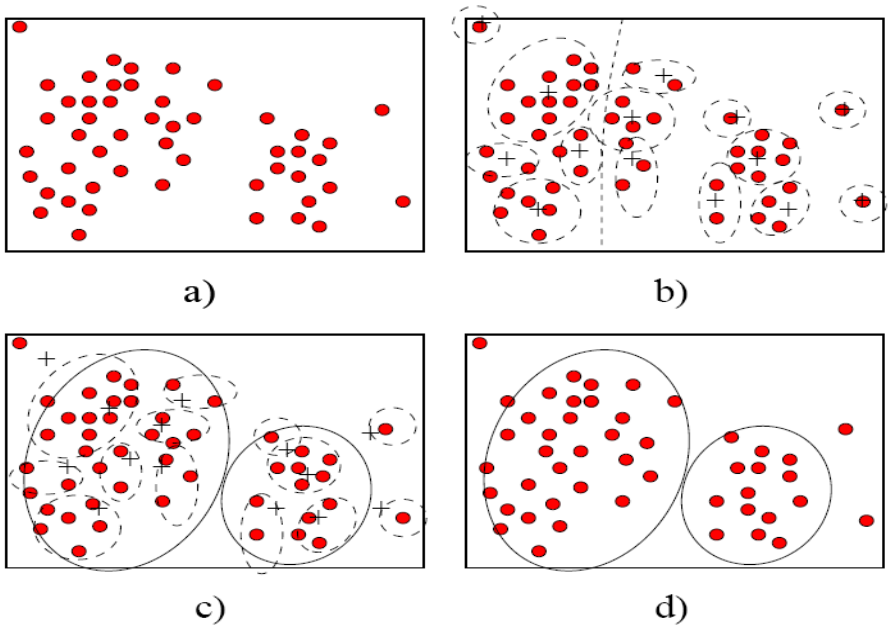
Để xử lý được các CSDL lớn, CURE sử dụng mẫu ngẫu nhiên và phân hoạch, một mẫu là được xác định ngẫu nhiên trước khi được phân hoạch, và sau đó tiến hành phân cụm trên mỗi phân hoạch, như vậy trên mỗi phân hoạch là từng phần đã được phân cụm, quá trình này lặp lại cho đến khi ta thu được phân hoạch đủ tốt. Các cụm thu được lại được phân cụm lần thứ hai để thu được các cụm con mong muốn, nhưng mẫu ngẫu nhiên không nhất thiết đưa ra một mô tả tốt cho toàn bộ tập dữ liệu.

Độ phức tạp của thuật toán CURE là O ($n^2$1og(n)). CURE là thuật toán tin cậy trong việc khám phá ra các cụm với hình dạng bất kỳ và có thể áp dụng tốt đối với dữ liệu có phần tử ngoại lai, và trên các tập dữ liệu hai chiều. Tuy nhiên, nó lại rất nhạy cảm với các tham số như số các đối tượng đại diện, tỉ lệ của các phần tử đại diện.
Thuật toán CURE được thực hiện qua các bước cơ bản như hình 2.10 sau:

1. Chọn một mẫu ngẫu nhiên từ tập dữ liệu ban đầu.
2. Phân hoạch mẫu này thành nhiều nhóm dữ liệu có kích thước bằng nhau: Ý tưởng chính ở đây là phân hoạch mẫu thành p nhóm dữ liệu bằng nhau, kích thước của mỗi phân hoạch là n'/p (n' là kích thước của mẫu).
3. Phân cụm các điểm của mỗi nhóm: Thực hiện PCDL cho các nhóm cho đến khi được phân thành n'/(pq) cụm (với q > 1).
4. Loại bỏ các phân tử ngoại lai: Trước hết, khi các cụm được hình thành cho đến khi số các cụm giảm xuống một phần so với số các cụm ban đầu. Sau đó, trong trường hợp các phân tử ngoại lai được lấy mẫu cùng với quá trình pha khởi tạo mẫu dữ liệu, thuật toán sẽ tự động loại bỏ các nhóm nhỏ.
5. Phân cụm các cụm không gian: Các đối tượng đại diện cho các cụm di chuyển về hướng trung tâm cụm, nghĩa là chúng được thay thế bởi các đối tượng gần trung tâm hơn.
6. Đánh dấu dữ liệu với các nhãn tương ứng.

Hinh 2. 10 Các bước thực hiện cơ bản của thuật toán CURE

Hình vẽ dưới đây là một ví dụ về phân cụm sử dụng thuật toán CURE



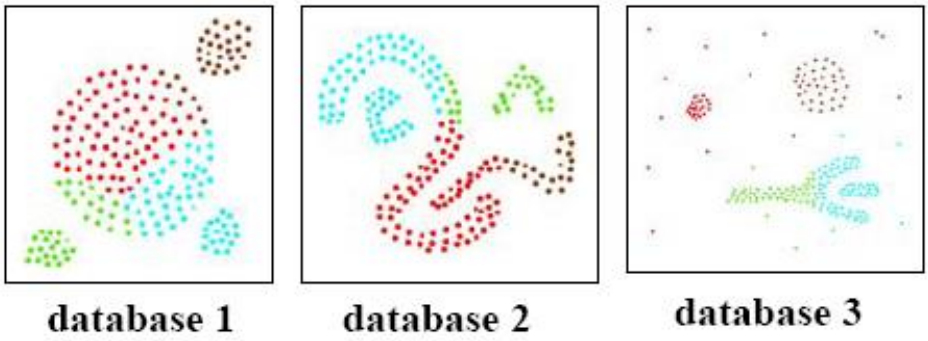
Hinh 2. 11 Ví dụ thực hiện phân cụm bằng thuật toán CURE

- Ngoài thuật toán CURE ra, phân cụm phân cấp còn bao gồm một số thuật toán khac như: Thuật toán BIRCH; Thuật toán AGNES; Thuật toán DIANA; Thuật toán ROCK; Thuật toán CHANMELEON.

2.2.3 Các thuật toán phân cụm dựa trên mật độ.

Phương pháp này nhóm các đối tượng theo hàm mật độ xác định. Mật độ xác định được định nghĩa như là số các đối tượng lân cận của một đối tượng dữ liệu theo một ngưỡng nào đó. Trong cách tiếp cận này, khi một cụm dữ liệu mới miễn là số các đối tượng lân cận của các đối tượng này phải lớn hơn 1 ngưỡng đã được xác định trước. Phương pháp phân cụm dựa vào mật độ của các đối tượng để xác định các cụm dữ liệu và có thể phát hiện ra các cụm dữ liệu với nhiều hình dạng bất kỳ. Tuy vậy, việc xác định các tham số mật độ của thuật toán rất khó khăn, trong khi các tham số này lại có thể tác động rất lớn đến kết quả của PCDL.

Hình 2.12 minh hoạ về các cụm dữ liệu với các hình thù khác nhau dựa trên mật độ được khám phá từ 3CSDL khác nhau.



Hinh 2. 12 Một số hình dạng khám phá bởi phân cụm dựa trên mật độ

Các cụm có thể được xem như các vùng mật độ cao, được tách ra bởi các vùng không có hoặc ít mật độ. Khái niệm mật độ ở đây được xem như là các số các đối tượng láng giềng.

Một thuật toán PCDL dựa trên mật độ điển hình như DBSCAN, OPTICS, DENCLUE, SNN,....

- Thuật toán DBSCAN

DBSCAN (*Density based Spatial Clutering of Application with Noise*) phân cụm dựa trên sự quan sát thực tế thấy rằng, mật độ của những điểm trong cùng một

cụm thì lớn hơn rất nhiều so với mật độ của những điểm không thuộc cụm đó. Từ quan sát đó, DBSCAN thực hiện chia các cụm sao cho mật độ của các đối tượng dữ liệu trong từng cụm lớn hơn một ngưỡng đặt ra.

Thuật toán DBSCAN yêu cầu hai tham số là *Eps* và *minpts* từ người dùng. Tham số *Eps* xác định tập các đối tượng lân cận của một đối tượng dữ liệu. *Minpts* là tham số ngưỡng mật độ của các đối tượng dữ liệu.

Một số khái niệm sử dụng trong DBSCAN:

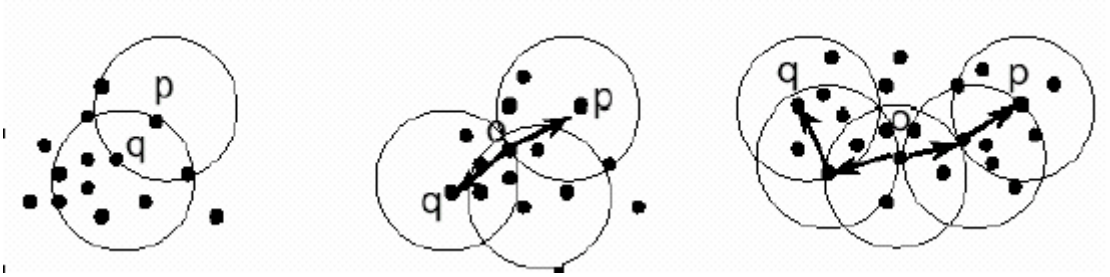
Lân cận với ngưỡng Eps của một điểm: Lân cận với ngưỡng Eps của một điểm p ký hiệu $N_{Eps}(p)$ được xác định như sau: $N_{Eps}(p) = \{q \in \mathbf{D} \mid dis(p,q) \leq Eps\}$

Một điểm dữ liệu p được gọi là điểm nhân (core - point) nếu miền lân cận của p với bán kính Eps có ít nhất là minpt điểm.

q được gọi là đến được theo mật độ trực tiếp (directly density reachble) nếu p là điểm nhân và $q \in$ Neighbor(p, Eps).

q được gọi là đến được theo mật độ (density reachable) từ p nếu có một dãy $p = p0, p, \ldots, p_n = q$ với p_i là đến được theo mật độ trực tiếp từ p_{i+1}.

Một điểm *p* gọi là nối mật độ với q nếu có một điểm 0 mà cả *p* và *q* đều là đến được theo mật độ từ 0.



Hình 2. 13 a) Mật độ trực tiếp, b) Đến được mật độ, c) Mật độ liên thông

Một tập con *C* khác rỗng của *D* được gọi là một cụm (*cluter*) theo *Eps* và *minpts* nếu thoả mãn hai điều kiện:

$\forall p, q \in D$, nếu $p \in C$ và *q* có thể đến được từ *p* theo *Eps* và *Minpts* thì $p \in$ C.

$\forall$p, $q \in$ C, *p* liên thông theo mật độ với *q* theo *Eps* và *Minpts*.

Dữ liệu nhiễu (*noise*): Một điểm dữ liệu nếu không phụ thuộc vào cụm nào thì gọi là nhiễu: *nhiễu* = $\{p \mid \forall i = 1 \ldots k, p \notin c_i\}$.

Để tìm ra các cụm, DBSCAN lần lượt duyệt lại mọi đối tượng thuộc cơ sở dữ liệu và mở rộng đến tất cả những điểm có cùng mật độ có thể đi đến được từ p với hai tham số *Eps* và *minpts*. Nếu đối tượng dữ liệu *p* là đối tượng dữ liệu nhân thì tập các điểm đến được mật độ từ *p* sẽ tạo ra một cụm. Trong trường hợp ngược lại, duyệt đến đối tượng dữ liệu kế tiếp trong cơ sở dữ liệu cho đến khi tất cả các đối tượng dữ liệu đã được duyệt qua.

Eps và *Minpts* được xác định trước bởi người dùng. *Minpts* thường được đặt bằng 2^n với *n* là đối tượng không gian dữ liệu. *Eps* được xác định bởi người sử dụng trong từng ứng dụng cụ thể. Việc lựa chọn gía trị *Eps* có thể được hỗ trợ bởi đồ thị 2^n – *dist* (đồ thị biểu diễn hàm ánh xạ mỗi một điểm *p* đến khoảng cách của điểm lân cận thứ 2^n của điểm *p*)

DBSCAN được thiết kế để xử lý với dữ liệu có nhiễu và hiệu quả trong việc loại trừ ngoại lai. Mặc dù DBSCAN có thể tìm ra được cụm với hình thù bất kỳ nhưng DBSCAN không thể xác định được cụm với hình dạng lồng nhau. Một điểm yếu của DBSCAN là DBSCAN yêu cầu hai tham biến từ người sử dụng là Eps và Minpts được xác định cố định trên toàn bộ cơ sở dữ liệu nhưng Eps thì được xác định lại sau mỗi lần chạy của DBSCAN.

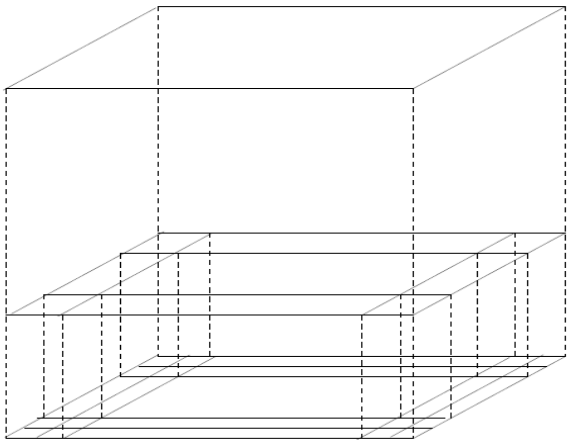
DBSCAN có thể áp dụng với dữ liệu lớn và thứ tự của dữ liệu đầu vào không ảnh hưởng tới kết quả phân cụm. Thời gian chạy của thuật toán là *O(NlogN)*. Tuy nhiên trong thực tế, thời gian để tính toán và dự đoán giá trị *Eps* là khá lớn. DBSCAN không xử lý được với cơ sở dữ liệu nhiều nhiễu.

- **Ngoài thuật toán DBSCAN ra,** phân cụm dựa trên mật độ còn bao gồm 2 thuật toán khác như: Thuật toán OPTICS; Thuật toán DENCLUE.

2.2.4 Các thuật toán phân cụm dựa trên lưới.

Kỹ thuật phân cụm dựa trên mật độ không thích hợp với dữ liệu nhiều chiều, để giải quyết cho đòi hỏi này, người ta đã sử dụng phương pháp phân cụm dựa trên lưới. Đây là phương pháp dựa trên cấu trúc dữ liệu lưới để PCDL, phương pháp này chủ yếu tập trung áp dụng cho lớp dữ liệu không gian. Thí dụ như dữ liệu được biểu diễn dưới dạng cấu trúc hình học của đối tượng trong không gian cùng với các quan

hệ, các thuộc tính, các hoạt động của chúng. Mục tiêu của phương pháp này là lượng tử hoá tập dữ liệu thành các ô (cell), các ô này tạo thành cấu trúc dữ liệu lưới; Sau đó, các thao tác PCDL làm việc với các đối tượng trong từng ô này. Cách tiếp cận dựa trên lưới này không di chuyển các đối tượng trong các ô mà xây dựng nhiều mức phân cấp của nhóm các đối tượng trong một ô. Trong ngữ cảnh này, phương pháp này gần giống phương pháp phân cụm phân cấp nhưng chỉ có điều chúng không trộn các ô. Do vậy, các cụm không dựa trên độ đo khoảng cách (hay còn gọi là độ đo tương tự đối với các dữ liệu không gian) mà nó được quyết định bởi 1 tham số xác định trước. Ưu điểm của phương pháp PCDL dựa trên lưới là thời gian xử lý nhanh và độc lập với số đối tượng dữ liệu trong tập dữ liệu ban đầu, thay vào đó là chúng phụ thuộc vào số ô trong mỗi chiều của không gian lưới. Một thí dụ về cấu trúc dữ liệu lưới chứa các ô trong không gian như hình sau:



Hinh 2. 14 Mô hình cấu trúc dữ liệu lưới

Tầng 1

Mức 1 (mức cao nhất)
có thể chỉ chứa 1 ô

Một số thuật toán PCDL dựa trên cấu trúc lưới điểu hình như STING, Wavecluster, CLIQUE.....

Tầng i-1

Mức 1 (mức cao nhất) có
thể chỉ chứa 1 ô

- Thuật toán STING

STING (STatistical Information Grid) là kỹ thuật phân cụm đa phân giải dựa trên lưới, trong đó vùng không gian dữ liệu được phân rã thành số hữu hạn các cells chữ nhật, điều này có nghĩa là các cells lưới được hình thành từ các cells lưới con để thực hiện phân cụm. Có nhiều mức của các cells chữ nhật tương ứng với các mức khác nhau của phân giải trong cấu trúc lưới, và các cells này hình thành cấu trúc phân cấp: mỗi cells ở mức cao được phân hoạch thành số các cells nhỏ ở mức thấp hơn tiếp theo trong cấu trúc phân cấp. Các điểm dữ liệu được nạp từ CSDL, giá trị của các tham số thống kê gồm: số trung bình - mean, số tối đa - max, số tối thiểu - min, số đếm - count, độ lệch chuẩn - s, …

Các đối tượng dữ liệu lần lượt được chèn vào lưới và các tham số thống kê ở trên được tính trực tiếp thông qua các đối tượng dữ liệu này. Các truy vấn không gian được thực hiện bằng cách xét các cells thích hợp tại mỗi mức của phân cấp. Một truy vấn không gian được xác định như là một thông tin khôi phục lại của dữ liệu không gian và các quan hệ của chúng. STING có khả năng mở rộng cao, nhưng do sử dụng phương pháp đa phân nên nó phụ thuộc chặt chẽ vào trọng tâm của mức thấp nhất. Đa phân giải là khả năng phân rã tập dữ liệu thành các mức chi tiết khác nhau. Khi hoà nhập các cells của cấu trúc lưới để hình thành các cụm, nó không xem xét quan hệ không gian giữa các nút của mức con không được hoà nhập phù hợp (do chúng phải tương ứng với các cha của nó) và hình dạng của các cụm dữ liệu khám phá là isothetic, tất cả danh giới của các cụm có các biên ngang và dọc, theo biên của các cells,và không có đường biên chéo được phát hiện ra.

Thuật toán STING gồm các bước sau:

1. Xác định tầng để bắt đầu.
2. Với mỗi cái của tầng này, tính toán khoảng tin cậy (hoặc ước lượng khoảng) của xác nhận mà cells này liên quan tới truy vấn.
3. Từ khoảng tin cậy của tính toán trên, gán nhãn là cho có liên quan hoặc không liên quan.

4.Nếu lớp này là lớp dưới cùng, chuyển sang bước 6; nếu khác thì chuyển sang bước 5.

5. Duyệt xuống dưới của cấu trúc cây phân cấp một mức. Chuyển sang bước 2 cho các cells mà hình thành các cells liên quan của lớp có mức cao hơn.

6. Nếu đặc tả được câu truy vấn, chuyển sang bước 8, nếu không thì chuyển sang bước 7.

7. Truy lục dữ liệu vào trong các cells liên quan và thực hiện xử lý. Trả lại kết quả thực hiện yêu cầu của truy vấn. Chuyển sang bước 9.

8. Tìm thấy các miền có các cells liên quan.Trả lại miền phù hợp với yêu cầu truy vấn.Chuyển xang bước 9.

9. Dừng.

Hinh 2. 15 Các bước thực hiện thuật toán STING

Các lợi thế của các tiếp cận này so với các phương pháp cụm khác:

- Tính toán dựa trên lưới là truy vấn độc lập vì thông tin thống kê được bảo quản trong mỗi cell đại diện nên chỉ cần thông tin tóm tắt của dữ liệu trong cell lưới chứ không phải dữ liệu thực tế và không phụ thuộc vào câu truy vấn.
- Cấu trúc dữ liệu lưới thuận tiện cho quá trình xử lí song song và cập nhật liên tục.
- Duyệt toàn bộ CSDL cho một lần để tính toán các đại lượng thống kê cho mỗi cells, nên nó rất hiệu quả và do đó độ phức tạp thời gian để tạo độ xấp xỉ O(n), trong đó n là tổng số các đối tượng. Sau khi xây dựng cấu trúc phân cấp, thời gian xử lí cho truy vấn là O(g), trong đó g là tổng số cells lưới ở mức thấp(g<<n);

Các hạn chế của thuật toán này:

- Trong khi sử dụng các tiếp cận đa phân giải để thực hiện phân tích cụm chất lượng của phân cụm STING hoàn toàn phụ thuộc vào tính chất hộp ở mức thấp của cấu trúc lưới. Nếu tính chất hộp là mịn, dẫn đến thời gian chi phí ,thời gian xử lí tăng, tính toán trở lên phức tạp và nếu mức dưới cùng là quá thô thì nó có thể làm giảm bớt chất lượng và độ chính xác của phân tích cụm.

- Ngoài thuật toán STING ra**,** phân cụm dựa trên lưới còn có thêm một thuật toán khác là:Thuật toán CLIQUE.

2.2.5. Các thuật toán phân cụm dựa trên mô hình.

Phương pháp này cố gắng khám phá các phép xấp xỉ tốt của các tham số mô hình sao cho khớp với dữ liệu một cách tốt nhất. Chúng có thể sử dụng chiến lược phân cụm phân hoạch hoặc phân cụm phân cấp, dựa trên cấu trúc hoặc mô hình mà chúng giả định về tập dữ liệu và cách chúng hiệu chỉnh các mô hình này để nhận dạng ra các phân hoạch.

Phương pháp phân cụm dựa trên mô hình cố gắng khớp giữa các dữ liệu với mô hình toán học, nó dựa trên giả định rằng dữ liệu được tạo ra bằng hỗn hợp phân phối xác suất cơ bản. Các thuật toán phân cụm dựa trên mô hình có hai cách tiếp cận chính: *mô hình thống kê và mạng nơron.* Phương pháp này gần giống với phương pháp phân cụm dựa trên mật độ, vì chúng phát triển các cụm riêng biệt nhằm cải tiến các mô hình đã được xác định trước đó, nhưng đôi khi nó không bắt đầu với một số cụm cố định và không sử dụng cùng một khái niệm mật độ cho các cụm. Một thuật toán PCDL dựa trên mô hình điển hình như EM, COBWEB

- Thuật toán EM

Thuật toán EM (Expectation Maximization) được nghiên cứu từ năm 1958 bởi Hartley và được nghiên cứu đầy đủ bởi Dempster, Laird và Rubin công bố năm1977. Thuật toán này nhằm tìm ra sự ước lượng về khả năng lớn nhất của các tham số trong mô hình xác suất (Các mô hình phụ thuộc vào các biến tiềm ẩn chưa được quan sát), nó được xem như là thuật toán dựa trên mô hình hoặc là mở rộng của thuật toán K-means .Thật vậy, EM gán các đối tượng cho các cụm đã cho theo xác suất phân phối thành phần của đối tượng đó, phân phối xác suất thường được sử dụng phân phối xác suất Gaussian với mục đích là khám phá lặp các giá trị tốt cho các tham số của nó bằng hàm tiêu chuẩn là hàm logarit khả năng của đối tượng dữ liệu, đây là hàm tốt để mô hình xác suất cho các đối tượng dữ liệu. EM có thể khám phá ra nhiều hình dạng cụm khác nhau, tuy nhiên do thời gian lặp của thuật toán khá mhiều nhằm xác định các tham số tốt lên chi phí tính toán khá cao. Đã có một số cải tiến được đề xuất

cho EM dụa trên các tính chất của dữ liệu: có thể nén, có thể sao lưu trong bộ nhớ và có thể huỷ bỏ.Trong các cải tiến này,các đối tượng bị huỷ bỏ khi biết chắc chắn được nhãn phân cụm của nó, chúng được nén khi không bị loại bỏ và thuộc về một cụm quá lớn trong bộ nhớ và chúng sẽ được lưu lại trong các trường hợp còn lại.

Thuật toán được chia thành 2 bước xử lý: Đánh giá dữ liệu chưa được gán nhãn (bước E) và đánh giá các tham số của mô hình, khả năng lớn nhất có thể xảy ra (bước M).

E: $\mu \to a = \frac{\frac{1}{2}}{\frac{1}{2}+\mu} h, \quad b = \frac{\mu}{\frac{1}{2}+\mu} h$

M: $a,b \to \mu = \frac{a+b}{6(b+c+d)}$

1. Khởi tạo tham số:

$\lambda_0 = \{ \mu_1^{(0)}, \mu_2^{(0)}, \ldots, \mu_k^{(0)}, p_1^{(0)}, p_2^{(0)}, \ldots . p_k^{(0)} \}$

2. Bước E:

$$P(\omega_i | x_k, \lambda_t) = \frac{P(x_k | \omega_j, \lambda_t) P(\omega_j, \lambda_t)}{P(x_k \lambda_t)} = \frac{P(x_k | \omega_i, \lambda_i^{(t)}, \sigma^2) P_i^{(t)}}{\sum_k P(x_k | \omega_j, \lambda_j^{(t)}, \sigma^2) P_j^{(t)}}$$

3. Bước M:

$$\mu_i^{(t+1)} = \frac{\sum_k P(\omega_i | x_k, \lambda_t) x_k}{\sum_k P(\omega_i | x_k, \lambda_t)}$$

$$P_i^{(t+1)} = \frac{\sum_k P(\omega_i | x_k, \lambda_t)}{R}$$

4. Lặp lại bước 2 và 3 cho đến khi đạt được kết quả.

Hinh 2. 16Các bước thực hiện thuật toán EM

- Ngoài thuật toán EM ra, phân cụm dựa trên mô hình còn có thêm một thuật toán khác là: Thuật toán COBWEB.

2.2.6. Các thuật toán phân cụm có dữ liệu ràng buộc

Sự phát triển của PCDL không gian trên CSDL lớn đã cung cấp nhiều công cụ tiện lợi cho phân tích thông tin địa lý, tuy nhiên hầu hết các thuật toán này cung cấp rất ít cách thức cho người dùng để xác định các ràng buộc trong thế giới thực cần phải được thoả mãn trong quá trình phân cụm. Để PCDL không gian hiệu quả hơn, các nghiên cứu bổ sung cần được thực hiện để cung cấp cho người dùng khả năng kết hợp các ràng buộc trong thuật toán phân cụm.

Hiện nay các phương pháp phân cụm trên đã và đang phát triển và áp dụng nhiều trong các lĩnh vực khác nhau và đã có một số nhánh nghiên cứu được phát triển trên cơ sở các phương pháp đó như:

- Phân cụm thống kê: Dựa trên các khái niệm phân tích hệ thống, nhánh nghiên cứu này sử dụng các độ đo tương tự để phân hoạch các đối tượng, nhưng chỉ áp dụng cho các dữ liệu có thuộc tính số.

- Phân cụm khái niệm: Các kỹ thuật phân cụm được phát triển áp dụng cho dữ liệu hạng mục, chúng phân cụm các đối tượng theo các khái niệm mà chúng xử lý.

- Phân cụm mờ: Thông thường mỗi phương pháp PCDL phân một tập dữ liệu ban đầu thành các cụm dữ liệu có tính tự nhiên và mỗi đối tượng dữ liệu chỉ thuộc về một cụm dữ liệu, phương pháp này chỉ phù hợp với việc khám phá ra các cụm có mật độ cao và rời nhau. Tuy nhiên, trong thực tế, các cụm dữ liệu lại có thể chồng lên nhau (một số các đối tượng dữ liệu thuộc về nhiều các cụm khác nhau), người ta đã áp dụng lý thuyết về tập mờ trong PCDL để giải quyết cho trường hợp này, cách thức kết hợp này được gọi là phân cụm mờ. Trong phương pháp phân cụm mờ, độ phụ thuộc của đối tượng dữ liệu x_k tới cụm thứ i (u_{ik}) có giá trị thuộc khoảng [0,1]. Ý tưởng trên đã được giới thiệu bởi Ruspini (1969) và được Dunn áp dụng năm 1973 nhằm xây dựng một phương pháp phân cụm mờ dựa trên tối thiểu hóa hàm tiêu chuẩn. Bezdek (1982) đã tổng quát hóa phương pháp này và xây dựng thành thuật toán phân cụm mờ c-means có sử dụng trọng số mũ.

C-means là thuật toán phân cụm mờ (của k-means). Thuật toán c – means mờ hay còn gọi tắt là thuật toán FCM (Fuzzy c-mens) đã được áp dụng thành công trong giải

quyết một số lớn các bài toán PCDL như trong nhận dạng mẫu, xử lý ảnh, y học, … Tuy nhiên, nhược điểm lớn nhất của thuật toán FCM là nhạy cảm với các nhiễu và phần tử ngoại lai, nghĩa là các trung tâm cụm có thể nằm xa so với trung tâm thực tế của cụm.

Đã có nhiều phương pháp đề xuất để cải tiến cho nhược điểm trên của thuật toán FCM bao gồm: Phân cụm dựa trên xác suất (keller, 1993), phân cụm nhiễu mờ (Dave, 1991), phân cụm dựa trên toán tử L_p Norm (kersten, 1999). Thuật toán ε-Insensitive Fuzzy c-means (ε FCM- không nhạy cảm mờ c-means).

Thuật toán Phân cụm mờ: FCM, εFCM và FCM-Cải tiến

Tóm lại: Các kỹ thuật PCDL trình bày ở trên được sử dụng rỗng rãi trong thực tế, thế nhưng hầu hết chúng chỉ nhằm áp dụng cho tập dữ liệu với cùng một kiểu thuộc tính. Vì vậy việc PCDL trên tập dữ liệu có kiểu hỗn hợp là một vấn đề đặt ra trong KPDL ở giai đoạn hiện nay.

2.3 Các ứng dụng của phân cụm dữ liệu

Phân cụm dữ liệu là một công cụ quan trọng trong một số ứng dụng. Sau đây là một số ứng dụng của nó:

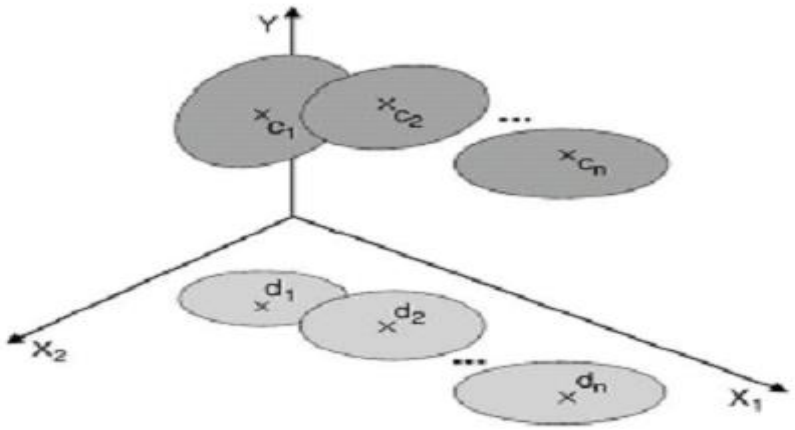
- Giảm dữ liệu
- Rút ra các giả thuyết
- Kiểm định giả thuyết
- Dự đoán dựa trên các cụm
- Thương mại
- Sinh học
- Lập quy hoạch đô thị
- Nghiên cứu trái đất
- Địa lý

-Web Mining

CHƯƠNG 3. XÂY DỰNG HỆ LUẬT MỜ ỨNG DỤNG PHÂN CỤM TRỪ MỜ.

3.1 Xây dựng hệ luật mờ từ dữ liệu vào/ra của hệ thống

Mỗi cụm thể hiện các phần nhất định của hành vi hệ thống, được biểu diễn bằng một luật IF – THEN như vậy ứng với một cụm ta có một luật. Hệ thống có bao nhiêu cụm thì có bấy nhiêu luật. Khi đó, các cụm sẽ được chiếu trên không gian đầu vào (Hình 3.1) để xây dựng luật.



Hinh 3. 1 Luật được hình thành qua phép chiếu vào không gian đầu vào *X*

và khi không gian đầu vào là hai biến luật có dạng:

R_1: If x_1 is A_1^1 and x_2 is A_2^1 then $y_1 = p_0^1 + p_1^1 x_1 + p_2^1 x_2$

..... (3.1)

R_K: If x_1 is A_1^K and x_2 is A_2^K then $y_1 = p_0^K + p_1^K x_1 + p_2^K x_2$

Trong đó: hai biến đầu vào x_1 và x_2 và biến đầu ra y^k (với $k = 1, 2, ..., K$) là chỉ số của luật k. A_1^k và A_2^k (với $k = 1, 2, ..., K$) là các tập mờ của luật thứ k nhận được bằng cách chiếu các cụm vào không gian đầu vào và p_i^k (với $i = 1, 2; k = 1, 2, ..., K$) là các tham số hồi quy kết quả.

Như vậy là từ tập dữ liệu vào ra (x_{1t}, x_{2t}, y_t) với $t = 1, 2, ..., m$.

Ta có thể viết:

$$y_t = \frac{\sum_{k=1}^{K} w_t^k y_t^k}{\sum_{k=1}^{K} w_t^k} = \sum_{k=1}^{K} \beta_t^k y_t^k \qquad (3.2)$$

Trong đó: $w_t^k = (A_1^k(x_{1t}) \wedge A_2^k(x_{2t}))$ và $\beta_t^k = w_t^k / \sum_{k=1}^{K} w_t^k$

Dưới dạng ma trận, đầu ra hệ thống là: $[Y] = [X][P]$

Trong đó ma trận đầu vào *X* được xác định là:

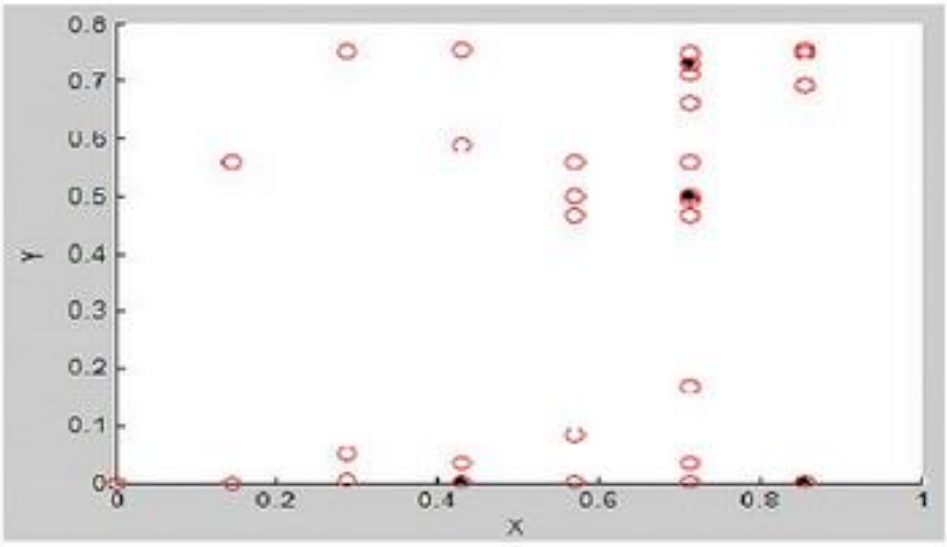
$$X = \begin{bmatrix} \beta_1^1,.., \beta_1^K, \beta_1^1 x_{11},.., \beta_1^K x_{11}, \beta_1^1 x_{21},.., \beta_1^K x_{21} \\ ... \\ \beta_m^1,.., \beta_m^K, \beta_m^1 x_{1m},.., \beta_m^K x_{1m}, \beta_m^1 x_{2m},.., \beta_m^K x_{2m} \end{bmatrix}$$

Với ma trận đầu ra được biết là: $Y = [y_1,..., y_m]^T$

Khi đó ma trận P xác định là theo bình phương cực tiểu:

$$P = \left(X^T X\right) X^T Y \qquad (3.3)$$

Giả sử rằng với 30 điểm dữ liệu được phân cụm trừ thành 4 cụm như hình 3.2 :



Hinh 3. 2 Dữ liệu được phân cụm trừ , tâm cụm là điểm đơn

Ví dụ với 30 điểm dư liệu đầu vào/ra ta có Bảng 3.1

Điểm	**x**	**y**	**Điểm**	**x**	**Y**
1	0	0	**16**	0.58	0.47
2	0.17	0	**17**	0.58	0.5
3	0.29	0	**18**	0.58	0.55
4	0.42	0	**19**	0.71	0.2
5	0.58	0	**20**	0.71	0.47
6	0.71	0	**21**	0.71	0.5
7	0.84	0	**22**	0.71	0.50
8	0.29	0.05	**23**	0.71	0.68
9	0.42	0.03	**24**	0.71	0.7
10	0.58	0.1	**25**	0.71	0.72
11	0.71	0.02	**26**	0.71	0.75
12	0.17	0.55	**27**	0.71	0.45
13	0.29	0.75	**28**	0.83	0.75
14	0.42	0.6	**29**	0.83	0.69
15	0.42	0.75	**30**	0.83	0.75

Bảng 3. 1 Luật mờ được xây dựng từ phân cụm trừ SC

Với $r_a = 1$, theo công thức $\mu_{ik} = e^{-\frac{4}{r_a^2}\|x_i - x_k\|^2}$ xác định độ thuộc của các điểm với tâm các cụm (chia các điểm thành 4 cụm như Bảng 3.2)

Điểm QS	**Cụm 1**	**Cụm 2**	**Cụm 3**	**Cụm 4**
	0.51	0.21	0.17	0.11
	0.54	0.31	0.12	0.03
	0.60	0.24	0.13	0.03
	0.98	0.02	0	0
	0.41	0.35	0.15	0.09
	0.35	0.39	0.15	0.11
	0.42	0.98	0	0

	0.49	0.31	0.12	0.08
	0.94	0.03	0.02	0.01
	0.40	0.36	0.15	0.09
	0.32	0.37	0.17	0.14
	0.25	0.11	0.30	0.34
	0.19	0.14	0.34	0.33
	0.42	0.32	0.18	0.08
	0.41	0.34	0.17	0.08
	0.11	0.17	0.37	0.40
	0.10	0.19	0.40	0.39
	0.11	0.12	0.42	0.35
	0.34	0.42	0.23	0.11
	0.01	0.02	0.94	0.05
	0	0	0.98	0.02
	0.01	0.03	0.81	0.15
	0.03	0.04	0.42	0.51
	0.01	0.01	0.11	0.87
	0	0	0.03	0.97
	0	0	0.02	0.98
	0.02	0.04	0.84	0.10
	0.03	0.07	0.07	0.54
	0.04	0.06	0.06	0.51
	0.03	0.07	0.07	0.54

Bảng 3. 2 Các cụm được xây dựng qua phân cụm trừ

Vậy ta có tâm các cụm có tọa độ tương ứng là:

	Cụm 1	**Cụm 2**	**Cụm 3**	**Cụm 4**
x	0,42	0,84	0,21	0,70
y	0	0	0,55	0,71

Bảng 3. 3 Tọa độ tâm các cụm

Chương trình Matlab sau đây cho ta số lượng các luật, hàm thuộc đầu vào và hàm tuyến tính đầu ra theo hệ mờ TS

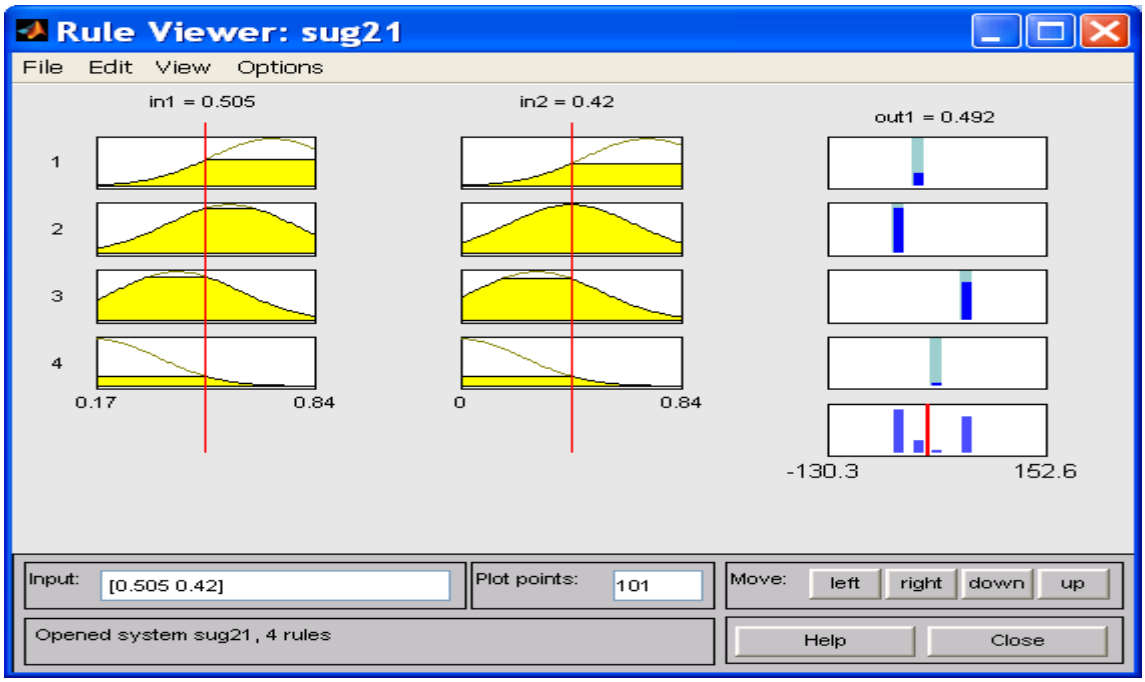
Dữ liệu từ Bảng 3.1

x=[0 0.17 0.29 .42 .58 .71 .84 .29 .42 .58 .71 .17 .29 .42 .42 .58 .58 .58 .71 .71 .71 .71 .71 .71 .71 .71 .71 .71 .71 .83 .83 .83]

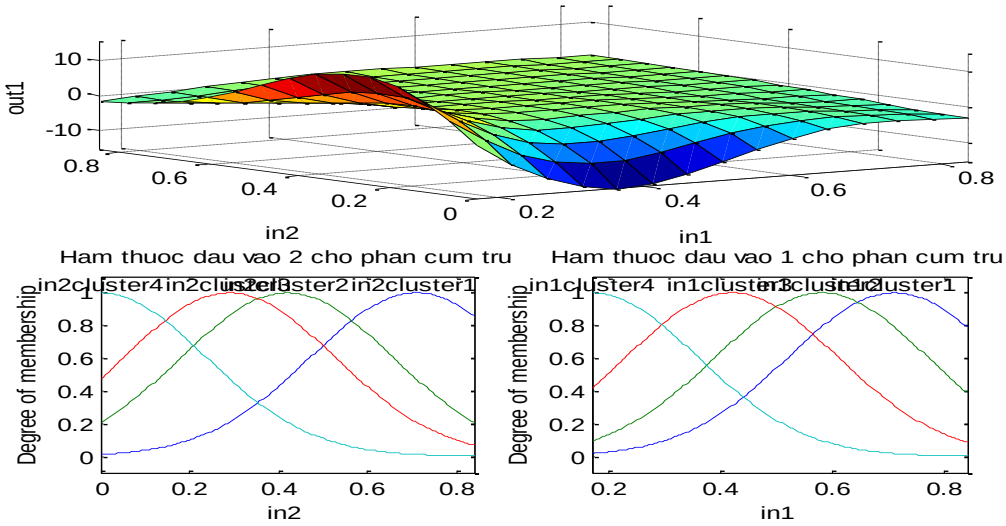
```
y=[0 0 0 0 0 0 0 .05 .03 .1 .02 .55 .75 .6 .75 .47 .5 .55 .2 .47 .5 .5 .68 .7 .72 .75 .45
.75 .69 .75]
trndata=[x(2:30); x(1:29)]';
datout=y(1:29)'
figure
hold on
subplot(2,1,1), plot(trndata);
subplot(2,1,2), plot(datout);
%////////////////////////////////////////////////
%Chuong trinh tu day la SUBSTRACTIVE CLUSTERING /
%////////////////////////////////////////////////
chkdatin=trndata;
fismat=genfis2(trndata,datout,0.8);
fuzout=evalfis(trndata,fismat);
ruleview(fismat)
ruleedit(fismat)
%showrule(fismat)
getfis(fismat,'output',1,'mf',1)
getfis(fismat,'output',1,'mf',2)
figure
subplot(2,1,1);
gensurf(fismat);
%blackbg;
subplot(2,2,4);
%hold on
plotmf(fismat,'input',1);
Title('Ham thuoc dau vao 1 cho phan cum tru')
%subplot(223);
subplot(2,2,3)
```

plotmf(fismat,'input',2);

Title('Ham thuoc dau vao 2 cho phan cum tru')



Hình 3. 3Số lượng luật hình thành qua phan cụn trừ từ Bảng dữ liệu 3.1



Hình 3. 4 Mặt suy diễn và hàm thuộc đầu vào của Bảng dữ liệu 3.1

3.2 Ứng dụng cho bài toán lò nhiệt

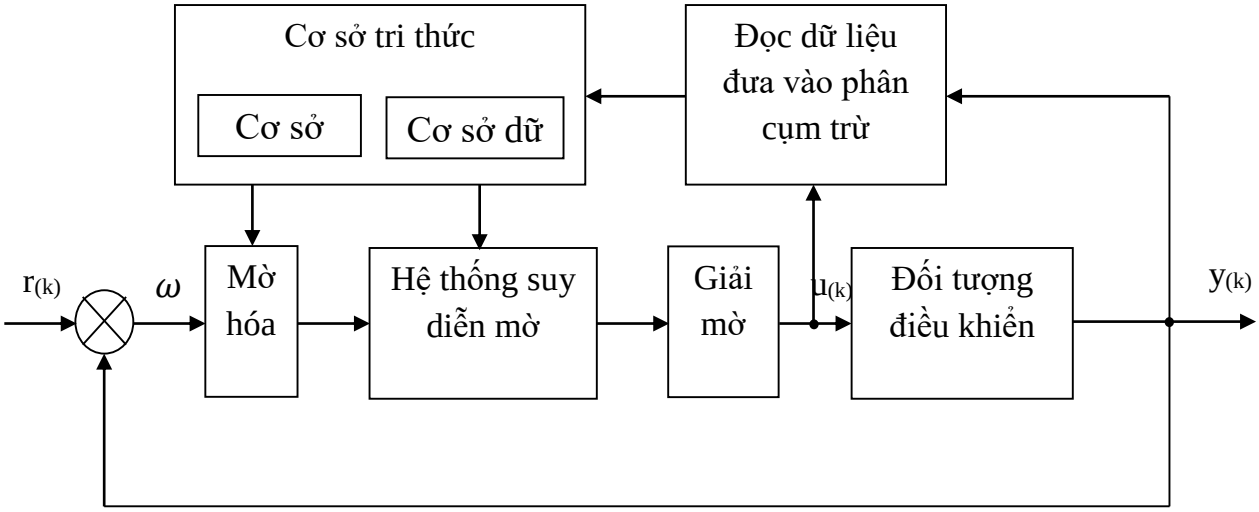
3.2.1 Phát biểu bài toán

Thông thường khi xây dựng hệ điều khiển mờ thì việc đầu tiên cần thiết phải xây dựng luật điều khiển mờ. Luật điều khiển mờ thể hiện dưới dạng luật:

Nếu <*điều kiện*> Thì <*tác động*> (3.4)

Luật điều khiển mờ dạng (3.4) thường được xây dựng từ hệ thức của các chuyên gia điều khiển, những người đã làm việc lâu năm trong ngành điều khiển hoặc những chuyên gia có nhiều kinh nghiệm trong lĩnh vực công nghệ cần điều khiển hoặc hệ luật (3.4) có thể xây dựng từ dữ liệu thu nhận được của các quá trình công nghệ khi cho tín hiệu vào đối tượng với một khoảng giá trị nào đó là sẽ có được các tín hiệu ra. Để xây dựng hệ luật từ dữ liệu này ta phải sử dụng phân cụm trừ từ dữ liệu.

Hệ điều khiển mờ tổng quát điều khiển cho các đối tượng được biểu diễn trên hình 3.5



Hinh 3. 5 Sơ đồ tổng quát hệ điều khiển mờ xây dựng từ dữ liệu

Trên hình 3.5 Bộ điều khiển mờ cho điều khiển qua trinh bao gồm các thành phần sau:

- Cơ sở tri thức
- Phân cụm trừ từ dữ liệu để tạo luật.
- Đối tượng điều khiển
- Thành phần mờ hóa và giải mờ
- Hệ thống suy diễn mờ.

Các tín hiệu trên hình 3.5 bao gồm:

- *r(k):* là tín hiệu đặt ở thời điểm k. Ví dụ, ta muốn đặt nhiệt độ cần là 70^0C chẳng hạn.

- *y(k)*: là tín hiệu ra ở thời điểm k từ đối tượng điều khiển.

- *u(k):* là tín hiệu điều khiển được đưa vào để điều khiển lò nhiệt ở thời điểm *k*.

3.2.2 Mô hình động học của hệ thống lò nhiệt

Một mô hình hệ động học có thể được biểu diễn như sau :

$X_{(k+1)} = f(x_{(k)}, u_{(k)})$ *(3.4)*

Trong đó: $x_{(k)}$ là biến trạng thái

$u_{(k)}$ là biến đầu vào

k là thời điểm ta xét.

Tương tự như vậy, chúng ta có hệ động học vào/ra của một hệ thống được miêu tả như sau :

$y_{(k+1)} = f(y_{(k)}, y_{(k-1)},...y_{(k-ny+1)}, u_{(k)}, u_{(k-1)},...,u_{(k-nu+1)})$ *(3.5)*

Trong đó :

$y_{(k)},..., y_{(k-ny+1)}$ là các biến ra ở các thời điểm k

$u_{(k)},..., u_{(k-ny+1)}$ là các biến vào ở các thời điểm k

Và như vậy hệ mờ thể hiện cho hệ động học (3.5) sẽ có dạng ở luật thứ i như sau :

R_i: If $y_{(k)}$ *is* A_{i1} and $y_{(k-1)}$ *is* A_{i2} ...$y_{(k-ny-1)}$ *is* A_{in} and $u_{(k)}$ *is* B_{i1} and $u_{(k-1)}$ *is* B_{i2} ...$u_{(k-nu-1)}$ *is* B_{in} then $y_{(k+1)}$ *is* C_i (3.6)

Áp dụng cho hệ thống nhiệt, ta có phương trình động học của hệ thống nhiệt [12] như sau:

$y(k+1)=ay(k)+b/(1+exp(0.5y(k)-r))u(k)+(1-a)y_0$ *(3.7)*

với $a=exp(-pTs)$; $b=(q/p)(1-exp(pTs))$

$Ts=25;r=40;y_0=25;p=1.00151*10^{(-4)};q=8.6797*10^{(-3)}$

3.3 Chương trình xử lý bài toàn và mô phỏng.

3.3.1 Thu thập dữ liệu vào ra của hệ thống

Dữ liệu được đưa vào thành phần phân cụm trừ được thu thập từ đầu vào hệ điều khiển là $u_{(k)}$ và đầu ra của nó là $y_{(k)}$. Quá trình thu thập dữ liệu được thực hiện như sau:

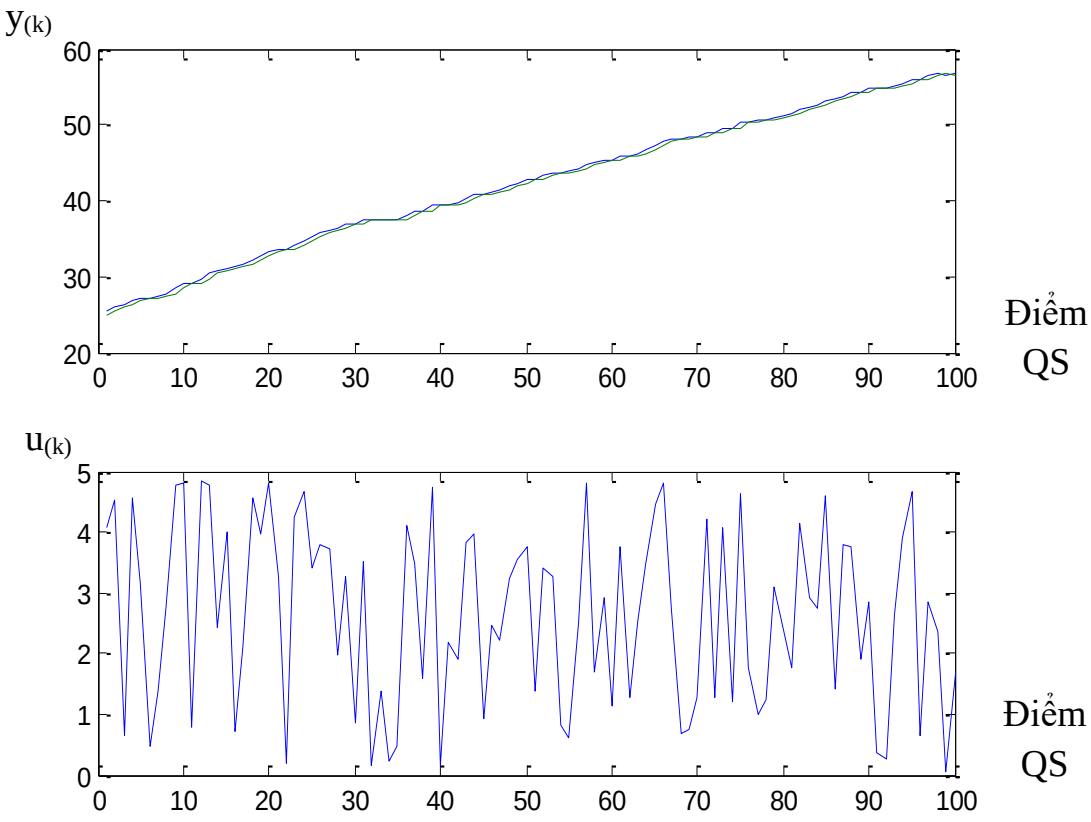
Với phương trình động học của đối tượng điều khiển (3.7), tín hiệu u(k) đầu vào cho chạy từ (0 - 5) ta sẽ được đầu ra y(k) trong 100 điểm quan sát (QS).

Bảng 3. 4 Dữ liệu thu thập từ đầu vào/ra của hệ thống điều khiển lò nhiệt

Điểm QS	u	Y	Điểm QS	u	y	Điểm QS	u	y
1	5.0000	25.0000	35	0.1154	35.0000	68	0.2885	50.0000
2	5.0000	25.6505	36	0.1154	35.0000	69	0.2885	50.0000
3	5.0000	26.3000	37	0.1154	35.0000	70	0.2885	50.0000
4	5.0000	26.9485	38	0.1154	35.0000	71	0.2885	50.0000
5	5.0000	27.5961	39	0.1154	35.0000	72	0.2885	50.0000
6	5.0000	28.2427	40	5.0000	35.0000	73	0.2885	50.0000
7	5.0000	28.8883	41	5.0000	5.6355	74	0.2885	50.0000
8	5.0000	29.5330	42	5.0000	36.2700	75	0.2885	50.0000
9	5.0000	30.1767	43	5.0000	36.9036	76	0.2885	50.0000
10	5.0000	30.8194	44	5.0000	37.5362	77	0.2885	50.0000
11	5.0000	31.4611	45	5.0000	38.1679	78	0.2885	50.0000
12	5.0000	32.1019	46	5.0000	38.7986	79	0.2885	50.0000
13	5.0000	32.7417	47	5.0000	39.4284	80	5.0000	50.0000
14	5.0000	33.3806	48	5.0000	40.0572	81	5.0000	50.6130
15	5.0000	34.0185	49	5.0000	40.6851	82	5.0000	51.2250
16	2.7597	34.6555	50	5.0000	41.3120	83	5.0000	51.8361
17	0.1154	35.0000	51	5.0000	41.9380	84	5.0000	52.4463
18	0.1154	35.0000	52	5.0000	42.5631	85	5.0000	53.0556
19	0.1154	35.0000	53	5.0000	43.1872	86	5.0000	53.6640
20	0.1154	35.0000	54	5.0000	43.8104	87	5.0000	54.2714
21	0.1154	35.0000	55	5.0000	44.4326	88	5.0000	54.8780
22	0.1154	35.0000	56	5.0000	45.0540	89	5.0000	55.4836
23	0.1154	35.0000	57	5.0000	45.6743	90	5.0000	56.0884
24	0.1154	35.0000	58	5.0000	46.2938	91	5.0000	56.6922
25	0.1154	35.0000	59	5.0000	46.9123	92	5.0000	57.2951
26	0.1154	35.0000	60	5.0000	47.5299	93	5.0000	57.8971

27	0.1154	35.0000	61	5.0000	48.1466	94	5.0000	58.4982
28	0.1154	35.0000	62	5.0000	48.7623	95	5.0000	59.0984
29	0.1154	35.0000	63	5.0000	49.3771	96	5.0000	59.6976
30	0.1154	35.0000	64	0.3572	49.9910	97	5.0000	60.2960
31	0.1154	35.0000	65	0.2885	50.0000	98	5.0000	60.8935
32	0.1154	35.0000	66	0.2885	50.0000	99	5.0000	61.4901
33	0.1154	35.0000	67	0.2885	50.0000	100	5.0000	62.0857
34	0.1154	35.0000						

Bảng dữ liệu được thu thập trên có thể biểu diễn trên đồ thị hình 3.5



Hinh 3. 6 Đồ thị biểu diễn số liệu thu thập được ở bảng 3.4

3.3.2 Hệ luật mờ cho điều khiển lò nhiệt từ phân cụm trừ

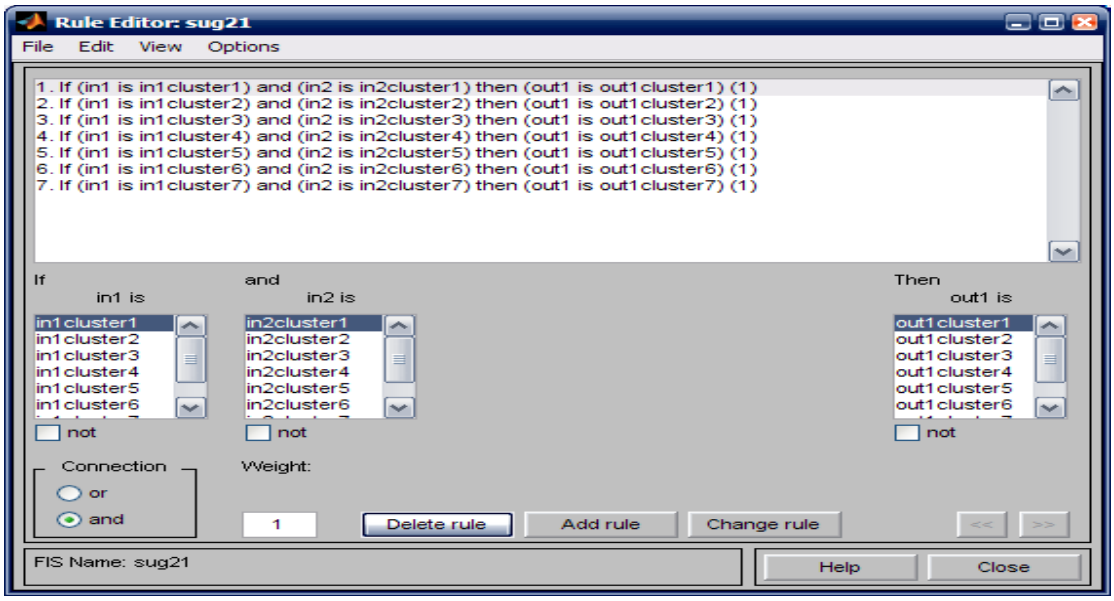
a) Hệ cơ sở luật (hệ cơ sở tri thức)

Hệ cơ sở luật (hệ cơ sở tri thức) thường được thực hiện qua các tham vấn của các chuyên gia điều khiển và chuyên gia công nghệ. Xong trong thực tế người ta có thể xây dựng hệ tri thức thông qua quá trình xử lý dữ liệu như sau:

- Chọn các biến vào ra, cấu trúc của luật, các phương pháp mờ hóa và giải mờ.
- Xác định các giá trị ngôn ngữ như ‘lớn’, ‘nhỏ’, ‘rất lớn’, ‘rất nhỏ’ để từ đó xác định hàm thuộc.
- Tạo dựng hệ luật mờ.

b) Hệ luật mờ cho điều khiển lò nhiệt

Hệ luật điều khiển mờ được tự động tạo ra từ chương trình matlab có dạng như sau:



Hinh 3. 7 Hệ luật mờ hình thành sau khi phân cụm trừ

Từ hình vẽ trên ta thấy:

Luật 1: Nếu đầu vào 1 là cụm 1 và đầu vào 2 là cụm 1 thì đầu ra 1 là ra1 cụm1.

Luật 2: Nếu đầu vào 1 là cụm 2 và đầu vào 2 là cụm 2 thì đầu ra 1 là ra1 cụm2.

Luật 3: Nếu đầu vào 1 là cụm 3 và đầu vào 2 là cụm 3 thì đầu ra 1 là ra1 cụm3.

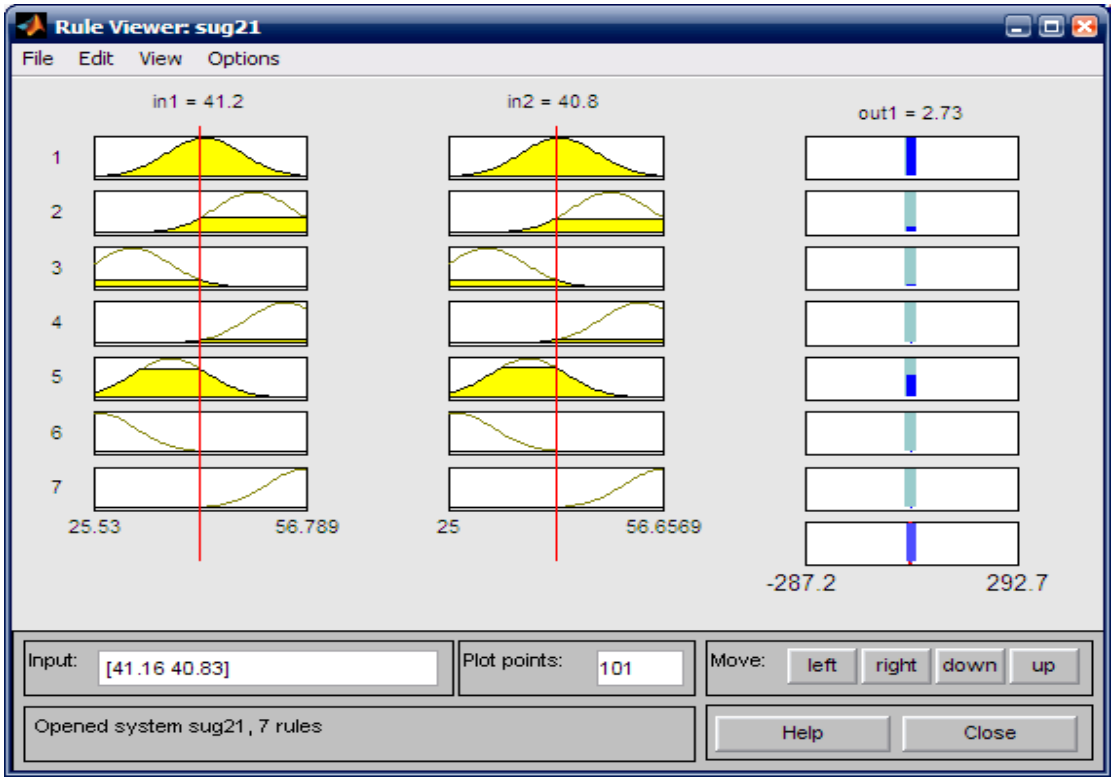
Luật 4: Nếu đầu vào 1 là cụm 4 và đầu vào 2 là cụm 4 thì đầu ra 1 là ra1 cụm4.

Luật 5: Nếu đầu vào 1 là cụm 5 và đầu vào 2 là cụm 5 thì đầu ra 1 là ra1 cụm5.

Luật 6: Nếu đầu vào 1 là cụm 6 và đầu vào 2 là cụm 6 thì đầu ra 1 là ra1 cụm6.

Luật 7: Nếu đầu vào 1 là cụm 7 và đầu vào 2 là cụm 7 thì đầu ra 1 là ra1 cụm7.

Giả thiết hàm thuộc của đầu vào 1 và đầu vào 2 có dạng chuông như hình vẽ, tại đầu vào in1 = 41,2 và in2 = 40,8 thì đầu ra out1 = 2,73 có dạng là phần đồ thị màu xanh.



Hình 3. 8 Hệ luật mờ cho điều khiển nhiệt độ

3.3.3 Hệ suy diễn mờ

Ta đã biết trong mục 2.2, thành phần phân cụm trừ sẽ tạo ra hệ luật mờ dưới dạng Takagi - Sugenco (TS). Dạng luật mô hình TS có dạng như sau:

R_i: *If* x *is* A_i *then* $y_i = f_i(x)$ i =1, 2, ..,k (3.8)

Dạng hàm $f_i(x)$ thường được chọn là hàm tuyến tính...(3.8) có thể viết là:

R_i: *If* x *is* A_i *then* $y_i = a_i^T x + b_i$ $i = 1, 2, .., k$ (3.9)

Trong đó các thông số a_i và b_i là chưa biết

Sử dụng quá trình suy diễn mờ với đầu ra là một tập mờ đơn điệu ta có:

R_i: *If* x *is* A_i *then* $y_i = b_i$ i =1, 2, ..,k (3.10a)

Triển khai (3.10.a) ra ta có thể viết:

R_i: *If* x_1 *is* A_{i1} *and* x_2 *is* A_{i2} *and ... then* $y_i = b_i$ (3.10b)

Từ (3.10.b),quá trình suy diễn mờ của hệ mờ TS có dạng:

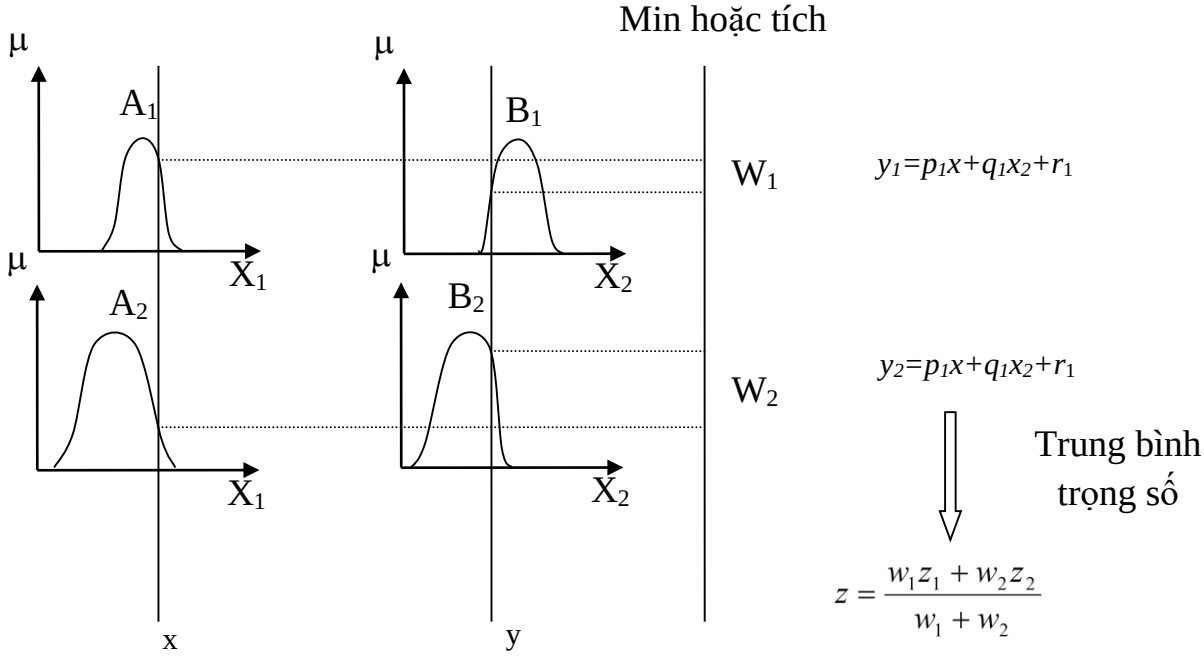
$$y = \frac{\sum_{i=1}^{k} \beta_i y_i}{\sum_{j=1}^{k} \beta_j} = \frac{\sum_{i=1}^{k} \beta_i (a_i^T x + b_i)}{\sum_{j=1}^{k} \beta_j} \qquad (3.11)$$

Và nếu $y_i = b_i$ thì mô hình là đơn điệu.

Trong đó:

$w = \mu_{Ai1}(x_1) \wedge \mu_{Ai2}(x_2) \ldots \wedge \mu_{Aip}(x_p), \qquad 1 \le i \le k$

Giả sử hệ luật TS có 2 luật mờ như sau :

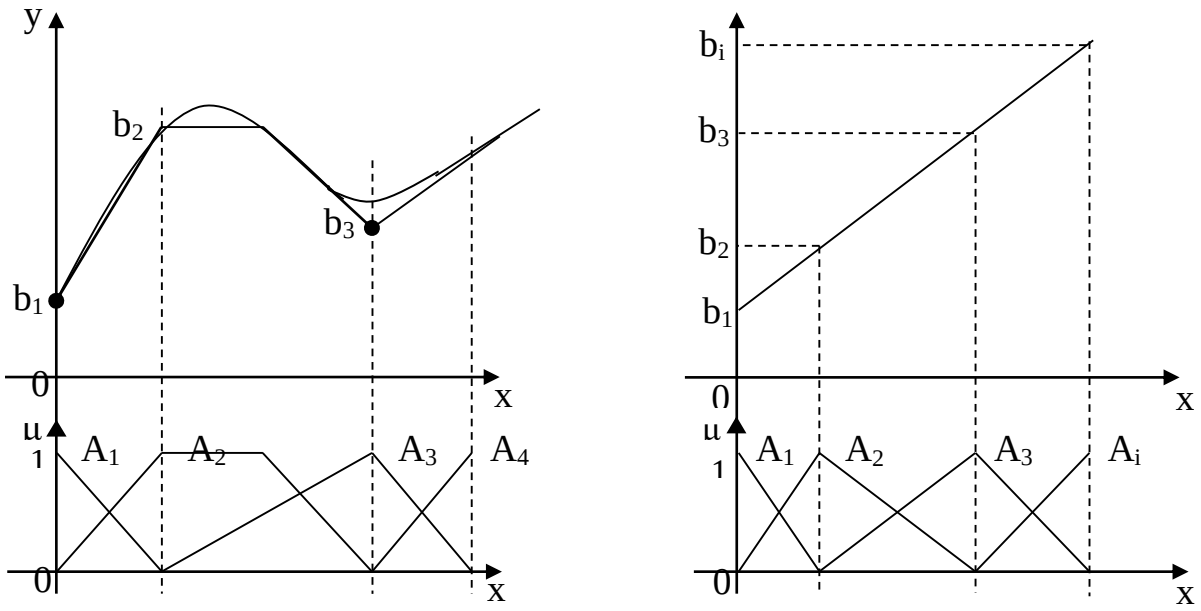


$$z = \frac{w_1 z_1 + w_2 z_2}{w_1 + w_2}$$

Hinh 3. 9 hàm liên thuộc của luật Điều khiển theo TS

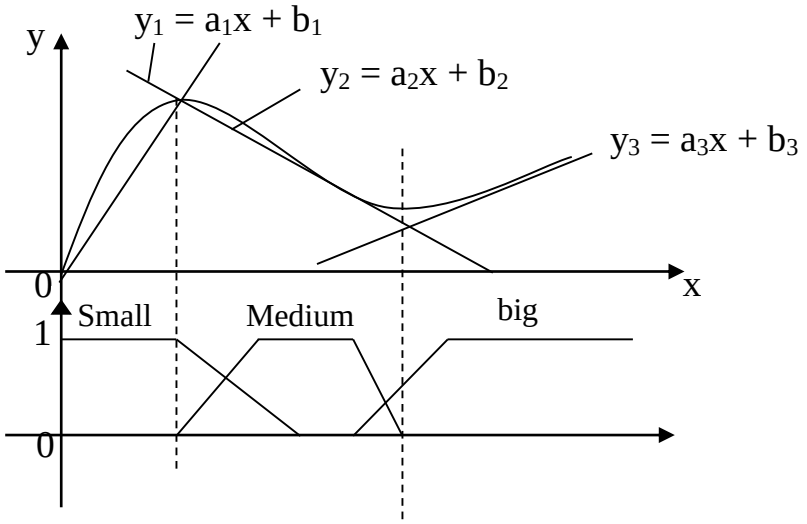
$$y = \frac{\sum_{i=1}^{k} \beta_i b_i}{\sum_{j=1}^{k} \beta_j} \qquad (3.12)$$

Ánh xạ vào/ ra của hệ mờ khi hàm đầu ra là $y = f(x)$ và $y = kx + q$



Hinh 3. 10 Mô hình đơn giản với các hàm thuộc hình thang và tam giác cho ánh xạ vào/ ra

Giả sử hàm đầu ra có dạng:



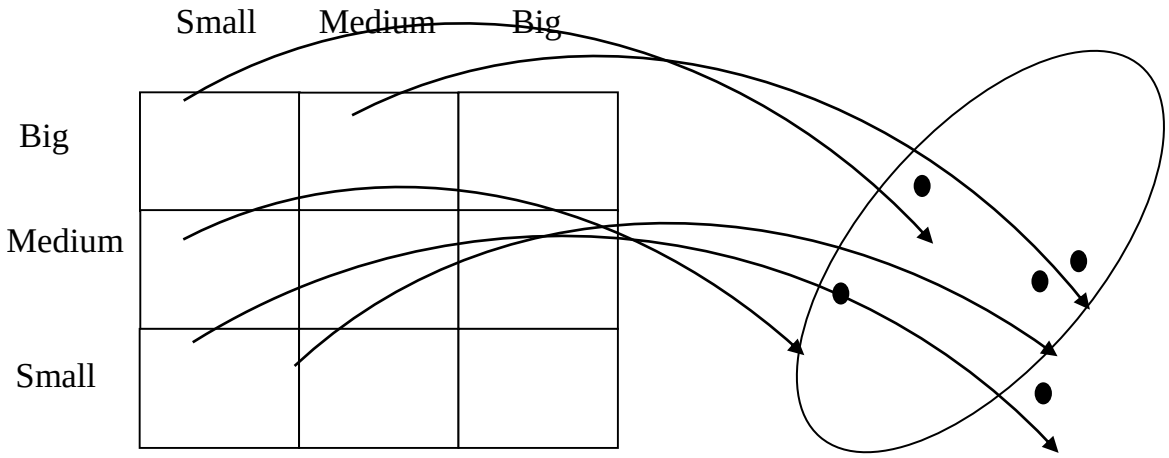
Hinh 3. 11Mô hình TS xấp xỉ từng đoạn cho hàm phi tuyến f(x)

Nếu đặt $r_i(x) = \frac{\beta_i(x)}{\sum_{j=1}^{k}\beta_j(x)}$, Từ (3.11) ta có:

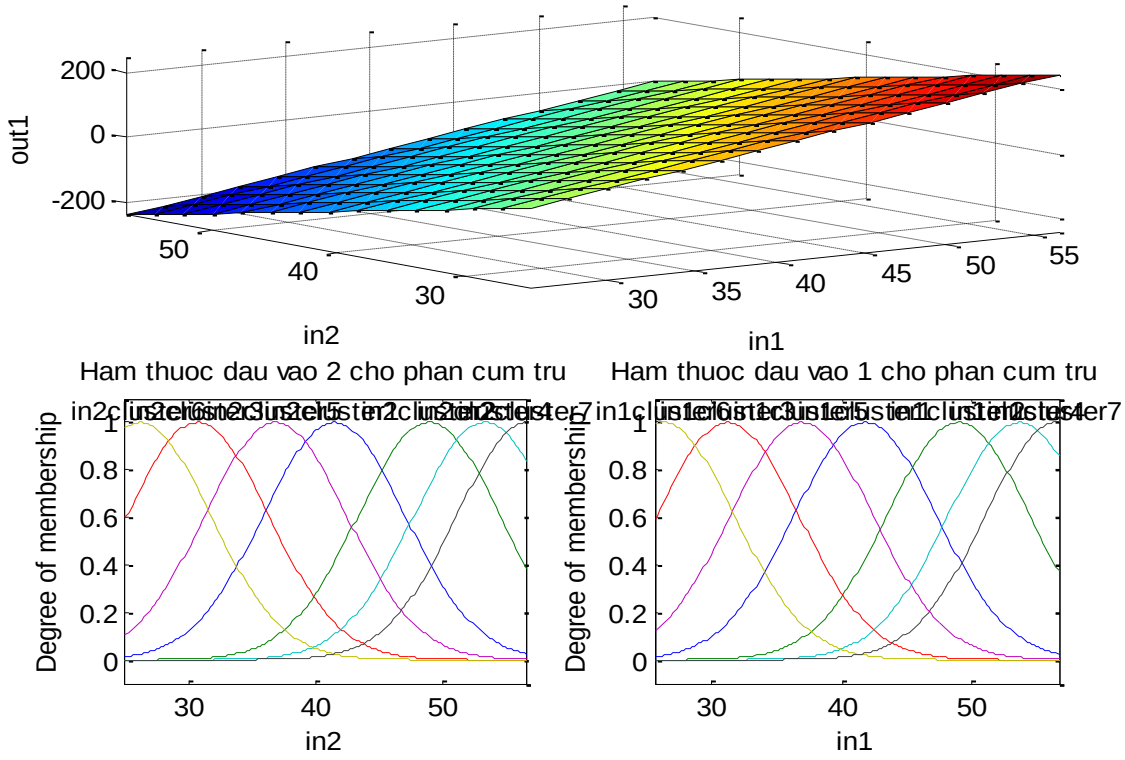
$$y = ((\sum_{i=1}^{k} r_i(x)a_i^T)X + \sum_{i=1}^{k} r_i(x)b_i = a^T(x)X + b(x) \quad (3.13)$$

Các thông số a_i, b_i có thể xác định được qua :

$$a(x) = \sum_{i=1}^{k} r_i(x)a_i \; ; \; b(x) = \sum_{i=1}^{k} r_i(x)b_i$$



Hinh 3. 12 Biểu diễn ánh xạ từ không gian vào đến không gian ra



Hinh 3. 13 Mặt suy diễn và các hàm thuộc đầu vào của hệ điều khiển

3.3.4 Mô phỏng hệ thống điều khiển lò nhiệt sử dụng hệ luật mờ từ phân cụm trừ

3.3.4.1 Các chức năng chương trình

Luận văn đã sử dụng phần mềm lập trình Matlab xây dựng chương trình ứng dụng phân cụm dữ liệu trừ trong việc đo và điều khiển nhiệt độ.

Công cụ Matlab được thực hiện qua các bước sau:

- Đọc dữ liệu
- Xây dựng cấu trúc dữ liệu
- Xử lý tập dữ liệu trước khi huấn luyện
- Khởi tạo mẫu và huấn luyện
- Mô phỏng kết quả.
- Phân tích kết quả để đưa ra các nhận xét

3.3.4.2 Chương trình mô phỏng hệ điều khiển lò nhiệt

Để điều khiển lò nhiệt trên cơ sở tự động xây dựng hệ luật mờ theo (3.4) chúng ta xuất phát từ dữ liệu thu thập được qua hình 3.10. Từ hình 3.10 ta sử dụng kỹ thuật phân cụm trừ để tạo ra luật điều khiển. Chương trình điều khiển lò nhiệt viết trên phần mềm matLab như sau:

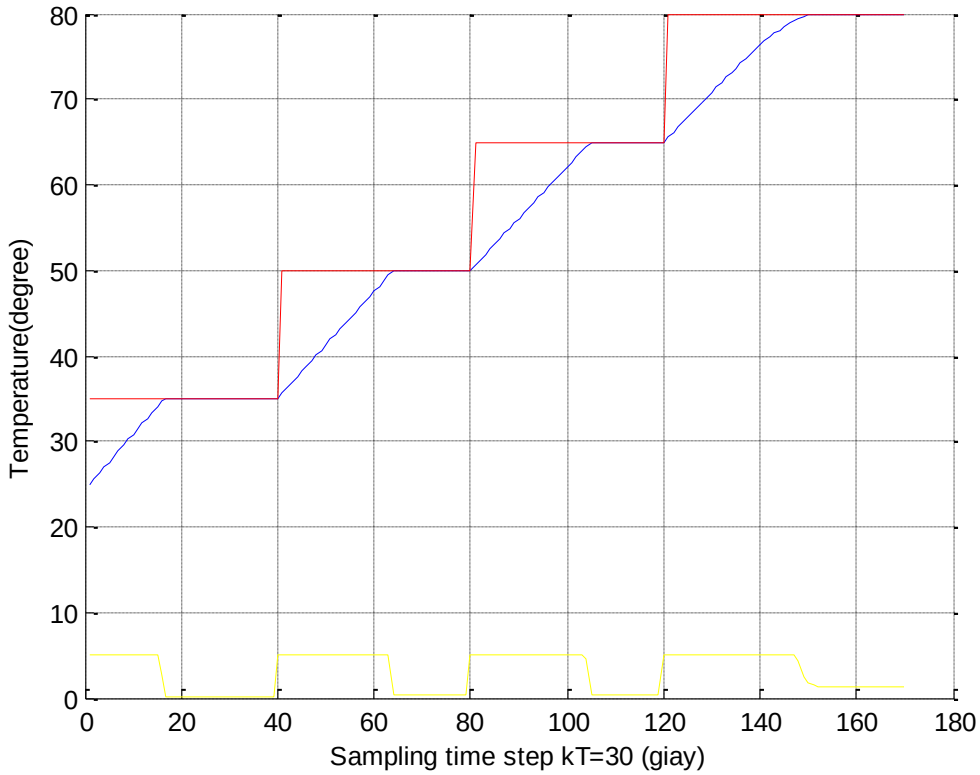
```
%                    TRUONG DAI HOC THAI NGUYEN
%////////////////////////////////////////////////////////////////////////////////////////////////////////////////
%          CHUONG TRINH DIEU KHIEN LO NHIET
%
%
%                    DE TAI DIEU KHIEN LO NHIET
%                    SU DUNG PHAN CUM DU LIEU TRU
%
%
%
%
%
%
%
%                              Nguoi thuc hien
%                               Do Thi Kim Dung
%                                   2017
%
%
%////////////////////////////////////////////////////////////////////////////////////////////////////////////////

Ts=15;p=1.00151*10^(-4);q=8.6797*10^(-3);r=40;y0=25;y(1)=y0;
a=exp(-p*Ts);b=(q/p)*(1-exp(-p*Ts));
%//////////////////////////////////
%Chuong trinh tu day la DATASET /
%//////////////////////////////////
for k=1:120
u(k)=rand(1,1)*5;
y(k+1)=a*y(k)+b/(1+exp(0.5*y(k)-r))*u(k)+(1-a)*y0;
```

```
end;
trndata=[y(2:101); y(1:100)]';
datout=u(1:100)'
figure
hold on
subplot(2,1,1), plot(trndata);
subplot(2,1,2), plot(datout);
%//////////////////////////////////////////////
%Chuong trinh tu day la SUBSTRACTIVE CLUSTERING /
%//////////////////////////////////////////////
chkdatin=trndata;
fismat=genfis2(trndata,datout,0.5); % ham phan cum tru
fuzout=evalfis(trndata,fismat);
ruleview(fismat)
ruleedit(fismat)
%showrule(fismat)
getfis(fismat,'output',1,'mf',1)
getfis(fismat,'output',1,'mf',2)
figure
subplot(2,1,1);
gensurf(fismat);
%blackbg;
subplot(2,2,4);
%hold on
plotmf(fismat,'input',1);
Title('Ham thuoc dau vao 1 cho phan cum tru')
%subplot(223);
subplot(2,2,3)
plotmf(fismat,'input',2);
```

```
Title('Ham thuoc dau vao 2 cho phan cum tru')
%plotmf(fismat,'input',1);
%subplot(2,2,4)
%plotmf(fismat,'output',1);
trnRMSE=norm(fuzout-datout)/sqrt(length(fuzout));
chkfuzout=evalfis(chkdatin,fismat);
%chkRMSE=norm(chkfuzout-chkdatout)/sqrt(length(chkfuzout))
Ts=25;y0=25;y(1)=y0;
for k=1:180
if k<=40 ref(k)=35
elseif (k>40 &k<=80) ref(k)=50
elseif (k>80 &k<=120) ref(k)=65
elseif(k >120) ref(k)=80;end;
end;
for k=1:179
u(k)=evalfis([ref(k+1) y(k)],fismat);
if (u(k)>=5) u(k)=5
else u(k)=u(k); end;
y(k+1)=a*y(k)+b/(1+exp(0.5*y(k)-r))*u(k)+(1-a)*y0;
end;
figure
hold on; grid
plot (y(1:170),'b');plot(ref(1:170),'-r');plot(u(1:170),'g');
figure
subplot(2,2,1);
plotmf(fismat2,'input',1);
Title('Ham thuoc dau vao 1')
%subplot(223);
subplot(2,2,2);
```

```
plotmf(fismat2,'input',2);
Title('Ham thuoc dau vao 2')
hold off
%Define training data
%blackbg;
subplot(2,2,3);
hold on; grid;
plot(y(1:180),'b');
plot(ref(1:180),'--r');
Title('Response-blue,desired-red');
hold off;
%blackbg;
subplot(2,2,4);
hold on; grid;
plot(u(1:180),'b');
plot((ref(1:180)-y(1:180)),'--r');
Title('Control-blue - and  Error-red - ');
hold off;
%//////////////////////////////
```

Quá trình điều khiển nhiệt độ của lò nhiệt được mô phỏng bằng phần mềm MatLab trên hình 3.19



Hình 3. 14 Đáp ứng ra (xanh) bám theo tín hiệu yêu cầu (đỏ)

Từ hình 3.19 ta thấy:

Với nhiệt độ đặt là 35^0C, nếu nhiệt độ trong lò dưới 35^0C thì tín hiệu điều khiển (đường màu xanh ở dưới) được đưa vào để điều khiển nhiệt độ trong lò tăng lên đến 35^0C (tiệm cận với nhiệt độ đặt là đường màu đỏ). Khi đó tín hiệu điều khiển được ngắt ra để nhiệt độ trong lò ở mức nhiệt độ đặt.

Tương tự khi nhiệt độ trong lò cần tăng lên mức 500C thì tín hiệu điều khiển (đường màu xanh ở dưới) được đưa vào để điều khiển nhiệt độ trong lò tăng lên đến 500C (tiệm cận với nhiệt độ đặt là đường màu đỏ). Khi đó tín hiệu điều khiển được ngắt ra để nhiệt độ trong lò ở mức nhiệt độ đặt.

Khi nhiệt độ trong lò cần tăng đến 800C thì tín hiệu điều khiển (đường màu xanh ở dưới) được đưa vào để điều khiển nhiệt độ trong lò tăng lên đến 800C (tiệm cận với nhiệt độ đặt là đường màu đỏ). Khi đó tín hiệu điều khiển được ngắt ra để nhiệt độ trong lò ở mức nhiệt độ đặt là 800C.

KẾT LUẬN

Sự phát triển nhanh chóng của các hệ thống điều khiển, các hệ thống thông tin như hiện nay, thì hệ mờ được áp dụng thành công trong nhiều lĩnh vực như điều khiển tự động, phân lớp dữ liệu, phân tích việc ra quyết định, các hệ chuyên gia. Hệ luật mờ xây dựng từ tri thức nói chung hay hệ suy luận mờ nói riêng được xây dựng theo suy diễn của con người, là một phần quan trọng trong ứng dụng logic mờ cũng như trong lý thuyết tập mờ vào thực tế. Trong nhiều ứng dụng cho thiết kế các hệ thống điều khiển thông minh cũng như trong xây dựng các hệ trợ giúp quyết định, hệ mờ được xây dựng theo phân lớp dữ liệu, phân cụm dữ liệu, xây dựng cây quyết định. Hệ điều khiển mờ được thực hiện từ các luật mờ, các luật mờ được xây dựng từ các tri thức của các chuyên gia trong một lĩnh vực cụ thể hoặc được xây dựng từ dữ liệu.

Phân cụm dữ liệu đang là một vấn đề quan tâm nghiên cứu của các tác giả trong và ngoài nước [2,3,4,5] và có nhiều thuật toán phân cụm được đề xuất. Tuy nhiên các thuật toán được đưa ra mới chỉ xét đến khía cạnh phân chia dữ liệu thành các cụm với độ chính xác cao mà chưa để tâm đến sự tối ưu các luật sử dụng, ví dụ giải thuật GA K-means được sử dụng trong bài toán thị trường mua sắm trực tuyến do Kyoung-jae Kim và Hyunchul Ahn đưa ra là thuật toán sử dụng K-means kết hợp với giải thuật di truyền và được chứng tỏ có sự cải thiện đáng kể trong việc thực hiện phân nhóm so với các thuật toán phân cụm điển hình khác. Hoặc phương pháp phân cụm bán giám sát (Semi- Supervisor Clustering) dùng giải thuật di truyền do Ayhan Demiriz; Kristin P. Bennett và Mark J. Embrechts thuộc Rensselaer Polytechnic Institute - Troy, NY 12180 đề xuất năm 1999 là sự kết hợp các ưu điểm của các phương pháp học có giám sát và học không giám sát. Bằng các kết quả thực nghiệm, phương pháp này chỉ ra lợi thế trong trường hợp có ít mẫu huấn luyện.

TÀI LIỆU THAM KHẢO

Tiếng Việt

[1] Lê Bá Dũng, Các hệ cơ sở tri thức (knowledge based system) và ứng dụng, Bài giảng ĐHBK Hà nội – Genetic computer school joint education program.

[2] Bùi Công Cường, Nguyễn Doãn Phước, "Lý thuyết mờ và công nghệ tính toán mềm", *Hệ mờ mạng nơron và ứng dụng*, Nhà xuất bản Khoa học và Kỹ thuật, pp.53-89, 2006.

[3] Nguyễn Trung Sơn, *Phương pháp phân cụm và ứng dụng*, Khoa công nghệ thông tin - Đại học Thái Nguyên, luận văn thạc sĩ, 2009.

[4] Nguyễn Đình Thúc (2000), *Trí tuệ nhân tạo Mạng nơron phương pháp & ứng dụng* , Nhà xuất bản Giáo dục.

[5] Đỗ Phúc, giáo trình khai thác dữ liệu, NXB Đại học quốc gia TP HCM Data.Mining.Concepts.and.Techniques.2nd.Ed-1558609016.

[6] Trần Mạnh Tuấn, Lê Bá Dũng, *Ứng dụng phân cụm trừ mờ cho bài toán nhận dạng hệ điều khiển tự động từ dữ liệu*, Tạp chí Khoa Học Công Nghệ 116(02), 73-77, 2014.

Tiếng Anh

[6] Blake, C. L., & Merz, C. J. (1998). UCI repository of machine learning databases,1998.

[7] S. Nascimento, B. Mirkin and F. Moura-Pires, A Fuzzy Clustering Model of Data and Fuzzyc- Means.

[8] W.Pedrycz, "Algorithms of fuzzy clustering with partial supervision", Pattern Recognition, vol. 23, pp.121-146, 1990.

[9] Jiawei Han, Micheline Kamber, *Datamining Concepts and Techniques*, Morgan Kaufmann Publishers, 2nd edition, 2006.

[10] A.K. Jain, R.C. Dubes, *Algorithms for clustering data*, Ptentice Hall, Englewood Cliffs, NJ, 1988.

[11] M.P.Windham, "Cluster validity for fuzzy clustering algorithms", *Fuzzy Sets and System*, vol. 3, pp.177-183, 1981.

[12] W.Pedrycz, “Algorithms of fuzzy clustering with partial supervision”, *Pattern Recognition*, vol. 23, pp.121-146, 1990.

[13] Gita Sastria, Choong Yeun Liong, Ishak Hashim, “Application of Fuzzy Subtractive Clustering for Enzymes Classification”, *Applied Computing Conference*, Istanbul, Turkey, 2008