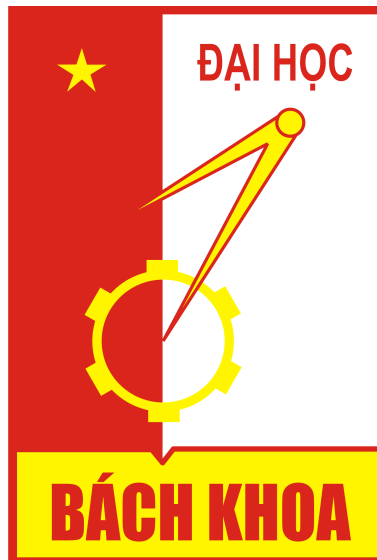


**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC**



**Báo cáo môn học**  
**PHÂN TÍCH SỐ LIỆU**

**Đề tài: Clustering, distance methods, and ordination**

**Giảng viên: Thạc sỹ Lê Xuân Lý**

<b>Sinh viên:</b>	Nguyễn Hoàng Quốc Anh	20185320
	Hoàng Tuấn Tài	20185401
	Vũ Việt Hoàng	20183926
	Đỗ Mạnh Dũng	20195860
	Nguyễn Anh Minh	20194117

**Ngày 11 tháng 1 năm 2022**

# Mục lục

<b>Mở đầu</b>	<b>4</b>
<b>1 Giới thiệu</b>	<b>6</b>
<b>2 Các biện pháp tương tự</b>	<b>9</b>
2.1 Khoảng cách và hệ số tương tự cho các cặp quan sát . . . . .	9
2.2 Sự tương tự và các thước đo liên kết cho các cặp biến . . . . .	14
2.3 Kết luận về sự tương tự . . . . .	16
<b>3 Thuật toán phân cụm phân cấp</b>	<b>18</b>
3.1 Giới thiệu . . . . .	18
3.1.1 Phương pháp phân cụm theo cấp . . . . .	18
3.1.2 Biểu diễn kết quả . . . . .	18
3.2 Các phương pháp phân cụm tổng gộp theo cấp . . . . .	20
3.2.1 Các loại kết nối . . . . .	20
3.2.2 Phân cụm theo mức của Ward . . . . .	23
3.2.3 BIRCH . . . . .	25
3.2.4 ROCK . . . . .	28
3.3 Phân cụm phân tách theo cấp . . . . .	30
3.3.1 DIANA . . . . .	30
3.4 Kết quả thực nghiệm . . . . .	32
3.5 Kết luận chung . . . . .	35
<b>4 Thuật toán phân cụm không phân cấp</b>	<b>37</b>
4.1 Phương pháp luận . . . . .	37
4.1.1 Thuật toán K-Means . . . . .	37
4.1.2 Phân tích độ hiệu quả của thuật toán cải tiến K-Means++ . . . . .	43

4.2	Kết quả thực nghiệm . . . . .	48
4.2.1	Loại bỏ dữ liệu ngoại lai . . . . .	48
4.2.2	Phân tích Silhoutte và Phương pháp khuỷu tay . . . . .	48
4.2.3	Kết quả cuối cùng . . . . .	50
<b>5</b>	<b>Thuật toán phân cụm theo mô hình xác suất thống kê</b>	<b>52</b>
5.1	Phương pháp luận . . . . .	52
5.1.1	Vấn đề đặt ra . . . . .	52
5.1.2	Mô hình Gaussian hỗn hợp . . . . .	52
5.1.3	Thuật toán cực đại hóa kì vọng . . . . .	54
5.1.4	Xác định số phân cụm dựa vào tiêu chuẩn AIC và BIC . . . . .	58
5.2	Kết quả thực nghiệm . . . . .	60
<b>6</b>	<b>Thuật toán phân cụm theo phổ (Spectral Clustering)</b>	<b>64</b>
6.1	Giới thiệu . . . . .	64
6.2	Đồ thị tương tự . . . . .	64
6.2.1	Ký hiệu đồ thị . . . . .	64
6.2.2	Các loại đồ thị tương tự . . . . .	65
6.3	Đồ thị Laplace và đặc điểm . . . . .	66
6.3.1	Ma trận Laplace không chuẩn hóa . . . . .	66
6.3.2	Ma trận Laplace chuẩn hóa . . . . .	67
6.4	Thuật toán phân cụm phổ . . . . .	68
6.4.1	Phân cụm phổ không chuẩn hóa . . . . .	68
6.4.2	Phân cụm phổ chuẩn hóa theo Shi and Malik . . . . .	69
6.5	Lát cắt đồ thị . . . . .	69
6.5.1	Xấp xỉ RatioCut . . . . .	70
6.5.2	Xấp xỉ Ncut . . . . .	71
6.6	Thảo luận . . . . .	72
6.6.1	Xây dựng đồ thị tương tự . . . . .	72

6.6.2	Số cụm . . . . .	74
6.6.3	Lựa chọn thuật toán . . . . .	74
6.7	Thực nghiệm . . . . .	74
<b>7</b>	<b>Chia tỷ lệ đa chiều (Multidimensional scaling)</b>	<b>77</b>
7.1	Giới thiệu . . . . .	77
7.2	Thuật toán cơ bản . . . . .	77
7.3	Thực nghiệm . . . . .	79

## Mở đầu

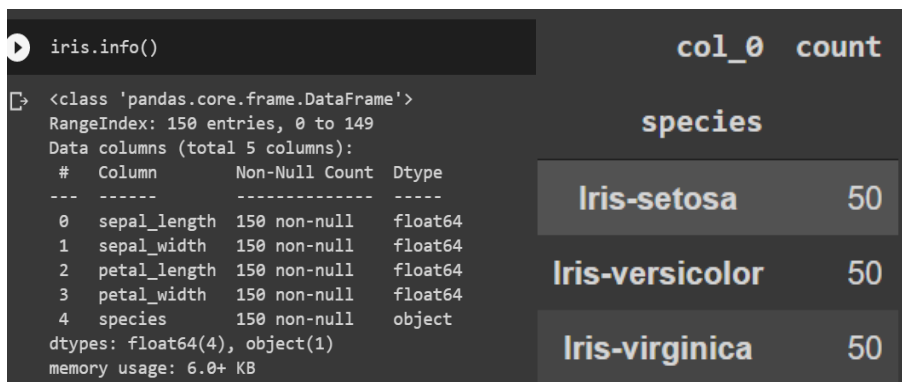
Trước khi trình bày nội dung chính của bản báo cáo, chúng em xin được gửi lời cảm ơn sâu sắc tới thầy Lê Xuân Lý - giảng viên giảng dạy môn "Phân tích số liệu". Những kiến thức nền tảng thầy truyền đạt cho chúng em trong môn học đóng vai trò vô cùng quan trọng, là cơ sở giúp chúng em hoàn thành nội dung bản báo cáo này. Chúng em cũng xin gửi lời cảm ơn tới tất cả các bạn đã thuyết trình rất tâm huyết và đã giúp chúng em tích lũy được những kiến thức cơ bản quan trọng trong suốt quá trình học tập.

Bài báo cáo này đã được nghiên cứu và viết bởi tất cả các thành viên trong nhóm. Trong quá trình làm việc, nhóm đã thống nhất các buổi họp hàng tuần vào tối chủ nhật để hoàn thành bài thuyết trình cũng như bài báo cáo một cách chẵn chu nhất. Các thành viên trong nhóm ai cũng đã cố gắng và đã đều nắm được hết các kiến thức căn bản của các thuật toán phân cụm. Không những vậy, các thành viên trong nhóm còn tích cực tìm hiểu các thuật toán phân cụm nâng cao khác ngoài khuôn khổ của giáo trình. Vì vậy, đánh giá mức độ tích cực của các thành viên đều là **5/5**.

Do thời gian tìm hiểu và nghiên cứu có giới hạn nên bản báo cáo không tránh khỏi những sai sót, chúng em rất mong nhận được những ý kiến đóng góp, sửa chữa của thầy Lê Xuân Lý, để bản báo cáo được hoàn thiện và chính xác hơn. Chúng em xin chân thành cảm ơn!

### Thông tin bộ dữ liệu

Trong bài báo cáo này, để thống nhất và đồng bộ, nhóm sẽ sử dụng bộ dữ liệu Iris trong tất cả các phần chạy thực nghiệm của mỗi phương pháp. Bộ dữ liệu Iris bao gồm 150 điểm dữ liệu là các bông hoa Iris với 4 đặc tính quan sát là độ dài, độ rộng cánh hoa và độ dài, độ rộng nhụy hoa. 150 bông hoa Iris thuộc 3 chủng Iris khác nhau là Iris-setosa, Iris-versicolor, Iris-virginica. Mỗi chủng gồm 50 điểm dữ liệu.



The image shows a Jupyter Notebook interface. On the left, a code cell contains the command `iris.info()`. Below it, the output is displayed, showing the DataFrame's structure: 150 entries, 5 columns (sepal\_length, sepal\_width, petal\_length, petal\_width, species). The output also shows the data types (float64 for measurements, object for species) and memory usage (6.0+ KB). On the right, a summary table is shown with columns 'col\_0' and 'count'. The 'col\_0' column lists the species: 'Iris-setosa', 'Iris-versicolor', and 'Iris-virginica'. The 'count' column shows the number of samples for each species: 50 for each.

col_0	count
Iris-setosa	50
Iris-versicolor	50
Iris-virginica	50

Hình 1: Mô tả dữ liệu

Mỗi điểm dữ liệu sẽ có 4 dữ liệu tượng trưng cho 4 đặc điểm quan sát, mô tả 5 điểm dữ liệu đầu như sau

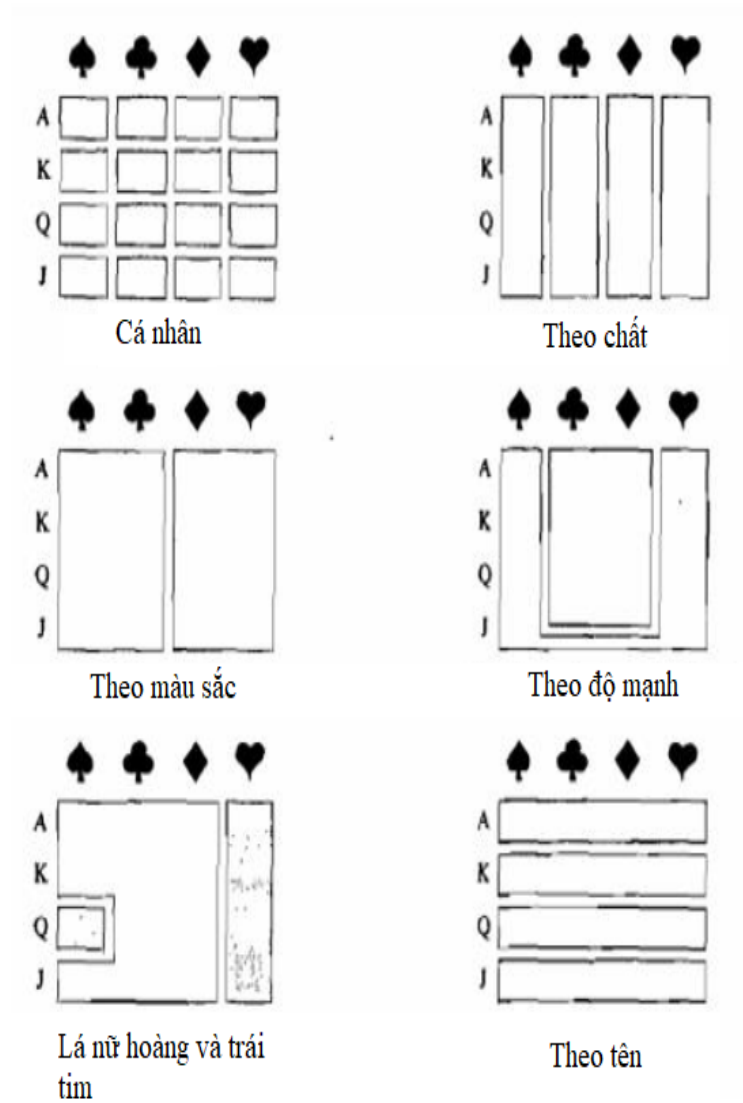
	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

Hình 2: Mô tả dữ liệu

# 1 Giới thiệu

Giới thiệu Các thủ tục thăm dò, thô sơ thường khá hữu ích trong việc hiểu bản chất phức tạp của các mối quan hệ đa biến. Ví dụ, trong suốt môn học này, chúng ta đã nhấn mạnh giá trị của các ô dữ liệu. Trong báo cáo này, chúng ta sẽ đề cập đến một số hiển thị bổ sung dựa trên các thước đo khoảng cách nhất định và các quy tắc thuật toán để nhóm các đối tượng (các biến hoặc các quan sát). Tìm kiếm dữ liệu cho cấu trúc của các nhóm "tự nhiên" là một kỹ thuật khám phá quan trọng. Việc phân nhóm có thể cung cấp một phương tiện không chính thức để đánh giá các chiều, xác định các giá trị ngoại lai và đề xuất các giả thuyết thú vị bao hàm các mối quan hệ. Phân nhóm, hoặc phân cụm, khác với các phương pháp phân loại đã thảo luận trong chương trước. Phân loại liên quan đến một số nhóm đã biết và mục tiêu hoạt động là chỉ định các quan sát mới cho một trong các nhóm này. Phân tích cụm là một kỹ thuật nguyên thủy hơn trong đó không có giả định nào được đưa ra liên quan đến số lượng nhóm hoặc cấu trúc nhóm. Việc phân nhóm được thực hiện trên cơ sở các điểm tương đồng hoặc khoảng cách (hay còn hiểu là sự khác nhau). Các đầu vào yêu cầu là các thước đo hoặc dữ liệu về độ tương đồng mà từ đó các điểm tương đồng có thể được tính toán.

Để minh họa bản chất của khó khăn trong việc xác định một nhóm tự nhiên, hãy xem xét việc sắp xếp 16 lá bài có mặt trong một bộ bài bình thường thành các cụm có các đối tượng giống nhau. Một số nhóm được minh họa trong hình:



Rõ ràng ngay lập tức rằng các phân nhóm có ý nghĩa phụ thuộc vào định nghĩa của tương tự. Trong hầu hết các ứng dụng thực tế của phân tích cụm, nhà phân tích biết đủ về vấn đề đang nghiên cứu để phân biệt nhóm "tốt" với nhóm "xấu". Vậy thì có một câu hỏi được đặt ra ở đây đó là tại sao không liệt kê tất cả các nhóm có thể có và chọn những nhóm "tốt nhất" để nghiên cứu thêm?

Một công thức được sử dụng để tính số lượng cách phân  $n$  phần tử thành  $k$  nhóm là công thức số Stirling loại 2:

$$\left\{ \begin{matrix} n \\ k \end{matrix} \right\} = \frac{1}{k!} \sum_{i=0}^k (-1)^i \binom{k}{i} (k-i)^n$$



Chúng ta có thể thấy một số kết quả phân nhóm như ở bảng dưới đây:

$n \backslash k$	0	1	2	3	4	5	6	7	số 8	9	10
0	1										
1	0	1									
2	0	1	1								
3	0	1	3	1							
4	0	1	7	6	1						
5	0	1	15	25	10	1					
6	0	1	31	90	65	15	1				
7	0	1	63	301	350	140	21	1			
số 8	0	1	127	966	1701	1050	266	28	1		
9	0	1	255	3025	7770	6951	2646	462	36	1	
10	0	1	511	9330	34105	42525	22827	5880	750	45	1

Quay trở lại với ví dụ chơi bài, có một cách để tạo thành một nhóm gồm 16 thẻ mặt, có 32.767 cách để phân chia các thẻ mặt thành hai nhóm (có kích thước khác nhau), có 7.141.686 cách để sắp xếp các thẻ mặt thành ba nhóm (có kích thước khác nhau), v.v. ' Rõ ràng, những hạn chế về thời gian khiến chúng ta không thể xác định nhóm tốt nhất của các đối tượng tương tự từ danh sách tất cả các cấu trúc có thể có. Ngay cả các máy tính nhanh cũng dễ dàng bị áp đảo bởi số lượng trường hợp lớn, vì vậy người ta phải giải quyết các thuật toán tìm kiếm các nhóm tốt, nhưng không nhất thiết là tốt nhất. Trong chương sau được dành để thảo luận về các biện pháp tương tự. Sau phần đó, chúng ta sẽ mô tả một số thuật toán phổ biến hơn để sắp xếp các đối tượng thành các nhóm.

## 2 Các biện pháp tương tự

Các biện pháp tương tự hầu hết các nỗ lực để tạo ra một cấu trúc nhóm khá đơn giản từ một tập dữ liệu phức tạp đều đặt ra một thước đo về "tính không giống nhau" hoặc "tính tương tự". Thường có rất nhiều yếu tố liên quan đến việc lựa chọn một thước đo tương tự. Những cân nhắc quan trọng bao gồm bản chất của các biến (rời rạc, liên tục, nhị phân), thang đo lường (đanh nghĩa, thứ tự, khoảng, tỷ lệ) và kiến thức chủ đề. Khi các quan sát (đơn vị hoặc trường hợp) được nhóm lại, khoảng cách thường được biểu thị bằng một số loại khoảng cách. Ngược lại, các biến thường được nhóm lại trên cơ sở các hệ số tương quan hoặc như các thước đo liên kết.

### 2.1 Khoảng cách và hệ số tương tự cho các cặp quan sát

Chúng ta đã thảo luận về khái niệm khoảng cách trong phần đầu của môn học này, Nhó lại rằng khoảng cách Euclide (đường thẳng) giữa hai quan sát  $p$ -chiều:

$$\begin{aligned}x' &= [x_1, x_2, \dots, x_p] \text{ và } y' = [y_1, y_2, \dots, y_p] \\d(x, y) &= \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_p - y_p)^2} \\&= \sqrt{(x - y)'(x - y)}\end{aligned}$$

Khoảng cách thống kê giữa hai quan sát giống nhau có dạng:

$$d(x, y) = \sqrt{(x - y)'A(x - y)}$$

Thông thường  $A = S^{-1}$ , trong đó  $S$  chứa phương sai và hiệp phương sai mẫu. Tuy nhiên, nếu không có thông tin thì không thể tính được các đại lượng này. Vì lí do này, khoảng cách Euclide thường được ưu tiên phân cụm. Một thước đo khoảng cách khác là số liệu Minkowski:

$$d(x, y) = \left[ \sum_{i=1}^p |x_i - y_i|^m \right]^{1/m}$$

- Với  $m=1$ ,  $d(x, y)$  đo khoảng cách giữa hai điểm theo  $p$  chiều
- Với  $m=2$ ,  $d(x, y)$  trở thành khoảng cách Euclidean

Hai thước đo phổ biến bổ sung về "khoảng cách" hoặc sự khác biệt được đưa ra bởi hệ số Canberra và hệ số Czekanowski. Cả hai thước đo này chỉ được xác định cho các biến không âm.

- *Hệ số Canberra:*

$$d(x, y) = \sum_{i=1}^p \frac{|x_i - y_i|}{(x_i + y_i)}$$

- *Hệ số Czekanowski:*

$$d(x, y) = 1 - \frac{2 \sum_{i=1}^p \min(x_i, y_i)}{\sum_{i=1}^p (x_i + y_i)}$$

Nên sử dụng khoảng cách "true" - nghĩa là khoảng cách thỏa mãn các thuộc tính (đối với các đối tượng phân cụm):

$$d(P, Q) = d(Q, P)$$

$$d(P, Q) > 0 \text{ nếu } P \neq Q$$

$$d(P, Q) = 0 \text{ nếu } P = Q$$

$$d(P, Q) \leq d(P, R) + d(R, Q) \text{ (bất đẳng thức tam giác)}$$

Mặt khác, hầu hết các thuật toán phân cụm sẽ chấp nhận các số khoảng cách được ấn định một cách chủ quan có thể không thỏa mãn, như là bất đẳng thức tam giác.

Khi các vật phẩm không thể được biểu thị bằng các đặc tính p-chiều có ý nghĩa, thì các cặp vật phẩm thường được so sánh trên cơ sở có hoặc không có các đặc điểm nhất định. Các mặt hàng tương tự có nhiều đặc điểm chung hơn là các mặt hàng khác nhau. Sự hiện diện hoặc không có của một đặc tính có thể được mô tả bằng toán học bằng cách đưa vào một biến nhị phân, giả định giá trị 1 nếu có đặc tính và giá trị 0 nếu không có đặc tính.

Ví dụ: đối với biến nhị phân  $p = 5$ , giá trị cho hai quan sát  $i$  và  $k$  có thể được sắp xếp như sau:

	Giá trị				
	1	2	3	4	5
Quan sát $i$	1	0	0	1	1
Quan sát $k$	1	1	0	1	0

Bảng 1

Trong trường hợp này, có hai cặp 1-1, một cặp 0-0 và hai cặp không khớp.

Gọi  $x_{ij}$  là giá trị (1 hoặc 0) của biến nhị phân thứ  $j$  trên quan sát thứ  $i$  và  $x_{kj}$  là giá trị (1 hoặc 0) của biến nhị phân thứ  $j$  trên quan sát thứ  $k$ ,  $j = 1, 2, \dots, p$

Kết quả là:

$$(x_{ij} - x_{kj})^2 = \begin{cases} 0 & \text{nếu } x_{ij} = x_{kj} = 1 \text{ hoặc } x_{ij} = x_{kj} = 0 \\ 1 & \text{nếu } x_{ij} \neq x_{kj} \end{cases}$$

Bình phương khoảng cách Euclidean, cung cấp số lượng các cặp không khớp. Một khoảng cách lớn tương ứng với nhiều cặp không khớp - nghĩa là, khác biệt các quan sát. Từ công thức trên, bình phương khoảng cách giữa các quan sát i và k sẽ là

$$\begin{aligned} \sum_{j=1}^5 (x_{ij} - x_{kj})^2 &= (1-1)^2 + (0-1)^2 + (0-0)^2 + (1-1)^2 + (1-0)^2 \\ &= 2 \end{aligned}$$

Mặc dù khoảng cách dựa trên công thức Euclidean có thể được sử dụng để đo độ tương tự, nhưng nó không có nghĩa là cân bằng tỷ số các cặp 1-1 và 0-0. Trong một số trường hợp, cặp 1-1 có điểm tương đồng mạnh hơn so với cặp 0-0

Ta xét một ví dụ như sau khi bàn về việc học ba môn toán lí hóa của hai người A và B, và ta biết A học giỏi toán và B học giỏi lí, nếu cả hai người cùng giỏi hóa thì đây là một cặp tương đồng mạnh khi xét môn hóa, nhưng nếu cả hai không giỏi hóa thì nó không thực sự là một cặp tương đồng mạnh vì một người giỏi toán còn một người giỏi lí không hề giống nhau.

Do đó, có thể là hợp lý khi giảm giá trị các cặp 0-0 hoặc thậm chí bỏ qua chúng hoàn toàn. Để cho phép xử lý khác biệt giữa các cặp 1-1 và các cặp 0-0, một số phương án để xác định các độ tương tự đã được đề xuất. Để giới thiệu những phương án này, chúng ta hãy sắp xếp tần suất của các kết quả phù hợp và không phù hợp cho các quan sát i và k dưới dạng Bảng 2:

		Quan sát k		Tổng
		1	0	
Quan sát i	1	a	b	a + b
	0	c	d	c + d
Tổng		a + c	b + d	p = a + b + c + d

Bảng 2

Trong bảng này, a đại diện cho tần suất của các cặp 1-1, b là tần suất của các cặp 1-0,... Với năm cặp kết quả nhị phân ở trên, a = 2 và b = c = d = 1.

Bảng 3 sẽ liệt kê các hệ số tương tự phổ biến được xác định theo tần suất trong bảng trên. Cơ sở lý luận ngắn gọn sau mỗi định nghĩa.

Hệ số	Cơ sở lý luận
$\frac{a+d}{p}$	Tỉ số cân bằng của cặp 1-1 và 0-0
$\frac{2(a+d)}{2(a+d)+b+c}$	Nhân hai tỉ số cân bằng của cặp 1-1 và 0-0
$\frac{a+d}{a+d+2(b+c)}$	Nhân hai tỉ số của các cặp không phù hợp
$\frac{a}{p}$	Không có cặp 0-0
$\frac{a}{a+b+c}$	Bỏ qua cặp 0-0
$\frac{2a}{2a+b+c}$	Bỏ qua cặp 0-0 và nhân hai tỉ số của cặp 1-1
$\frac{a}{a+2(b+c)}$	Bỏ qua cặp 0-0 và nhân hai tỉ số cặp không phù hợp
$\frac{a}{b+c}$	Tỉ lệ các cặp không phù hợp với cặp 0-0 bị loại trừ

Bảng 3

Hệ số 1, 2 và 3 trong bảng có quan hệ đơn điệu với nhau. Giả sử hệ số 1 được tính cho hai bảng dự phòng là Bảng I và Bảng II. Khi đó nếu, và hệ số 3 sẽ là ít nhất là lớn đối với Bảng I cũng như đối với Bảng II. Hệ số 5,6 và 7 cũng tính lại thứ tự tương đối của chúng.

Tính đơn điệu rất quan trọng, bởi vì một số thủ tục phân cụm sẽ không bị ảnh hưởng nếu định nghĩa về độ tương tự bị thay đổi theo cách làm thay đổi số lượng tương đối của các điểm tương đồng. Liên kết đơn và các thủ tục phân cấp liên kết hoàn chỉnh được thảo luận trong chương tiếp theo sẽ không bị ảnh hưởng. Đối với các phương pháp này, bất kỳ sự lựa chọn nào của các hệ số 1,2 và 3 trong Bảng trên sẽ tạo ra các nhóm tương tự. và bất kỳ sự lựa chọn nào của các hệ số 5, 6 và 7 sẽ mang lại các nhóm giống hệt nhau.

Ta xét một ví dụ về 5 cá thể có các đặc điểm về chiều cao, cân nặng, màu mắt, màu tóc, tay thuận, giới tính.

	Chiều cao	Cân nặng	Màu mắt	Màu tóc	Tay thuận	Giới tính
Cá nhân 1	68 in	140lb	xanh lá	đen	phải	nữ
Cá nhân 2	73 in	185lb	nâu	nâu	phải	nam
Cá nhân 3	67 in	165lb	xanh dương	đen	phải	nam
Cá nhân 4	64 in	120lb	nâu	nâu	phải	nữ
Cá nhân 5	76in	210lb	nâu	nâu	trái	nam

Bảng 4

Ta đặt 6 biến nhị phân  $X_1, X_2, X_3, X_4, X_5, X_6$  là:

$$X_1 = \begin{cases} 1 & \text{chiều cao} \geq 72in \\ 0 & \text{chiều cao} < 72in \end{cases} \quad X_4 = \begin{cases} 1 & \text{tóc đen} \\ 0 & \text{tóc nâu} \end{cases}$$

$$X_2 = \begin{cases} 1 & \text{cân nặng} \geq 150lb \\ 0 & \text{cân nặng} < 150lb \end{cases} \quad X_5 = \begin{cases} 1 & \text{tay phải} \\ 0 & \text{tay trái} \end{cases}$$

$$X_3 = \begin{cases} 1 & \text{mắt nâu} \\ 0 & \text{mắt khác} \end{cases} \quad X_6 = \begin{cases} 1 & \text{nữ} \\ 0 & \text{nam} \end{cases}$$

Ta xét giá trị cho 2 cá nhân 1,2 trên biến nhị phân  $p=6$  có:

	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$
Cá nhân 1	0	0	0	1	1	1
Cá nhân 2	1	1	1	0	1	0

Ta viết lại theo bảng tần số các cặp tương đồng:

		Cá nhân 2		
		1	0	Total
Cá nhân 1	1	1	2	3
	0	3	0	3
	Total	4	2	6

Sử dụng hệ số tương tự 1 trong Bảng 3 bên trên ta có trọng số cân bằng được tính như sau:

$$\frac{a+d}{p} = \frac{1+0}{6} = \frac{1}{6}$$

Tương tự như vậy ta tính toán với cả 6 cá nhân sẽ có được bảng tần suất của hệ số tương tự 1 với 6 người:

	1	2	3	4	5
1	1				
2	$\frac{1}{6}$	1			
3	$\frac{4}{6}$	$\frac{3}{6}$	1		
4	$\frac{4}{6}$	$\frac{3}{6}$	$\frac{2}{6}$	1	
5	0	$\frac{5}{6}$	$\frac{2}{6}$	$\frac{2}{6}$	1

Dựa trên độ lớn của hệ số tương tự, chúng ta có thể kết luận rằng cá thể 2 và 5 giống nhau nhất và cá thể 1 và 5 ít giống nhau nhất. Các cặp khác nằm giữa các thái cực này. Nếu chúng ta chia các cá thể thành hai nhóm con tương đối đồng nhất trên cơ sở các số lượng giống nhau, chúng ta có thể tạo thành các nhóm con (1 3 4) và (2 5). Lưu ý rằng  $X_3 = 0$  ngụ ý không có mắt nâu, do đó, hai người, một người có mắt xanh dương và một người có mắt xanh lá cây, sẽ có kết quả là cặp 0-0. Do đó, có thể không phù hợp khi sử dụng hệ số tương tự 1,2 hoặc 3 vì các hệ số này có cùng trọng số cho các cặp 1-1 và 0-0.

Chúng ta đã mô tả việc xây dựng các khoảng cách và các điểm tương đồng. Luôn luôn có thể xây dựng những điểm tương đồng từ khoảng cách.

Chúng ta có thể đặt:

$$\overline{S_{ik}} = \frac{1}{1 + d_{ik}}$$

Trong đó  $0 < \overline{S_{ik}} \leq 1$  là sự tương tự giữa các quan sát i và k,  $d_{ik}$  là khoảng cách tương ứng.

Tuy nhiên, các khoảng cách phải thỏa mãn các điều kiện và không phải lúc nào cũng được xây dựng từ các điểm tương đồng. Như Gower đã chỉ ra, điều này chỉ có thể được thực hiện nếu ma trận của các điểm tương tự là xác định không âm. Với điều kiện xác định không âm, và với độ tương tự tối đa được chia tỷ lệ sao cho  $s_{ii} = 1$  và

$$d_{ik} = \sqrt{2(1 - \overline{S_{ik}})}$$

có tính chất của khoảng cách.

## 2.2 Sự tương tự và các thước đo liên kết cho các cặp biến

Như vậy, chúng ta đã thảo luận về các biện pháp tương tự cho các quan sát. Trong một số ứng dụng, các biến, thay vì các quan sát, phải được nhóm lại. Các thước đo độ tương tự cho các

biến thường có dạng hệ số tương quan mẫu. Hơn nữa, trong một số ứng dụng phân cụm, các tương quan âm được thay thế bằng các giá trị tuyệt đối của chúng.

Khi các biến là nhị phân, dữ liệu lại có thể được sắp xếp dưới dạng một bảng dự phòng. Tuy nhiên, lần này, các biến, thay vì các quan sát, mô tả các danh mục. Đối với mỗi cặp biến, có n quan sát được phân loại trong bảng. Với mã hóa 0 và 1 thông thường, bảng sẽ trở thành như sau:

		Biến k		
		1	0	Total
Biến i	1	a	b	a + b
	0	c	d	c + d
Total		a + c	b + d	n = a + b + c + d

Ta có thể thấy ví dụ biến i bằng 1 và biến k bằng 0 đối với b trong số n quan sát. Công thức tương quan mômen sản phẩm thông thường được áp dụng cho các biến nhị phân trong bảng dự phòng trên cho công thức sau:

$$r = \frac{ad - bc}{[(a + d)(c + d)(a + c)(b + d)]^{\frac{1}{2}}}$$

Con số này có thể được coi là thước đo mức độ tương tự giữa hai biến. Hệ số tương quan có liên quan đến thống kê chi bình phương  $r^2 = \frac{\chi^2}{n}$  để kiểm tra tính độc lập của hai biến phân loại. Đối với n cố định, một sự tương tự lớn (hoặc tương quan) là phù hợp với sự hiện diện của sự phụ thuộc.

Trong bảng dự phòng trên, có thể phát triển các phép đo liên kết (hoặc độ tương tự) tương tự chính xác với các phép đo được liệt kê trong Bảng 3. Thay đổi duy nhất được yêu cầu là thay n (số quan sát) cho p (số biến).



### 2.3 Kết luận về sự tương tự

Để tóm tắt phần này, chúng ta lưu ý rằng có nhiều cách để đo mức độ tương tự giữa các cặp đối tượng. Có vẻ như đa phần mọi người sử dụng khoảng cách hoặc các hệ số trong bảng hệ số tương tự để phân cụm các quan sát và tương quan với các biến cụm. Tuy nhiên, đôi khi, đầu vào cho các thuật toán phân cụm có thể là các tần số đơn giản.

Xét ví dụ sau: (Đo lường sự giống nhau của các ngôn ngữ) Ý nghĩa của các từ thay đổi theo tiến trình lịch sử. Tuy nhiên, ý nghĩa của các số 1, 2, 3, ... đại diện cho một ngoại lệ dễ thấy. Vì vậy, so sánh đầu tiên của các ngôn ngữ có thể chỉ dựa trên các chữ số. Bảng dưới đưa ra 10 số đầu tiên bằng tiếng Anh, tiếng Ba Lan, tiếng Hungary và tám ngôn ngữ châu Âu hiện đại khác. (Chỉ những ngôn ngữ sử dụng bảng chữ cái La Mã mới được xem xét và các dấu trọng âm, dấu thanh, dấu phụ, v.v., v.v.) , Hà Lan, và Đức) rất giống nhau. tiếng Pháp, tiếng Tây Ban Nha và tiếng Ý thậm chí còn có thỏa thuận chặt chẽ hơn. Tiếng Hungary và tiếng Phần Lan dường như đứng riêng, và tiếng Ba Lan có một số đặc điểm của các ngôn ngữ trong mỗi nhóm con lớn hơn.

Bảng ngôn ngữ										
Anh (E)	Na uy (N)	Đan Mạch (Da)	Hà Lan (Du)	Đức (G)	Pháp (Fr)	Tây Ban Nha (Sp)	Ý (I)	Ba Lan (P)	Hung-ga-ri (H)	Phần Lan (Fi)
one	en	en	een	eins	un	uno	uno	jeden	egy	yksi
two	to	to	twee	zwei	deux	dos	due	dwa	ketto	kaksi
three	tre	tre	drie	drei	trois	tres	tre	trzy	harom	kolme
four	fire	fire	vier	vier	quatre	cuatro	quattro	cztery	negy	nelja
five	fem	fem	vijf	funf	cinq	cinco	cinque	piec	ot	viisi
six	seks	seks	zes	sechs	six	seis	sei	szesc	hat	kuusi
seven	sju	syv	zeven	sieben	sept	siete	sette	siedem	het	seitseman
eight	atte	otte	acht	acht	huit	ocho	otto	osiem	nyolc	kahdeksan
nine	ni	ni	negen	neun	neuf	nueve	nove	dziewiec	kilenc	yhdeksan
ten	ti	ti	tien	zehn	dix	diez	dieci	dziesiec	tiz	kymmenen

Bảng 5

Với mục đích minh họa, chúng ta có thể so sánh các ngôn ngữ bằng cách xem các chữ cái đầu tiên của các con số. ta gọi các từ cho cùng một số bằng hai ngôn ngữ khác nhau là đồng

nhất nếu chúng có cùng chữ cái đầu tiên và bất hòa nếu chúng không có.

Từ Bảng trên, bảng về sự phù hợp (tần số khớp với các chữ cái đầu tiên) cho các số từ 1-10 được đưa ra.

Bảng tần suất ngôn ngữ											
	E	N	Da	Du	G	Fr	Sp	I	P	H	Fi
E	10										
N	8	10									
Da	8	9	10								
Du	3	5	4	10							
G	4	6	5	5	10						
Fr	4	4	4	1	3	10					
Sp	4	4	5	1	3	8	10				
I	4	4	5	1	3	9	9	10			
P	3	3	4	0	2	5	7	6	10		
H	1	2	2	2	1	0	0	0	0	10	
Fi	1	1	1	1	1	1	1	1	1	2	10

Bảng 6

Chúng ta thấy rằng tiếng Anh và tiếng Na Uy có cùng một chữ cái đầu tiên cho 8 trong số 10 cặp từ. Các tần số còn lại được tính toán theo cách tương tự. Kết quả trong bảng tần số xác nhận ấn tượng trực quan ban đầu của chúng ta về bảng ngôn ngữ. Đó là, tiếng Anh, tiếng Na Uy, tiếng Đan Mạch, tiếng Hà Lan và tiếng Đức dường như tạo thành một nhóm. Tiếng Pháp, tiếng Tây Ban Nha, tiếng Ý và tiếng Ba Lan có thể được nhóm lại với nhau, trong khi tiếng Hungary và tiếng Phần Lan dường như đứng riêng. Trong các ví dụ từ trước cho đến nay, ta đã sử dụng ấn tượng trực quan về các biện pháp tương tự hoặc khoảng cách để tạo thành nhóm. Bây giờ chúng ta thảo luận về các phương pháp ít chủ quan hơn để tạo các cụm.

## 3 Thuật toán phân cụm phân cấp

### 3.1 Giới thiệu

#### 3.1.1 Phương pháp phân cụm theo cấp

Phân cụm theo cấp là một loại thuật toán thường dùng trong bài toán phân cụm dữ liệu. Chúng ta có thể xem xét tất cả các cách phân cụm nhưng việc đó sẽ tốn rất nhiều thời gian. Vì vậy thay vào đó chúng ta sẽ tìm các thuật toán phân cụm hợp lý mà không xem xét hết tất cả các cách phân cụm

Có 2 cách phân cụm theo mức

- Agglomerative clustering: Bắt đầu từ các cụm nhỏ nhất là mỗi cụm gồm 1 phần tử và gộp dần các phần tử lại cho đến khi thành 1 cụm duy nhất hay đến 1 ngưỡng cho trước.
- Divisive clustering: Bắt đầu từ cụm lớn nhất là 1 cụm chứa tất cả các phần tử và tách dần các cụm cho đến khi mỗi cụm còn 1 phần tử hay đến 1 ngưỡng cho trước.

Kết quả của cả 2 phương pháp đều có thể biểu diễn được bằng 1 biểu đồ 2 chiều gọi là dendrogram. Biểu đồ này thể hiện sự hợp hoặc phân tách các cụm ở từng mức khác nhau.

#### 3.1.2 Biểu diễn kết quả

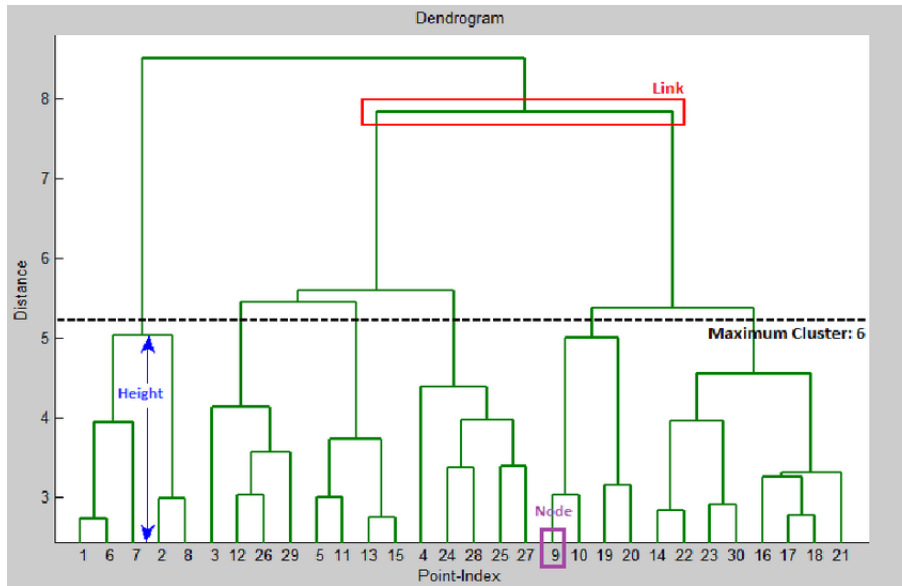
Ta có 2 cách biểu diễn kết quả cho các phương pháp phân cụm theo cấp

##### **Dendrogram**

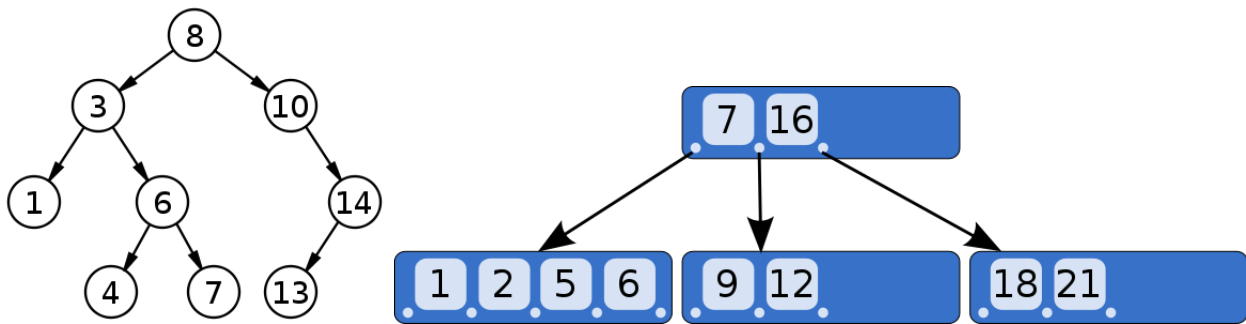
Biểu đồ có trục tung thể hiện độ lớn của mức gộp cụm, trục hoành gồm các phần tử của tập cần phân cụm, các đường nối vào nhau thể hiện việc gộp cụm xảy ra tại mức tương ứng dọc theo trục tung.

##### **Cây thuộc tính cụm**

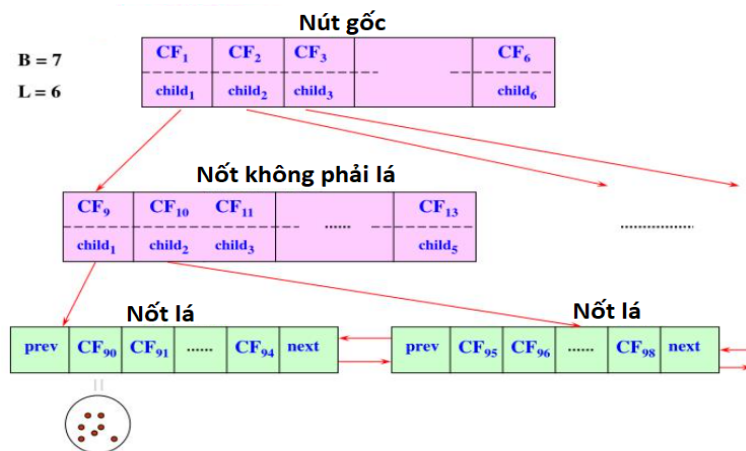
Cây thuộc tính cụm được trình bày trong thuật toán Birch có cấu trúc giống với *cây b+* thể hiện các cụm theo từng mức được phân theo các nhánh, nốt của cây.



Hình 3: hình ảnh giải thích cho dendrogram



Hình 4: cây nhị phân và b-cây



Hình 5: Cây thuộc tính cụm

### 3.2 Các phương pháp phân cụm tổng gộp theo cấp

Sau đây, ta sẽ trình bày các loại thuật toán phân cụm theo cách tổng gộp hay cụ thể là bắt đầu từ  $N$  cụm, ta tìm cách gộp dần cụm cho đến khi còn 1 cụm duy nhất.

Ta sẽ trình bày các phương pháp phân cụm sau:

- Single linkage (kết nối đơn)
- Complete linkage (kết nối toàn phần)
- Average linkage (kết nối trung bình)
- Ward's method
- BIRCH (Balanced Iterative Reducing and Clustering Using Hierarchies)
- ROCK (A robust clustering algorithm for categorical attributes) Using Dynamic Modeling

#### 3.2.1 Các loại kết nối

Với phương pháp phân cụm tổng gộp theo cấp sử dụng kết nối, tại mỗi bước ta sẽ gộp 2 cụm thành 1 cụm mới, ý tưởng chung sẽ là tiến hành gộp 2 cụm gần nhau nhất, trong đó tương ứng với 3 loại kết nối sẽ là 3 định nghĩa cho khoảng cách giữa 2 cụm  $U$  và  $V$ :

Kết nối đơn:

$$d(U, V) = \inf_{u \in U, v \in V} d(u, v)$$

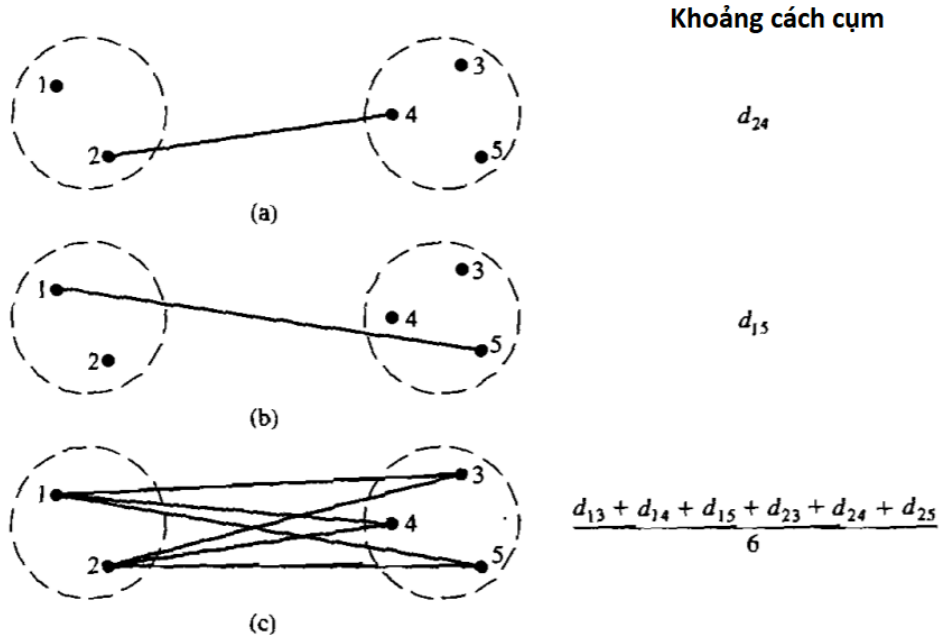
Kết nối toàn phần:

$$d(U, V) = \sup_{u \in U, v \in V} d(u, v)$$

Kết nối trung bình:

$$d(U, V) = \mathbb{E}d(u, v) \quad u \in U, v \in V$$

Hình ảnh minh họa biểu diễn cho 3 loại kết nối đơn, kết nối toàn phần, kết nối trung bình được thể hiện dưới đây



### 3.2.1.1 Thuật toán

Mã giả cho thuật toán phân cụm phân cấp sẽ được biểu diễn trong thuật toán 1.

---

#### Algorithm 1 Thuật toán chung cho phân cụm phân cấp

---

- 1: Bắt đầu với N cluster, mỗi cluster gồm 1 phần tử và ma trận  $N \times N$  khoảng cách (hay độ giống nhau) D.
  - 2: Tìm trong ma trận cặp 2 cụm gần nhau nhất (hay giống nhau nhất), gọi 2 cụm đó là U và V và khoảng cách là  $d_{UV}$ .
  - 3: Gộp 2 cụm U và V thành 1 cụm mới. Cập nhật ma trận D bằng cách xóa 2 dòng và cột tương ứng của U và V. Sau đó tính khoảng cách giữa cụm mới và các cụm cũ rồi thêm 1 dòng và cột tương ứng cho cụm mới và ma trận D.
  - 4: Lặp lại bước 2 và 3 tổng cộng N-1 lần.
- 

### 3.2.1.2 Kết nối đơn

Trong mỗi lần gộp cụm trong thuật toán được trình bày ở trên, ta cần cập nhật lại ma trận khoảng cách (hay độ tương đồng). Khi đó khoảng cách của cụm mới khi được gộp từ 2 cụm U và V đến cụm W là:

$$d_{(UV)W} = \min\{d_{UW}, d_{VW}\} \quad (1)$$

Để làm rõ hơn ta cùng xét 1 ví dụ.

**Ví dụ:** Xét 5 phần tử có ma trận khoảng cách D như sau:

$$\mathbf{D} = \{d_{ik}\} = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{bmatrix} 0 & & & & \\ 9 & 0 & & & \\ 3 & 7 & 0 & & \\ 6 & 5 & 9 & 0 & \\ 11 & 10 & \textcircled{2} & 8 & 0 \end{bmatrix} \end{matrix}$$

Ta thấy 2 là khoảng cách nhỏ nhất trong D, ứng với 2 cụm là 3 và 5. Tiến hành gộp cụm 3 và 5 và cập nhật D theo công thức 1 ta được:

$$\begin{matrix} & \begin{matrix} (35) & 1 & 2 & 4 \end{matrix} \\ \begin{matrix} (35) \\ 1 \\ 2 \\ 4 \end{matrix} & \begin{bmatrix} 0 & & & \\ \textcircled{3} & 0 & & \\ 7 & 9 & 0 & \\ 8 & 6 & 5 & 0 \end{bmatrix} \end{matrix}$$

Làm tương tự, gộp 2 và 4 ta được:

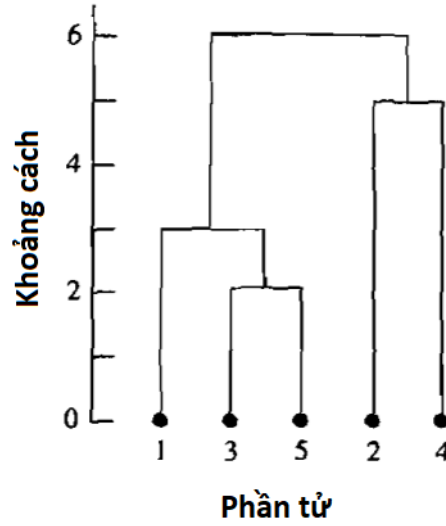
$$\begin{matrix} & \begin{matrix} (135) & 2 & 4 \end{matrix} \\ \begin{matrix} (135) \\ 2 \\ 4 \end{matrix} & \begin{bmatrix} 0 & & \\ 7 & 0 & \\ 6 & \textcircled{5} & 0 \end{bmatrix} \end{matrix}$$

Ta còn lại 2 cụm cuối cùng với ma trận D như sau:

$$\begin{matrix} & \begin{matrix} (135) & (24) \end{matrix} \\ \begin{matrix} (135) \\ (24) \end{matrix} & \begin{bmatrix} 0 & \\ \textcircled{6} & 0 \end{bmatrix} \end{matrix}$$

Sau khi gộp 2 cụm cuối cùng, ta còn 1 cụm duy nhất gồm tất cả các phần tử và do đó, thuật toán kết thúc.

Từ đồ thị ta thấy rõ được các cụm, cũng như từng bước gộp cụm xảy ra ở các ngưỡng tăng dần: ngưỡng  $d = 2$  khi gộp 3 và 5,  $d = 3$  khi gộp 1 và (3,5), ...



Hình 6: Biểu đồ thể hiện thuật toán phân cụm kết nối đơn

### 3.2.1.3 Kết nối toàn phần và kết nối trung bình

Đối với complete linkage và average linkage, thuật toán được thực hiện một cách hoàn toàn tương tự với cách cập nhật lại ma trận D như sau:

Kết nối toàn phần:

$$d_{(UV)W} = \max\{d_{UW}, d_{VW}\} \quad (2)$$

Kết nối trung bình:

$$d_{(UV)W} = \frac{|U||W|d(U, W) + |V||W|d(V, W)}{|UV||W|} \quad (3)$$

Với ký hiệu  $|W|$  là số lượng phần tử của cụm W. Chú ý: từ công thức cập nhật cho 3 loại kết nối ta thấy có duy nhất kết quả của kết nối trung bình có thể bị thay đổi khi thay 1 metric bảo toàn thứ tự.

### 3.2.2 Phân cụm theo mức của Ward

Phương pháp phân cụm của Ward dựa trên việc cực tiểu lượng thông tin mất mát trong các cụm. Lượng thông tin mất mát thường được định nghĩa bằng việc phương sai của cụm. Cụ thể, ta có định nghĩa ESS khi của 1 cụm X như sau:

$$ESS(X) = \sum_{x \in X} (x - \bar{x})'(x - \bar{x}) \quad (4)$$

Trong đó  $\bar{x}$  là trọng tâm hay trung bình của cụm. Phương pháp của Ward dựa trên quan niệm rằng các cụm trong quan sát nhiều chiều có xu hướng xấp xỉ hình dạng elliptic.



Phương pháp của Ward là 1 tiền đề cho các phương pháp phân cụm không theo mức khi tối ưu 1 tiêu chuẩn nào đó để chia dữ liệu thành các cụm, trong đó ta cực tiểu tổng lượng thông tin mất mát khi chia tập dữ liệu thành k cụm  $X_k$

$$\min \sum_{i=1}^K ESS(X_k)$$

Kí hiệu  $m_X$  là trọng tâm của cụm X, lượng thông tin mất mát khi gộp 2 cụm A và B là:

$$\begin{aligned} d(A, B) &= \sum_{i \in AB} \|\vec{x}_i - \vec{m}_{AB}\|^2 - \sum_{i \in A} \|\vec{x}_i - \vec{m}_A\|^2 - \sum_{i \in B} \|\vec{x}_i - \vec{m}_B\|^2 \\ &= \frac{|A||B|}{|AB|} \|\vec{m}_A - \vec{m}_B\|^2 \end{aligned}$$

Từ đó ta tìm được công thức cập nhật ma trận khoảng cách cho phương pháp của Ward.

### 3.2.2.1 Thuật toán Lance-Williams

Thuật toán Lance-Williams là thuật toán tổng quát sử dụng trong 1 nhóm các phương pháp tổng gộp theo cấp được thể hiện qua công thức đệ quy được sử dụng trong việc cập nhật ma trận khoảng cách hay độ giống nhau có dạng như sau:

$$d(UV, W) = \alpha(U)d(U, W) + \alpha(V)d(V, W) + \beta d(U, V) + \gamma |d(U, W) - d(V, W)| \quad (5)$$

Phương pháp	$\alpha_i$	$\alpha_j$	$\beta$	$\gamma$
Kết nối đơn	0.5	0.5	0	-0.5
Kết nối toàn phần	0.5	0.5	0	0.5
Trung bình nhóm	$\frac{n_i}{n_i + n_j}$	$\frac{n_j}{n_i + n_j}$	0	0
Trung bình nhóm trọng số	0.5	0.5	0	0
Trọng tâm	$\frac{n_i}{n_i + n_j}$	$\frac{n_j}{n_i + n_j}$	$\frac{-n_i \cdot n_j}{(n_i + n_j)^2}$	0
Ward	$\frac{n_i + n_k}{(n_i + n_j + n_k)}$	$\frac{n_j + n_k}{(n_i + n_j + n_k)}$	$\frac{-n_k}{(n_i + n_j + n_k)}$	0

Bảng 7: Bảng công thức cho 1 số phương pháp

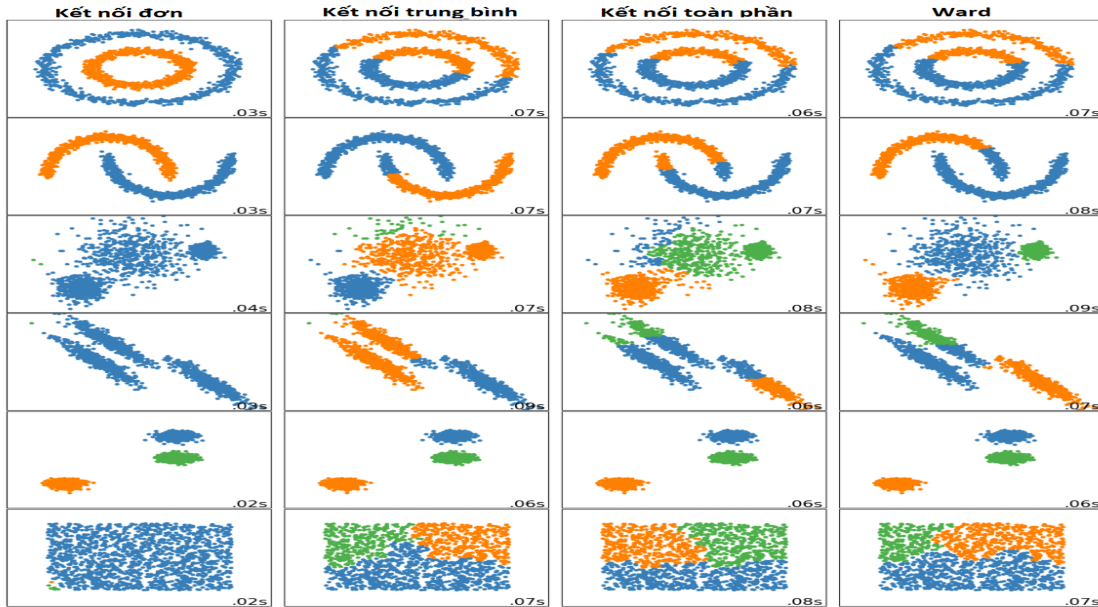
### 3.2.2.2 Cài đặt phương pháp của Ward

Theo công thức cập nhật ở trên, việc thực hiện phương pháp của Ward hoàn toàn tương tự 3 loại kết nối đã nêu:

$$d(UV, W) = \frac{|UW|}{|UVW|} d(U, W) + \frac{|VW|}{|UVW|} d(V, W) - \frac{|C|}{|UVW|} d(U, V) \quad (6)$$

### 3.2.2.3 Các phương pháp phân cụm kết nối và Ward

Dưới đây là một số kết quả của các phương pháp phân cụm theo kết nối và Ward. Tùy thuộc theo hình dáng của tập dữ liệu mà ta cần chọn cách phân cụm khác nhau sao cho phù hợp với mong muốn.



Hình 7: Kết quả cho 4 phương pháp

#### Một số nhận xét

- Các phương pháp đã giới thiệu tốn nhiều thời gian với dữ liệu lớn do độ phức tạp thuật toán  $O(n^2)$ .
- Khó thực hiện trên bộ dữ liệu lớn khi bộ nhớ bị giới hạn khi thuật toán yêu cầu lưu trữ thông tin về toàn bộ tập dữ liệu để thực hiện (ma trận khoảng cách)

### 3.2.3 BIRCH

BIRCH [14] (Balanced Iterative Reducing and Clustering Using Hierarchies) là một thuật toán phân cụm có thể thực hiện trên tập dữ liệu lớn bằng cách khởi tạo 1 tập nhỏ thể hiện tóm tắt thông tin của tập dữ liệu. Sau đó ta thực hiện phân cụm trên tập này thay vì toàn bộ dữ liệu.

BIRCH thường được sử dụng để hỗ trợ các thuật toán phân cụm khác, tuy nhiên nó có 1 điểm yếu lớn đó là chỉ sử dụng được trên dữ liệu có khoảng cách (trên không gian Euclid) mà không sử dụng được cho dữ liệu dạng phân lớp.

- Bước 1: Duyệt dữ liệu để xây dựng cây thuộc tính cụm bảo toàn 1 số tính chất, cấu trúc của dữ liệu
- Bước 2: Tiến hành phân cụm các nốt lá của cây.

### 3.2.3.1 Một số khái niệm

Cho 1 tập  $X$  có  $N$  phần tử trong không gian Euclid, ta có các định nghĩa sau

- Bán kính:  $R = \sqrt{\frac{1}{N} \sum_{x \in X} (x - m_x)^2}$
- Đường kính:  $D = \sqrt{\frac{1}{N(N-1)} \sum_{x, y \in X} (x - y)^2}$
- Momen bậc 1:  $LS = \sum_{x \in X} x$
- Momen bậc 2:  $SS = \sum_{x \in X} x^2$

**Thuộc tính cụm:** Với mỗi cụm ta sẽ lưu 3 thuộc tính cụm (clustering feature) tương ứng là momen bậc 0, 1, 2:

$$CF_X = (N_X, LS_X, SS_X) \quad (7)$$

Để thấy được khi gộp cụm 1 và cụm 2 thì ta có:

$$CF_{12} = CF_1 + CF_2 = (N_1 + N_2, L_1 + L_2, SS_1 + SS_2)$$

thuộc tính cụm cho phép ta biết thông tin về cụm và có thể được sử dụng để tính các khoảng cách cần thiết cho việc sử dụng các thuật toán phân cụm khác như khoảng cách giữa các trọng tâm, kết nối trung bình, bán kính, đường kính, lượng tăng phương sai, ...

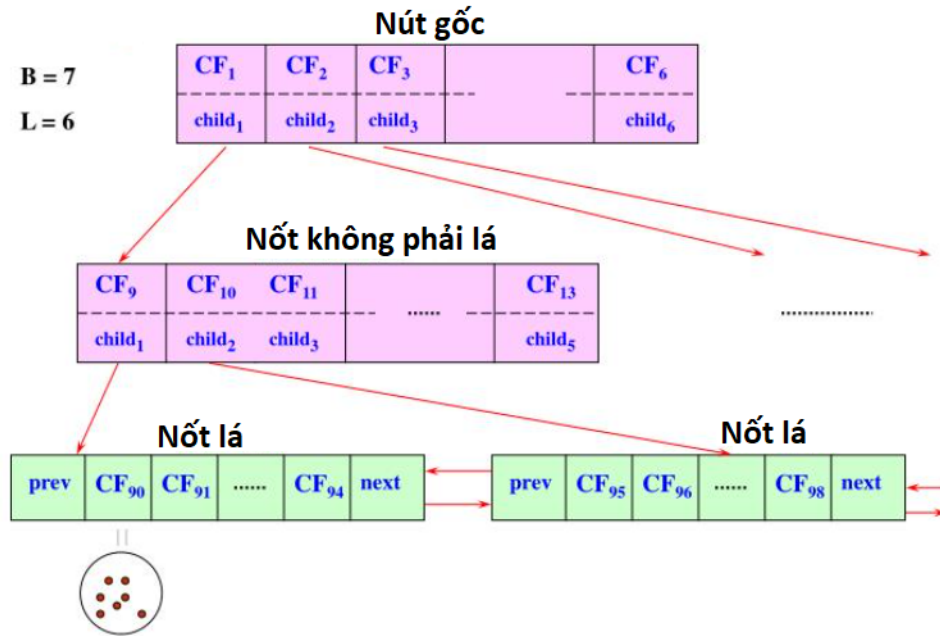
**Cây thuộc tính cụm:** Cây thuộc tính cụm (clustering feature hay CFT) là 1 cách thể hiện tập dữ liệu trong đó

- Mỗi nốt cha là 1 cụm gồm các cụm con với tối đa  $B$  (hệ số phân nhánh) cụm con.
- Mỗi nốt lá là 1 tập các cụm (tối đa  $L$  cụm) trong đó các cụm gồm các phần tử với số phần tử tối đa là  $L$  và đường kính tối đa là  $T$ .

### 3.2.3.2 Thuật toán Ta thực hiện các bước sau:

#### Bước 1

- Ta sẽ thực hiện duyệt lần lượt toàn bộ dữ liệu và xây dựng cây thuộc tính cụm. Gọi nốt đang xét là  $d$ , bắt đầu từ nút gốc ta chọn nốt gần nhất với  $d$  lần lượt dọc theo cây.



Hình 8: Cây thuộc tính cụm

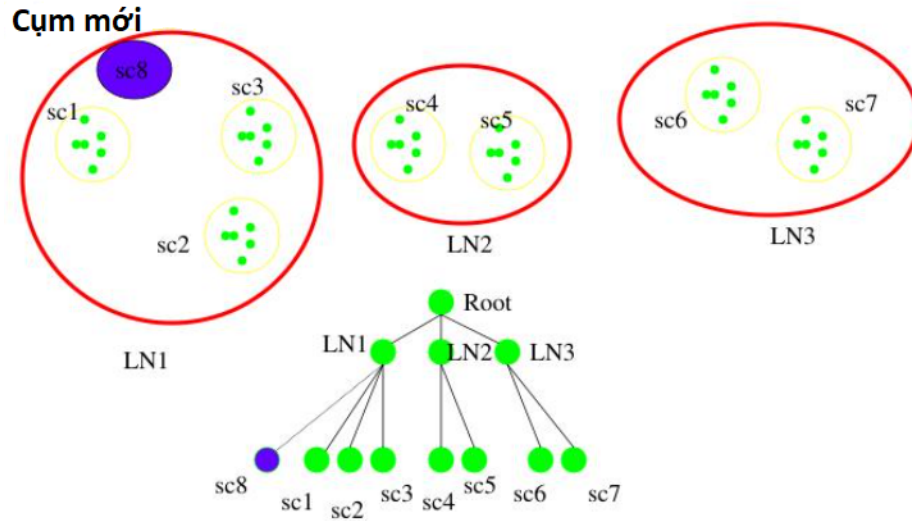
- Tại nốt lá, ta thêm d vào cụm gần nhất hoặc nếu không được, ta thêm 1 cụm mới tại nốt lá đó, nếu vẫn không được, ta tiến hành tách nốt.
- Tiến hành cập nhật các lớp cha nếu xảy ra tách.

## Bước 2

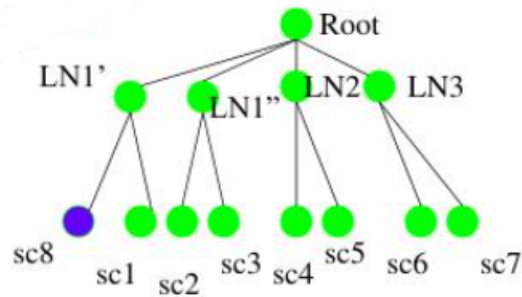
- Sử dụng các thuật toán phân cụm mà có thể thực hiện được khi biết thuộc tính cụm (kết nối đơn, Ward's method, ...)
- Duyệt toàn bộ dữ liệu để phân từng phần tử vào cụm tương ứng với cụm của nốt lá.

### 3.2.3.3 Ví dụ:

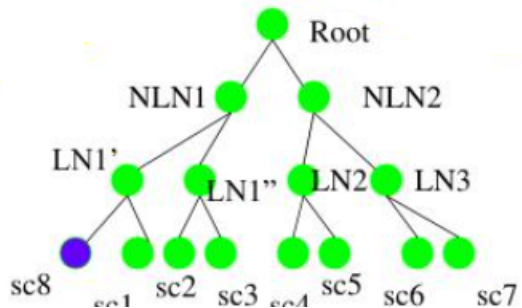
Giả sử ta cần thêm sc8 vào cây và giả sử số phần tử tối đa cùng hệ số phân nhánh là 3.



Do số phần tử tối đa là 3 nên ta tách nốt lá LN1



Do hệ số phân nhánh là 3 nên ta tách nốt gốc



### 3.2.4 ROCK

ROCK [4] (A robust clustering algorithm for categorical attributes) là một phương pháp phân cụm được xây dựng đặc biệt cho dữ liệu dạng phân lớp.

### 3.2.4.1 Một số khái niệm

Lân cận: ta định nghĩa lân cận của 1 phần tử  $x$  là tập các phần sao cho độ tương đồng của của phần tử đó với  $x$  lớn hơn 1 ngưỡng  $\theta$  cố định, hay  $y$  là lân cận của  $x$  khi và chỉ khi điều kiện sau thỏa mãn

$$\text{sim}(x, y) \geq \theta$$

trong đó  $\text{sim}$  là hàm tính độ tương đồng giữa 2 phần tử. Trong trường hợp dữ liệu là thông tin giao dịch mua hàng có mỗi giao dịch là 1 tập các mặt hàng ta có thể định nghĩa hàm này theo chỉ số jaccard:

$$\text{sim}(x, y) = \frac{|X \cap Y|}{|X \cup Y|} \quad (8)$$

Chú ý: lân cận của 1 phần tử có chứa chính nó.

**Liên kết:** ta định nghĩa  $\text{link}(x, y)$  là số lân cận chung của  $x$  và  $y$ . Khi đó xét tập dữ liệu gồm  $N$  phần tử thì ma trận lân cận  $A$  có kích thước  $N \times N$  với  $A = [a_{ij}] = [\mathbb{I}(\text{sim}(x_i, x_j) \geq \theta)]_{ij}$

Ta dễ dàng chứng minh được ma trận liên kết  $L = [\text{link}(x_i, x_j)]$  thỏa mãn:

$$L = A^2$$

Từ định nghĩa ta thấy được là 2 phần tử nếu liên kết giữa 2 phần tử là lớn thì khả năng cao chúng sẽ thuộc vào 1 cụm.

Tư tưởng của phương pháp là đi tìm cách phân cụm sao cho các phần tử trong 1 cụm có nhiều liên kết với nhau, hay lượng liên kết là lớn nhất. Tuy nhiên ta không muốn gộp tất cả với nhau hay gộp các phần tử có ít liên kết với nhau, vậy nên ta sẽ tối đa tổng lượng liên kết so với kì vọng, cụ thể ta cố gắng cực đại hóa hàm mục tiêu sau:

$$E_l = \sum_i^k n_i \sum_{x, y \in C_i} \frac{\text{link}(x, y)}{n_i^{(1+2f(\theta))}} \quad (9)$$

trong đó  $k$  là số lượng cụm,  $e(C_i) := n_i^{(1+2f(\theta))}$  là lượng liên kết kì vọng trong 1 cụm gồm  $n_i$  phần tử của cụm  $C_i$ . Hàm  $f(\theta)$  phụ thuộc vào dữ liệu, tuy nhiên người ta đã thấy được rằng lượng liên kết kì vọng sẽ có dạng như ở hàm  $e$ .

### 3.2.4.2 Thuật toán

Bắt đầu với  $N$  cụm riêng biệt, ta tiến hành gộp 2 cụm giống nhau nhất với độ giống nhau giữa 2 cụm  $U$  và  $V$  được định nghĩa như sau

$$g(U, V) = \frac{\text{link}(U, V)}{e(UV) - e(U) - e(V)} \quad (10)$$

trong đó đại lượng  $e(UV) - e(U) - e(V)$  có ý nghĩa là lượng liên kết kì vọng giữa 2 cụm U và V. Việc thực hiện ROCK sẽ tương tự với 3 loại linkage đã trình bày, tuy nhiên ta phải lưu và cập nhật 2 thông tin là liên kết và số lượng phần tử chứ không cập nhật trực tiếp độ giống nhau qua g.

### 3.3 Phân cụm phân tách theo cấp

#### 3.3.1 DIANA

Phân cụm phân tách thực hiện ngược so với phân cụm tổng gộp. Ở mỗi bước, ta cần tìm 1 phương pháp để tách 1 tập đang có thành 2 tập nhỏ hơn. Việc xét hết tất cả các cách tách là không khả thi với dữ liệu tương đối lớn. Cụ thể đối với 1 tập N phần tử thì ta cần xét  $2^{N-1} - 1$  cách tách nó thành 2 tập con. Ta giới thiệu DIANA là 1 trong các phương pháp phân cụm phân tách thuộc loại này.

##### 3.3.1.1 Một số khái niệm

Với mỗi phần tử x, ta xét lượng khác biệt trung bình của nó so với các phần tử khác trong 1 cụm U như sau

$$a(x, U) = \frac{1}{|U \setminus \{x\}|} \sum_{y \in U} d(x, y) \quad (11)$$

Ý tưởng chung sẽ là giữa 2 tập thì 1 phần tử nên ở tập có lượng khác biệt trung bình so với nó nhỏ hơn.

##### 3.3.1.2 Thuật toán

Bắt đầu với 1 cụm duy nhất gồm tất cả các phần tử, ta thực hiện các bước sau.

- Bước 1: Chọn ra cụm U có đường kính lớn nhất để tách.
- Bước 2: Ta xét việc tách 1 cụm U thành 2 cụm  $U_1$  và  $U_2$ 
  - Bước 1: Cho  $U_1 = U$  và  $U_2 = \emptyset$ , ta tiến hành chuyển phần tử trong  $U_1$  có lượng khác biệt trung bình lớn nhất so với  $U_1$  qua  $U_2$ .
  - Bước 2: Chuyển phần tử x có lượng chênh lệch  $a(x, U_1) - a(x, U_2)$  lớn nhất qua  $U_2$
  - Ta thực hiện bước 2 cho đến khi  $a(x, U_1) - a(x, U_2) < 0 \forall x \in U_1$
- Bước 3: Bổ sung 2 cụm vừa tách được  $U_1$  và  $U_2$  vào tập các cụm và lặp lại bước 1 cho đến khi tất cả các cụm có duy nhất 1 phần tử hay đạt ngưỡng tất cả các cụm có đường kính nhỏ hơn 1 số cho trước.

### 3.3.1.3 Ví dụ

Xét 5 phần tử có ma trận khoảng cách như sau:

$$\begin{matrix} & \begin{matrix} a & b & c & d & e \end{matrix} \\ \begin{matrix} a \\ b \\ c \\ d \\ e \end{matrix} & \left[ \begin{array}{ccccc} 0.0 & 2.0 & 6.0 & 10.0 & 9.0 \\ 2.0 & 0.0 & 5.0 & 9.0 & 8.0 \\ 6.0 & 5.0 & 0.0 & 4.0 & 5.0 \\ 10.0 & 9.0 & 4.0 & 0.0 & 3.0 \\ 9.0 & 8.0 & 5.0 & 3.0 & 0.0 \end{array} \right] \end{matrix}$$

Ta tính lượng khác biệt trung bình của các phần tử trong cụm

Phần tử	Khoảng cách trung bình đến các phần tử khác
<i>a</i>	$(2.0 + 6.0 + 10.0 + 9.0)/4 = 6.75$
<i>b</i>	$(2.0 + 5.0 + 9.0 + 8.0)/4 = 6.00$
<i>c</i>	$(6.0 + 5.0 + 4.0 + 5.0)/4 = 5.00$
<i>d</i>	$(10.0 + 9.0 + 4.0 + 3.0)/4 = 6.50$
<i>e</i>	$(9.0 + 8.0 + 5.0 + 3.0)/4 = 6.25$

Tiến hành tách phần tử *a* ra nhóm khác và tính sự khác biệt trung bình của mỗi phần tử so với 2 nhóm

Phần tử	Khoảng cách trung bình đến các phần tử còn lại	Khoảng cách trung bình đến nhóm mới	Lượng sai khác
<i>b</i>	$(5.0 + 9.0 + 8.0)/3 \approx 7.33$	2.00	5.33
<i>c</i>	$(5.0 + 4.0 + 5.0)/3 \approx 4.67$	6.00	-1.33
<i>d</i>	$(9.0 + 4.0 + 3.0)/3 \approx 5.33$	10.00	-4.67
<i>e</i>	$(8.0 + 5.0 + 3.0)/3 \approx 5.33$	9.00	-3.67

Tương tự tách phần tử *b* và tính

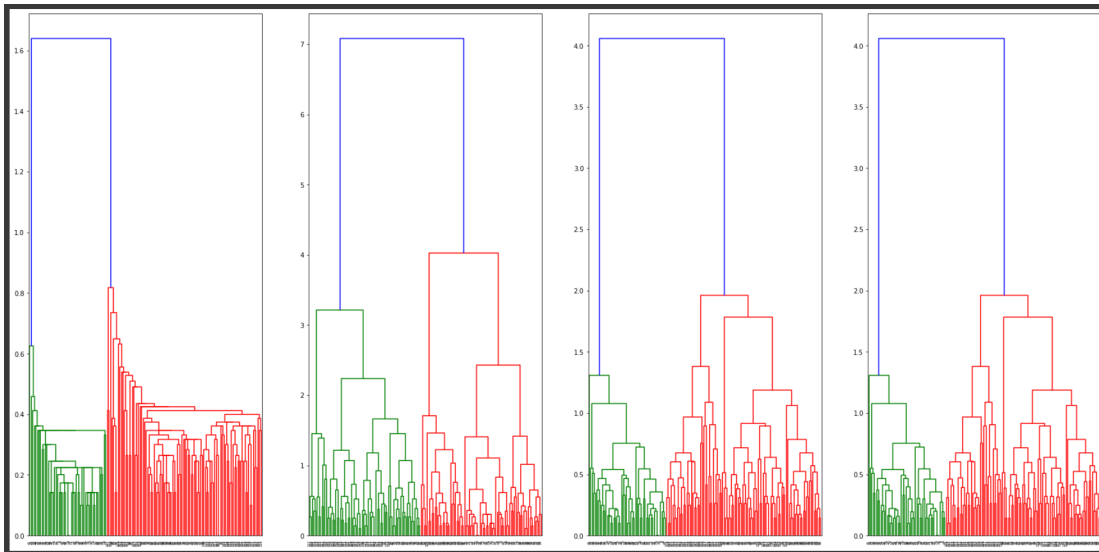


Phần tử	Khoảng cách trung bình đến các phần tử còn lại	Khoảng cách trung bình đến nhóm mới	Lượng sai khác
$c$	$(4.0 + 5.0)/2 = 4.50$	$(6.0 + 5.0)/2 = 5.50$	$-1.00$
$d$	$(4.0 + 3.0)/2 = 3.50$	$(10.0 + 9.0)/2 = 9.50$	$-6.00$
$e$	$(5.0 + 3.0)/2 = 4.00$	$(9.0 + 8.0)/2 = 8.50$	$-4.50$

Ta dừng do tất cả lượng chênh lệch  $\leq 0$

### 3.4 Kết quả thực nghiệm

Sử dụng 3 thuật toán phân cụm theo kết nối và phương pháp của Ward ta được biểu đồ dendrogram như hình dưới đây. Từ trái sang phải lần lượt là là kết nối đơn, kết nối toàn phần, kết nối trung bình, phương pháp của Ward.

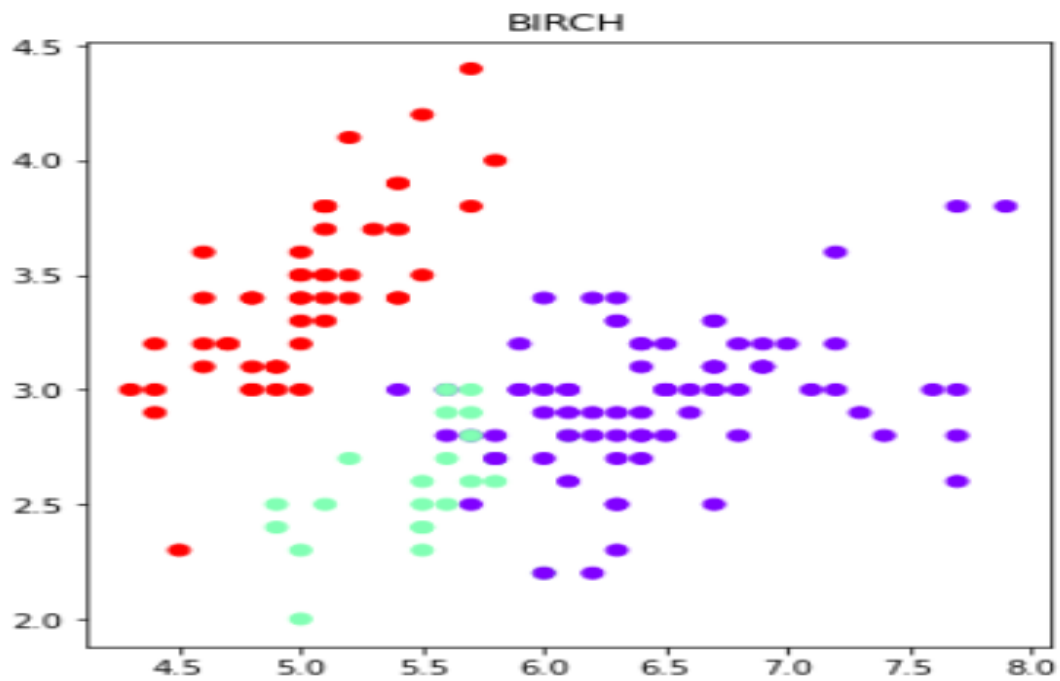


Hình 9: Biểu đồ dendrogram cho 4 phương pháp

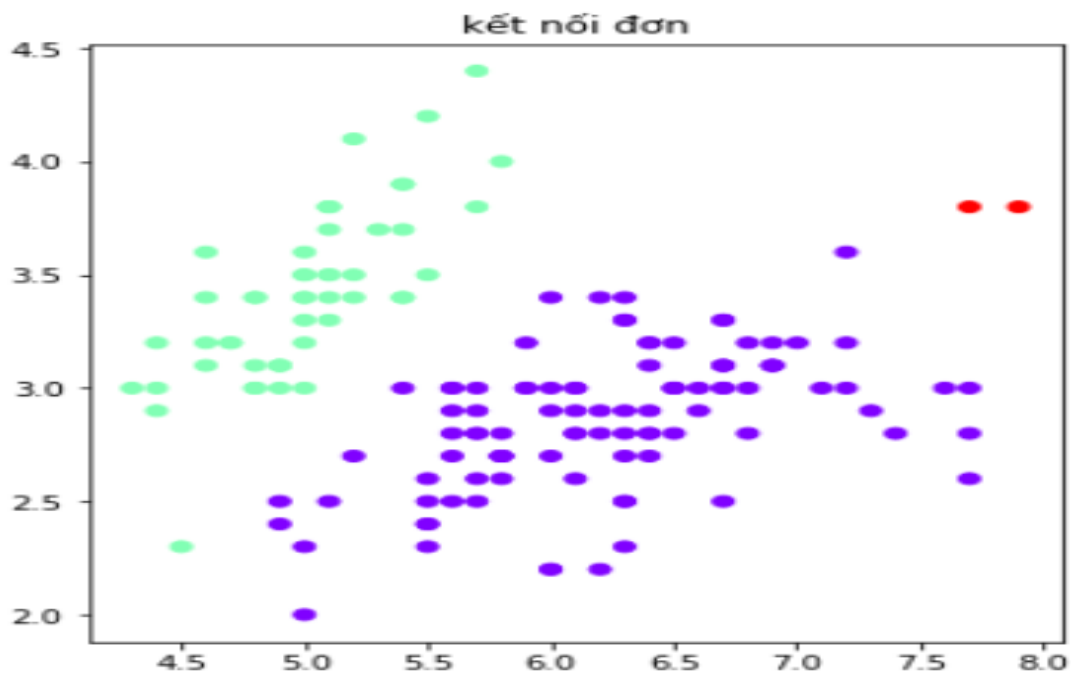
Nói chung tại mức có 3 cụm thì các phương pháp này cho việc phân cụm tương đối giống nhau trừ kết nối đơn. Tại các mức khác, một đặc điểm thường thấy của các phương pháp phân cụm kết nối được thể hiện khi số lượng phần tử trong mỗi nhóm có thể rất khác biệt nhau đặc biệt là ở kết nối đơn, cụ thể có cụm chỉ có 1 phần tử ở mức có số cụm là 7.

Để thể hiện việc phân cụm của các phương pháp trên đồ thị, ta chọn ra 2 đặc tính đầu tiên

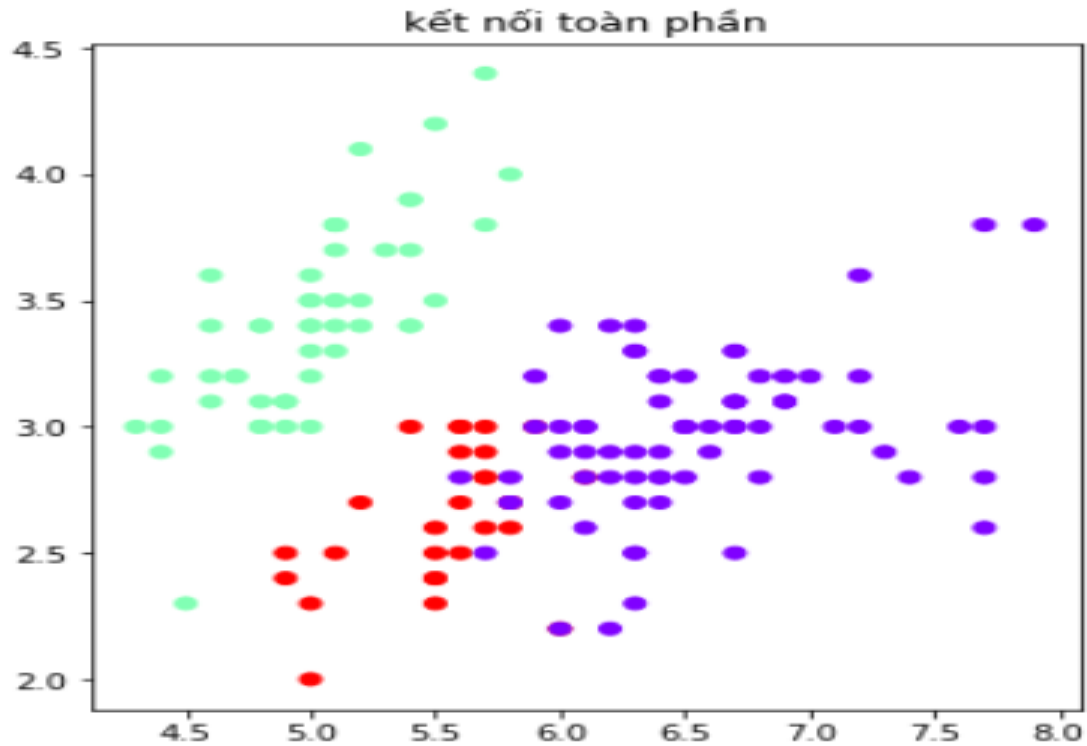
của điểm dữ liệu và thể hiện các cụm khác nhau bởi các cụm khác nhau, số lượng cụm trong các phương pháp là 3.



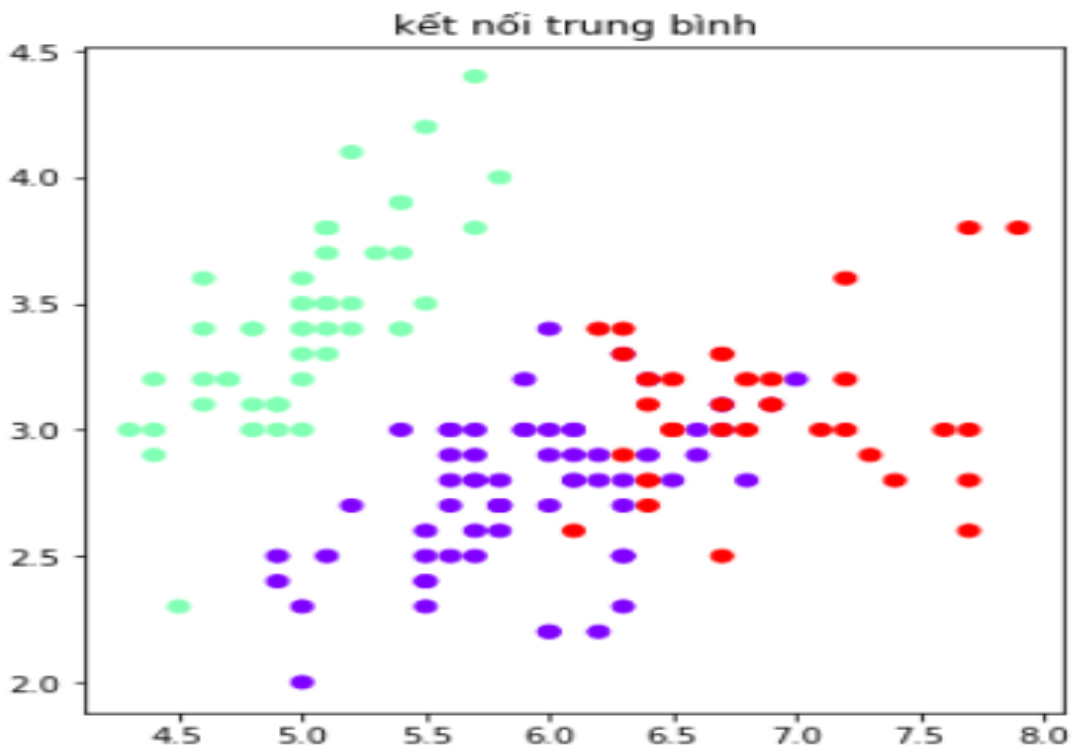
Hình 10: Kết quả chạy thuật toán Birch



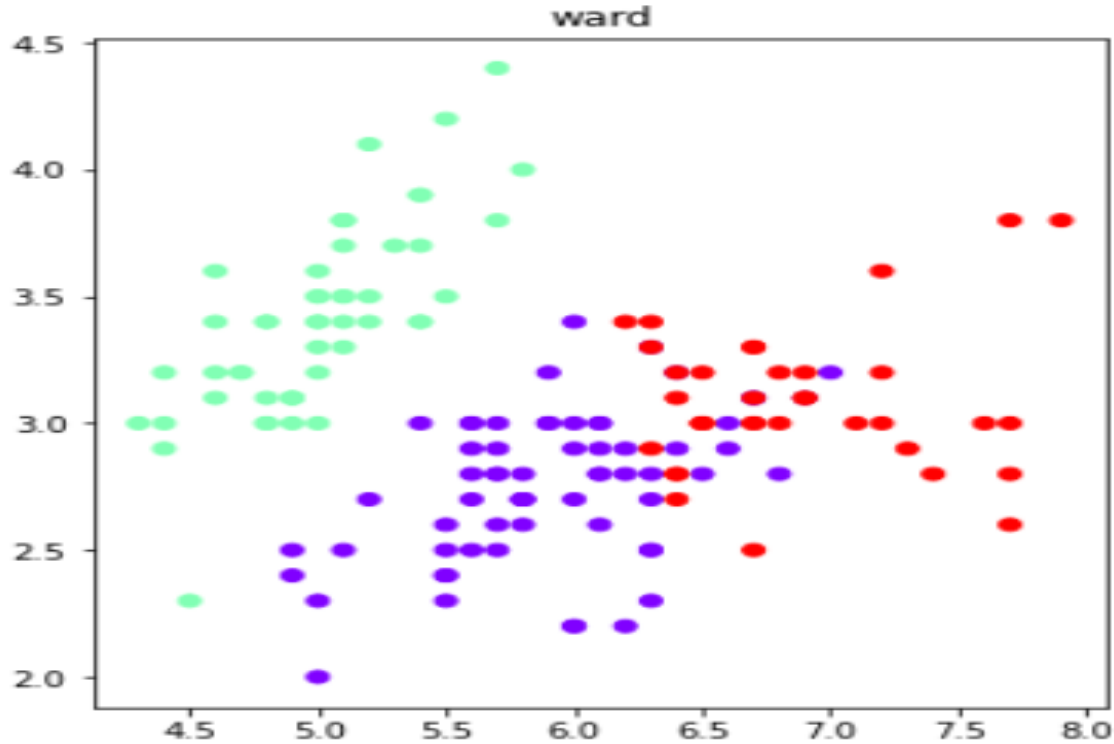
Hình 11: Kết quả chạy kết nối đơn



Hình 12: Kết quả chạy kết nối toàn phần



Hình 13: Kết quả chạy kết nối trung bình



Hình 14: Kết quả chạy phương pháp của Ward

Các kết quả thu được đã ở trong các phương pháp (trừ kết nối đơn nhạy cảm với nhiễu) đã phân được các cụm mà các điểm dữ liệu trong cùng một loài hoa có tỉ lệ trong cùng một cụm khá cao, nhất là (setosa) ở phần góc trên bên trái như trên đồ thị. Việc đó cho thấy các thuật toán này đã tương đối phù hợp để sử dụng với dữ liệu mà ta đang xét. Thuật toán ROCK được sử dụng cho dữ liệu dạng phân cấp nên không sử dụng trên bộ dữ liệu này.

### 3.5 *Kết luận chung*

Có rất nhiều phương pháp tổng gộp ngoài các phương pháp trên, tuy nhiên các phương pháp này đều có cấu trúc như thuật toán đã trình bày.

Với mỗi bài toán cụ thể, ta nên thử nghiệm các phương pháp phân cụm khác nhau, các cách đánh giá khoảng cách khác nhau. Nếu các cách đưa ra kết quả tương đối giống nhau, từ đó ta có thể tiến tới tổng hợp để đưa ra được 1 cách tự nhiên để nhóm các phần tử.

Đôi khi thuật toán có thể được kiểm tra tính ổn định bằng cách thêm các nhiễu. Nếu các cụm phân biệt rõ ràng với nhau, việc phân cụm trước và sau khi thêm nhiễu sẽ không quá khác biệt.

Trong 1 số thuật toán phân cụm theo mức, ở 1 mức có thể xảy ra các nhiễu cặp khoảng

cách nhỏ nhất, từ đó tạo ra nhiều cách phân cụm khác nhau. Việc xảy ra nhiều nghiệm không có nghĩa là thuật toán không tốt, nhưng chúng ta cần biết và hiểu điều này để có thể đánh giá biểu đồ dendrogram một cách hợp lý.

## 4 Thuật toán phân cụm không phân cấp

### 4.1 Phương pháp luận

#### 4.1.1 Thuật toán K-Means

##### 4.1.1.1 Giới thiệu bài toán

Thuật toán phân cụm không phân cấp được thiết kế để phân cụm bộ dữ liệu ban đầu thành  $K$  cụm (clusters) và từ  $K$  cụm đầu ra tiến hành quản lý bộ dữ liệu. Do thuật toán này không cần xác định ma trận khoảng cách giữa các điểm dữ liệu và các điểm dữ liệu cơ sở không cần phải được lưu trữ trong quá trình chạy thuật toán nên nó có thể được sử dụng với bộ dữ liệu lớn hơn nhiều so với thuật toán phân cụm phân cấp.

Một trong các thuật toán phân cụm không phân cấp nổi tiếng và phổ biến nhất đó chính là thuật toán K-Means và phiên bản cải tiến của nó: Thuật toán K-Means++. Có 2 hướng tiếp cận để khởi tạo thuật toán K-Means đó là:

1. Chia bộ dữ liệu thành các cụm sẵn
2. Khởi tạo các điểm trung tâm (center) của từng cụm

##### 4.1.1.2 Mô hình tổng quát

Giả sử có  $N$  điểm dữ liệu là:  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N] \in \mathbb{R}^{d \times N}$  và  $K < N$  là số cụm mà ta muốn phân chia. Chúng ta cần tìm các điểm trung tâm  $\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K \in \mathbb{R}^{d \times 1}$  và nhãn của các điểm dữ liệu (nhãn biểu thị cho dữ liệu thuộc cụm nào). Với mỗi điểm dữ liệu  $\mathbf{x}_i$  và  $\mathbf{y}_i = [y_{i1}, y_{i2}, \dots, y_{iK}]$  là vector nhãn của nó, trong đó  $\mathbf{x}_i$  được phân vào cụm  $k$  thì  $y_{ik} = 1$  và  $y_{ij} = 0 \ \forall j \neq k$ .

Nếu một điểm dữ liệu  $\mathbf{x}_i$  được phân vào cụm có điểm trung tâm là  $\mathbf{c}_k$  thì sai số là  $(\mathbf{x}_i - \mathbf{c}_k)$ . Ta cần làm cực tiểu hóa giá trị hàm sai số:

$$y_{ik} \|\mathbf{x}_i - \mathbf{c}_k\|_2^2 = \sum_{j=1}^K y_{ij} \|\mathbf{x}_i - \mathbf{c}_k\|_2^2$$

Khi đó sai số cho toàn bộ bộ dữ liệu là:  $L(\mathbf{Y}, \mathbf{C}) = \sum_{i=1}^N \sum_{j=1}^K y_{ij} \|\mathbf{x}_i - \mathbf{c}_j\|_2^2$ .

Trong đó  $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N]$  là ma trận nhãn,  $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K]$  là ma trận các điểm trung tâm.

Để tối ưu hàm mất mát thì ta cần tìm các ma trận  $\mathbf{Y}, \mathbf{C}$  sao cho:

$$\mathbf{Y}, \mathbf{C} = \underset{\mathbf{Y}, \mathbf{C}}{\operatorname{argmin}} \sum_{i=1}^N \sum_{j=1}^K y_{ij} \|\mathbf{x}_i - \mathbf{c}_j\|_2^2 \quad (1)$$

$$v.d.k \ y_{ij} \in \{0; 1\} \ \forall i, j; \ \sum_{j=1}^K y_{ij} = 1 \ \forall i$$

Bài toán (1) thuộc loại mix-integer programming (điều kiện biến là số nguyên) khá khó tìm nghiệm tối ưu toàn cục, nhưng ta vẫn có thể tìm được nghiệm gần đúng hoặc các nghiệm tối ưu địa phương. Một phương pháp đơn giản thường dùng để giải bài toán này là xen kẽ tìm  $\mathbf{Y}$ ,  $\mathbf{C}$  khi có một yếu tố cố định. Đây là một thuật toán lặp, cũng là kỹ thuật phổ biến khi giải bài toán tối ưu. Chúng ta sẽ lần lượt giải quyết hai bài toán sau đây:

**TH1** Giả sử đã tìm được các điểm trung tâm, hãy tìm các vector nhãn cho từng điểm sao cho hàm mất mát nhỏ nhất.

$$\mathbf{y}_i = \underset{\mathbf{y}_i}{\operatorname{argmin}} \sum_{j=1}^K y_{ij} \|\mathbf{x}_i - \mathbf{c}_j\|_2^2 \quad (2)$$

$$v.d.k \ y_{ij} \in \{0; 1\} \ \forall i, j; \ \sum_{j=1}^K y_{ij} = 1 \ \forall i$$

Vì chỉ có một phần tử của vector nhãn  $y_i = 1$  nên ta có thể giản lược bài toán (2) thành:

$$j = \underset{j}{\operatorname{argmin}} \|\mathbf{x}_i - \mathbf{c}_j\|_2^2$$

$\Rightarrow$  Mỗi điểm  $\mathbf{x}_i$  cần thuộc vào cụm có điểm trung tâm gần nó nhất

**TH2:** Giả sử đã tìm được cụm cho từng điểm, ta cập nhật lại điểm trung tâm sao cho giá trị hàm mất mát nhỏ nhất

$$\mathbf{c}_j = \underset{\mathbf{c}_j}{\operatorname{argmin}} \sum_{i=1}^N y_{ij} \|\mathbf{x}_i - \mathbf{c}_j\|_2^2$$

Đặt  $l(\mathbf{c}_j)$  là hàm bên trong dấu  $\operatorname{argmin}$ , ta có đạo hàm:

$$\frac{\partial l(\mathbf{c}_j)}{\partial \mathbf{c}_j} = 2 \sum_{i=1}^N y_{ij} (\mathbf{c}_j - \mathbf{x}_i)$$

Giải phương trình đạo hàm bằng 0 để tìm nghiệm tối ưu

$$\begin{aligned} \mathbf{c}_j \sum_{i=1}^N y_{ij} &= \sum_{i=1}^N y_{ij} \mathbf{x}_i \\ \Rightarrow \mathbf{c}_j &= \frac{\sum_{i=1}^N y_{ij} \mathbf{x}_i}{\sum_{i=1}^N y_{ij}} \end{aligned}$$

*Nhận xét:*

- Do thuật toán K-Means sử dụng việc đo khoảng cách giữa các điểm dữ liệu để đánh giá tương quan giữa các điểm dữ liệu, chính vì lẽ đó ta nên chuẩn hóa dữ liệu trước khi tiến hành chạy thuật toán
- Thiết lập các điểm trung tâm khác nhau cho các cụm sẽ dẫn tới những kết quả phân cụm khác nhau.

#### 4.1.1.3 Thuật toán K-Means dạng cơ bản

MacQueen đề xuất thuật toán K-Means dựa trên ý tưởng phân chia từng điểm dữ liệu trong bộ dữ liệu  $\chi$  vào cụm mà có điểm trung tâm gần nó nhất.

Thuật toán cơ bản:

---

#### Algorithm 2 Thuật toán K-Means++

---

- 1: Chọn ngẫu nhiên các điểm tạo ra tập  $C$  gồm  $K$  điểm trung tâm  $C = \{c_1, c_2, \dots, c_K\}$ .
  - 2: Với mỗi  $i \in \{1, 2, \dots, K\}$ , ta tạo ra cụm  $C_i$  gồm những điểm trong  $\chi$  gần với điểm trung tâm  $c_i$  nhất (gần hơn so với các điểm trung tâm  $c_j$  còn lại với  $j \neq i$ ).
  - 3: Với mỗi  $i \in \{1, 2, \dots, K\}$ , đặt lại  $c_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$  là điểm trung tâm của cụm  $C_i$ .
  - 4: Lặp lại bước 2 và bước 3 đến khi không còn sự thay đổi trong tập các điểm trung tâm  $C$ .
- 

#### Ví dụ minh họa chạy thuật toán

Ta cần phân 4 vật thể  $A, B, C, D$  với 2 đặc tính được quan sát thành 2 cụm

Điểm dữ liệu	Các biến quan sát	
	$x_1$	$x_2$
A	5	3
B	-1	1
C	1	-2
D	-3	-2

Khởi tạo 1 cách phân cụm bất kỳ chẳng hạn (AB) và (CD), ta được

Cụm	Tọa độ điểm trung tâm	
	$\bar{x}_1$	$\bar{x}_2$
(AB)	$\frac{5 + (-1)}{2} = 2$	$\frac{3 + 1}{2} = 2$
(CD)	$\frac{1 + (-3)}{2} = -1$	$\frac{-2 + (-2)}{2} = -2$



Công thức cập nhật tọa độ điểm trung tâm của cụm trong trường hợp một vật thể có  $p$  yếu tố biến quan sát là

$$\bar{x}_{i,new} = \frac{n\bar{x}_i + x_{ji}}{n+1} \quad \text{nếu vật thể thứ } j \text{ được thêm vào cụm}$$

$$\bar{x}_{i,new} = \frac{n\bar{x}_i - x_{ji}}{n-1} \quad \text{nếu vật thể thứ } j \text{ được loại bỏ khỏi cụm}$$

Trong đó  $n$  là số lượng vật thể trong cụm trước khi cập nhật với điểm trung tâm là  $\bar{\mathbf{x}}' = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_p)$ .

Với cách khởi tạo các cụm ban đầu như vậy, cụm (AB) có điểm trung tâm là (2, 2) và cụm (CD) có điểm trung tâm là (-1, -2). Giả sử ta chuyển vật thể A có tọa độ là (5, 3) tới cụm (CD). Sau đó ta cập nhật lại tọa độ điểm trung tâm của các cụm:

$$\text{Cụm(B)} \quad \bar{x}_{1,new} = \frac{2 \cdot 2 - 5}{2 - 1} = -1 \quad \bar{x}_{2,new} = \frac{2 \cdot 2 - 3}{2 - 1} = 1$$

$$\text{Cụm (ACD)} \quad \bar{x}_{1,new} = \frac{2 \cdot (-1) + 5}{2 + 1} = 1 \quad \bar{x}_{2,new} = \frac{2 \cdot (-2) + 3}{2 + 1} = -0.33$$

Tính bình phương khoảng cách các điểm đến cụm (chính là đến điểm trung tâm của cụm):

$$d^2(A, (AB)) = (5 - 2)^2 + (3 - 2)^2 = 10 \quad \text{nếu A không chuyển}$$

$$d^2(A, (CD)) = (5 + 1)^2 + (3 + 2)^2 = 61 \quad d^2(A, (B)) = (5 + 1)^2 + (3 - 1)^2 = 40 \quad \text{nếu A chuyển vào cụm (CD)}$$

$$d^2(A, (ACD)) = (5 - 1)^2 + (3 + 0.33)^2 = 27.09$$

$\Rightarrow$  Khoảng cách của A tới cụm (AB) ngắn nhất nên ta không chuyển A đi

Tính toán tương tự ta có bảng khoảng cách từ các điểm đến các cụm sau khi cập nhật lần cuối

Cụm	Điểm dữ liệu			
	A	B	C	D
A	0	40	41	89
(BCD)	52	4	5	5

Khoảng cách trong 1 cụm là:

Cụm (A): 0

Cụm (BCD):  $4 + 5 + 5 = 14 \Rightarrow$  Tổng khoảng cách trong 1 cụm với cách chia này là 14.

Tương tự đối với các cách chia cụm khác như  $\{(C), (ABD)\}, \{(AD), (BC)\}, \dots$  ta đều xác định được tổng khoảng cách trong 1 cụm nhưng cách phân cụm thành  $\{(A), (BCD)\}$  là cách phân cụm cho tổng khoảng cách trong 1 cụm nhỏ nhất.

#### 4.1.1.4 Đánh giá mô hình

Trong thực tế với mô hình học không giám sát hay thuật toán phân loại, không có cách đánh giá chính xác cho đầu ra của bài toán. Hơn nữa, việc thuật toán K-Means chọn một K cố định không phụ thuộc vào bộ dữ liệu nên không có tiêu chuẩn chung cho đầu ra. Một số cách để đánh giá mô hình phân cụm:

- Chỉ số Dunn
- Đánh giá Silhouette

*Cách thứ nhất: Chỉ số Dunn*

Ta định nghĩa chỉ số Dunn như sau:

$$\text{Chỉ số Dunn} = \min(\text{Khoảng cách giữa 2 cụm}) / \max(\text{Khoảng cách trong 1 cụm})$$

Khoảng cách trong một cụm được tính bằng tổng khoảng cách từ mọi điểm trong cụm đến điểm trung tâm của cụm. Khi khoảng cách trong một cụm nhỏ thì các điểm dữ liệu trong cụm sẽ có các đặc tính đang quan sát khá tương đồng nhau. Do đó, khoảng cách trong 1 cụm càng nhỏ thì ta càng có cơ sở để đánh giá cách phân cụm đó tốt (gồm những điểm có đặc điểm tương tự nhau).

Khoảng cách giữa 2 cụm được tính bằng khoảng cách giữa các điểm trung tâm của 2 cụm với nhau. Khi khoảng cách giữa 2 cụm lớn thì các đặc điểm đang quan sát của các điểm dữ liệu thuộc 2 cụm sẽ phân biệt với nhau. Do đó, cần tìm cách phân cụm làm cho khoảng cách giữa 2 cụm lớn.

$\Rightarrow$  Cách phân tập dữ liệu thành  $K$  cụm được đánh giá là hiệu quả khi chỉ số Dunn lớn.

*Cách thứ hai: Phân tích Silhouette*

Phân tích Silhouette dùng để đo độ phân biệt giữa các cụm, tính chỉ số Silhouette cho từng điểm trong cụm sau đó lấy trung bình lại cho ta giá trị Silhouette trung bình để đánh giá độ phân biệt giữa các cụm. Ta xét:

$$I(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

Với  $a(i)$  là trung bình của khoảng cách từ điểm  $i$  đến tất cả các điểm khác trong một cụm.

$b(i)$  là trung bình của khoảng cách từ điểm  $i$  đến tất cả các điểm trong một cụm gần nhất với cụm đang xét.

$I(i)$  nhận giá trị trong khoảng  $[-1; 1]$  nên lấy trung bình toàn bộ chỉ số Silhouette cho các điểm trong bộ dữ liệu ta được  $I$  là chỉ số Silhouette trung bình.

- Nếu  $I = 1$  thì 2 cụm đang xét cách xa nhau.
- Nếu  $I = 0$  thì 2 cụm đang xét đang rất gần nhau.
- Nếu  $I = -1$  thì các điểm đã bị gán sai nhãn.

Sau khi xác định được chỉ số Silhouette trung bình ta có thể đánh giá được cách phân cụm đó đã đủ tốt hay chưa.

#### 4.1.1.5 Cải tiến thuật toán

Cách chọn số K cụm và cách thiết lập các điểm trung tâm ảnh hưởng rất nhiều đến kết quả của thuật toán. Nên ta thường sử dụng 2 phương pháp sau khi chạy thuật toán K-Means

- Thuật toán khuỷu tay  $\Rightarrow$  Chọn số K cụm thích hợp cho một bộ dữ liệu.
- Thuật toán K-Means++ để thiết lập các điểm trung tâm cho các cụm.

Ngoài ra, do thuật toán K-Means nhạy cảm với điểm ngoại lai (outliers) nên người ta thường dùng các phương pháp để loại bỏ điểm ngoại lai trước khi tiến hành phân cụm.

- Z-score
- Box plot + scatter plot
- IQR score

*Cách thứ nhất: Z-score*

$$Zscore = \frac{x - \mu}{\sigma}$$

Z-score dùng để đo độ lệch giữa các điểm dữ liệu và giá trị trung bình. Ta đặt mức giới hạn cho Z-score để xác định outliers.

Nếu  $|Zscore| > 3$  thì điểm  $x$  ứng với giá trị đó là điểm dữ liệu ngoại lai.

*Cách thứ hai: Khoảng cách tứ phân vị IQR*

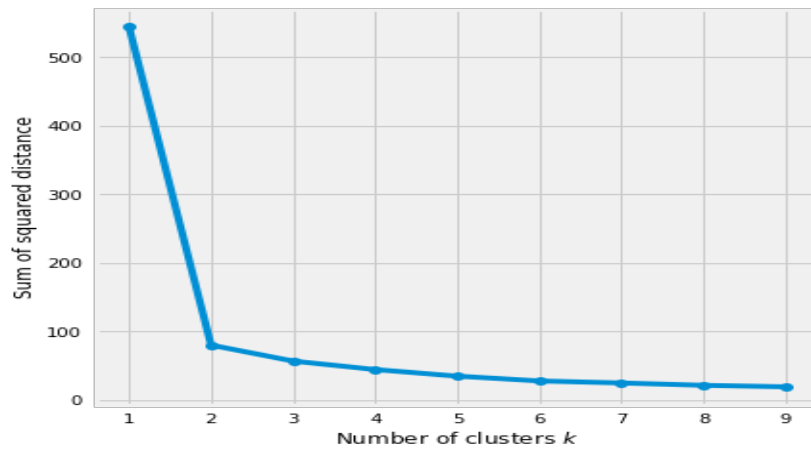
- Tứ phân vị thứ nhất  $Q1$ : Mức phân vị 25%
- Tứ phân vị thứ ba  $Q3$ : Mức phân vị 75%
- Khoảng cách tứ phân vị:  $IQR = Q3 - Q1$

Quy tắc  $1.5IQR$  để xác định giá trị ngoại lai: Giá trị  $x$  được gọi là điểm ngoại lai nếu  $x \notin [Q1 - 1.5IQR, Q3 + 1.5 * IQR]$

#### 4.1.1.6 Khởi tạo số K cụm thích hợp

Ta định nghĩa chỉ số SSE: Tổng khoảng cách trong một cụm của toàn bộ các cụm của bộ dữ liệu.

Phương pháp khuỷu tay sẽ tiến hành chạy thuật toán K-Means lần lượt các chỉ số  $K = 1, 2, 3, \dots$  đến một K đủ lớn (thường là 9). Sau đó tính toán SSE với từng cách chia K cụm và tiến hành vẽ đồ thị tương quan giữa SSE và số K cụm. Chọn số cụm tương ứng với điểm khuỷa tay trên đồ thị đây (điểm mà đồ thị gấp khúc nhiều nhất) sẽ cho ta cách chia cụm hợp lý theo phương pháp khuỷu tay.



Hình 15: Minh họa phương pháp khuỷu tay

#### 4.1.1.7 Thuật toán cải tiến K-Means++

Thuật toán này thường được chạy để tạo ra bộ điểm trung tâm trước khi chạy thuật toán K-Means. Các bước của thuật toán:

---

**Algorithm 3** Thuật toán K-Means++

---

- 1: Chọn ngẫu nhiên một điểm trung tâm đầu tiên từ bộ dữ liệu
  - 2: Tính khoảng cách  $D(x)$  với mỗi điểm dữ liệu  $x$  đến điểm trung tâm gần nó nhất
  - 3: Chọn điểm trung tâm mới với xác suất  $\frac{D(x)^2}{\sum_{x \in \mathcal{X}} D(x)^2}$  lớn nhất
  - 4: Lặp lại bước 2 và bước 3 đến khi chọn được  $K$  điểm dữ liệu
  - 5: Chạy thuật toán K-Means với  $K$  điểm trung tâm được khởi tạo
- 

### 4.1.2 Phân tích độ hiệu quả của thuật toán cải tiến K-Means++

Phần chứng minh sau được trích dẫn từ bài báo rất nổi tiếng về thuật toán K-means của tác giả David Arthur[1]

#### 4.1.2.1 Định nghĩa

Trong thuật toán K-Means, chúng ta có  $k$  cụm và tập  $\mathcal{X}$  gồm  $n$  điểm dữ liệu, chúng ta cần tìm ra tập  $C$  gồm  $k$  điểm trung tâm để làm cực tiểu hóa hàm mục tiêu:

$$\phi = \sum_{x \in \mathcal{X}} \min_{c \in C} \|x - c\|^2$$

Chúng ta ký hiệu  $C_{OPT}$  là tập các điểm trung tâm cho cách phân cụm tối ưu và  $\phi_{OPT}$  là hàm mục tiêu tương ứng.

Cho 1 cách phân cụm biểu diễn bởi tập các điểm trung tâm  $C$  với hàm mục tiêu  $\phi$  tương ứng, chúng ta ký hiệu  $\phi(A)$  là hàm mục tiêu ứng với một tập  $A \subset \mathcal{X}$

$$\phi(A) = \sum_{x \in A} \min_{c \in C} \|x - c\|^2$$

**Mệnh đề 1.1** Gọi  $S$  là một cụm tùy ý có điểm trung tâm là  $c(S)$ ,  $z$  là một điểm tùy ý. Khi đó,  $\sum_{x \in S} \|x - z\|^2 - \sum_{x \in S} \|x - c(S)\|^2 = |S| \cdot \|c(S) - z\|^2$

*Chứng minh*

Xét trong không gian 2 chiều,  $z = (z_x, z_y)$ ,  $S = \{p_1, p_2, \dots, p_n\}$  với  $p_i = (x_i, y_i)$

1.  $c(S) = \left(\frac{\sum_i x_i}{n}, \frac{\sum_i y_i}{n}\right)$
2.  $\sum_{x \in S} \|x - z\|^2 = \sum_{p=(x_i, y_i) \in S} ((x_i - z_x)^2 + (y_i - z_y)^2)$
3.  $\sum_{x \in S} \|x - c(S)\|^2 = \sum_i (x_i - \frac{\sum_i x_i}{n})^2 + \sum_i (y_i - \frac{\sum_i y_i}{n})^2$
4.  $\begin{aligned} \sum_{x \in S} \|x - z\|^2 - \sum_{x \in S} \|x - c(S)\|^2 &= nz_x^2 + nz_y^2 - 2z_x \cdot \sum x_i - 2z_y \cdot \sum y_i + n\left(\frac{\sum_i x_i}{n}\right)^2 + n\left(\frac{\sum_i y_i}{n}\right)^2 \\ &= n\left[z_x^2 + z_y^2 - 2z_x \cdot \frac{\sum x_i}{n} - 2z_y \cdot \frac{\sum y_i}{n} + \left(\frac{\sum_i x_i}{n}\right)^2 + \left(\frac{\sum_i y_i}{n}\right)^2\right] = |S| \cdot \|c(S) - z\|^2 \end{aligned}$

Tổng quát hóa cho trường hợp số chiều lớn hơn 2, ta được điều phải chứng minh.

#### 4.1.2.2 K-Means++ có độ hiệu quả $O(\log k)$

**Mệnh đề 2.1** Nếu  $A$  là một cụm ngẫu nhiên trong  $C_{OPT}$ ,  $C$  là cách phân cụm sao cho chỉ có một điểm trung tâm được chọn ra một cách ngẫu nhiên từ  $A$ . Khi đó,  $E[\phi(A)] = 2 \cdot \phi_{OPT}(A)$   
*Chứng minh:* Ký hiệu  $c(A)$  là điểm trung tâm của cụm  $A$ . Bằng mệnh đề 1.1, ta có:

$$E[\phi(A)] = \frac{1}{|A|} \sum_{a_0 \in A} \sum_{a \in A} \|a - a_0\|^2$$

$$\begin{aligned}
 &= \frac{1}{|A|} \sum_{a_0 \in A} \left( \sum_{a \in A} \|a - c(A)\|^2 + |A| \cdot \|a_0 - c(A)\|^2 \right) \\
 &= 2 \sum_{a \in A} \|a - c(A)\|^2.
 \end{aligned}$$

**Mệnh đề 2.2** Nếu  $A$  là một cụm ngẫu nhiên trong  $C_{OPT}$  và  $C$  là một cách phân cụm ngẫu nhiên. Nếu chúng ta thay vào  $C$  ngẫu nhiên một điểm trung tâm chọn từ  $A$  với xác suất  $D^2$  thì  $E[\phi(A)] \leq 8\phi_{OPT}(A)$

*Chứng minh* Xác suất để chúng ta chọn một điểm  $a_0$  từ  $A$  làm điểm trung tâm cho  $C$  là  $\frac{D(a_0)^2}{\sum_{a \in A} D(a)^2}$ . Sau khi chọn điểm trung tâm  $a_0$  từ  $A$  thì một điểm  $a$  thuộc  $A$  sẽ cộng thêm  $\min(D(a), \|a - a_0\|)^2$  vào hàm mục tiêu  $\phi(A)$

$$\implies E[\phi(A)] = \sum_{a_0 \in A} \frac{D(a_0)^2}{\sum_{a \in A} D(a)^2} \sum_{a \in A} \min(D(a), \|a - a_0\|)^2$$

Theo bất đẳng thức tam giác ta có  $D(a_0) \leq D(a) + \|a - a_0\| \forall a, a_0$ . Theo bất đẳng thức Cauchy-Schwarz, ta có  $D(a_0)^2 \leq 2D(a)^2 + 2\|a - a_0\|^2$ . Tính tổng cho toàn bộ  $a \in A$  dẫn đến  $D(a_0)^2 \leq \frac{2}{|A|} \sum_{a \in A} D(a)^2 + \frac{2}{|A|} \sum_{a \in A} \|a - a_0\|^2$

$$\begin{aligned}
 \implies E[\phi(A)] &\leq \frac{2}{|A|} \cdot \sum_{a_0 \in A} \frac{\sum_{a \in A} D(a)^2}{\sum_{a \in A} D(a)^2} \cdot \sum_{a \in A} \min(D(a), \|a - a_0\|)^2 \\
 &\quad + \frac{2}{|A|} \cdot \sum_{a_0 \in A} \frac{\sum_{a \in A} \|a - a_0\|^2}{\sum_{a \in A} D(a)^2} \cdot \sum_{a \in A} \min(D(a), \|a - a_0\|)^2
 \end{aligned}$$

Ta lần lượt thay  $\min(D(a), \|a - a_0\|)^2 \leq \|a - a_0\|^2$  và  $\min(D(a), \|a - a_0\|)^2 \leq D(a)^2$  vào bất đẳng thức trên

$$\implies E[\phi(A)] \leq \frac{4}{|A|} \cdot \sum_{a_0 \in A} \sum_{a \in A} \|a - a_0\|^2 = 8\phi_{OPT}(A)$$

(Đẳng thức cuối suy ra từ mệnh đề 2.1)

Chúng ta đã chứng minh xong tính bị chặn của hàm mục tiêu nếu như chọn lần lượt các điểm trung tâm từ các cụm của  $C_{OPT}$ . Ta sẽ chứng minh trong trường hợp tổng quát để đưa ra độ hiệu quả  $O(\log k)$ .

**Mệnh đề 2.3:** Với  $C$  là một cách phân cụm bất kỳ. Chọn  $u > 0$  cụm từ  $C_{OPT}$  và ký hiệu  $\chi_u$  cho tập các điểm trong các cụm ấy. Đặt  $\chi_c = \chi - \chi_u$ . Giả sử ta thêm ngẫu nhiên  $t \leq u$  điểm trung tâm cho  $C$  với xác suất được chọn là  $D^2$ . Ký hiệu  $C'$  là cách chia cụm mới sau khi đã cập nhật  $t$  điểm trung tâm mới và  $\phi'$  là hàm mục tiêu tương ứng. Khi đó

$$E[\phi'] \leq (\phi(\chi_c) + 8\phi_{OPT}(\chi_u)) \cdot (1 + H_t) + \frac{u-t}{u} \cdot \phi(\chi_u).$$

(Với  $H_t = 1 + \frac{1}{2} + \dots + \frac{1}{t}$  là tổng harmonic)

*Chứng minh*

Sử dụng quy nạp với giả thiết bất đẳng thức đúng cho  $(t-1, u)$  và  $(t-1, u-1)$  từ đó chứng minh bất đẳng thức đúng cho  $(t, u)$ .

*Trường hợp cơ sở:* Với  $t = 0, u > 0$ , suy ra  $1 + H_t = \frac{u-t}{u} = 1$ .

Do  $\phi' \leq \phi$  nên  $E[\phi'] \leq \phi + 8\phi_{OPT}(\chi_u) \implies$  bất đẳng thức đúng

*Trường hợp cơ sở:* Với  $t = u = 1$ , ta chọn một điểm trung tâm mới từ  $\chi_u$  với xác suất  $\frac{\phi(\chi_u)}{\phi}$ . Sử dụng mệnh đề 2.2, ta có  $E[\phi'] \leq 8\phi_{OPT}(\chi_u) + \phi(\chi_c)$

$$\begin{aligned} \implies E[\phi'] &\leq \frac{\phi(\chi_u)}{\phi} \cdot (\phi(\chi_c) + 8\phi_{OPT}(\chi_u)) + \frac{\phi(\chi_c)}{\phi} \cdot \phi \\ &\leq 2\phi(\chi_c) + 8\phi_{OPT}(\chi_u) \end{aligned}$$

Vì  $1 + H_t = 2$  nên bất đẳng thức đúng.

*Bước quy nạp:* Giả sử bất đẳng thức đúng cho  $(t-1, u)$  và  $(t-1, u-1)$  ta cần chứng minh bất đẳng thức đúng cho trường hợp  $(t, u)$ .

Xét 2 trường hợp, giả sử điểm trung tâm đầu tiên ta chuyển vào  $C$  được lấy ra từ  $\chi_c$ , khi đó theo giả thiết quy nạp ta có:

$$E[\phi'] \leq \frac{\phi(\chi_c)}{\phi} \left( (\phi(\chi_c) + 8\phi_{OPT}(\chi_u)) \cdot (1 + H_{t-1}) + \frac{u-t+1}{u} \cdot \phi(\chi_u) \right). \quad (1)$$

Giả sử điểm trung tâm đầu tiên được chọn ra từ cụm  $A$  nào đó trong  $u$  cụm nêu trên. Xác suất chọn một điểm trung tâm như vậy là  $\frac{\phi(A)}{\phi}$ . Ký hiệu  $p_a$  là xác suất để chọn một điểm  $a \in A$  làm điểm trung tâm, đặt lại điểm trung tâm đó cho  $A$ , ký hiệu  $\phi_A$  cho  $\phi(A)$  sau khi đặt  $a$  làm điểm trung tâm. Chuyển  $A$  sang  $\chi_c$  và áp dụng giả thiết quy nạp cho trường hợp  $(t-1, u-1)$ , ta được giá trị lớn nhất của  $E[\phi_{OPT}]$  là

$$\frac{\phi(A)}{\phi} \cdot \sum_{a \in A} p_a \cdot \left( (\phi(\chi_c) + \phi_a + 8\phi_{OPT}(\chi_u) - 8\phi_{OPT}(A)) \cdot (1 + H_{t-1}) + \frac{u-t}{u-1} \cdot (\phi(\chi_u) - \phi(A)) \right)$$

Theo mệnh đề 2.2  $\sum_{a \in A} p_a \phi_a \leq 8\phi_{OPT}(A)$

$$\implies E[\phi_{OPT}] \leq \frac{\phi(A)}{\phi} \left( (\phi(\chi_c) + 8\phi_{OPT}(\chi_u)) \cdot (1 + H_{t-1}) + \frac{u-t}{u-1} \cdot (\phi(\chi_u) - \phi(A)) \right)$$

Theo bất đẳng thức Cauchy-Schwarz,  $\sum_{A \subset \chi_u} \phi(A)^2 \geq \frac{1}{u} \cdot \phi(\chi_u)^2$ . Do đó, nếu ta cộng toàn bộ các cụm  $A$ , ta có:

$$\begin{aligned} E[\phi'] &\leq \frac{\phi(\chi_u)}{\phi} \cdot \left( (\phi(\chi_c) + 8\phi_{OPT}(\chi_u)) \cdot (1 + H_{t-1}) + \frac{1}{\phi} \cdot \frac{u-t}{u-1} \left( \phi(\chi_u)^2 - \frac{1}{u} \cdot \phi(\chi_u)^2 \right) \right) \\ \implies E[\phi'] &\leq \frac{\phi(\chi_u)}{\phi} \cdot \left( (\phi(\chi_c) + 8\phi_{OPT}(\chi_u)) \cdot (1 + H_{t-1}) + \frac{u-t}{u} \cdot \phi(\chi_u) \right) \end{aligned} \quad (2)$$

Kết hợp (1) và (2), ta được:

$$\begin{aligned} E[\phi'] &\leq \left( \phi(\chi_c) + 8\phi_{OPT}(\chi_u) \right) \cdot (1 + H_{t-1}) + \frac{u-t}{u} \cdot \phi(\chi_u) + \frac{\phi(\chi_c)}{\phi} \cdot \frac{\phi(\chi_u)}{u} \\ \implies E[\phi'] &\leq \left( \phi(\chi_c) + 8\phi_{OPT}(\chi_u) \right) \cdot \left( 1 + H_{t-1} + \frac{1}{u} \right) + \frac{u-t}{u} \cdot \phi(\chi_u) \end{aligned}$$

Kết hợp với  $\frac{1}{u} \leq \frac{1}{t}$ , ta suy ra bất đẳng thức được chứng minh.

Cuối cùng, ta thiết lập cận trên cho  $E[\phi]$  một cách tổng quát hơn. Trong thuật toán K-Means++ sau khi đã tiến hành chọn ngẫu nhiên 1 điểm làm điểm trung tâm (xong bước 1). Ký hiệu  $A$  là cụm chứa điểm trung tâm đầu tiên đó trong các cụm của  $C_{OPT}$ . Áp dụng mệnh đề 2.3 cho  $t = u = k - 1$ , ta được:

$$E[\phi'] \leq \left( \phi(A) + 8\phi_{OPT} - 8\phi_{OPT}(A) \right) \cdot (1 + H_{k-1})$$

Kết hợp với  $1 + H_{k-1} \leq 1 + \log k$  và mệnh đề 2.2 ta có:

$$E[\phi] \leq 8(\log k + 2)\phi_{OPT}$$

Vậy ta đã thiết lập được cận trên cho giá trị kỳ vọng của độ phức tạp thuật toán K-Means với việc khởi tạo các điểm trung tâm ban đầu theo thuật toán K-Means++. Cận trên  $8(\log k + 2)\phi_{OPT}$  giúp ta khẳng định rằng độ hiệu quả của thuật toán K-Means++ là  $O(\log k)$ .



## 4.2 Kết quả thực nghiệm

### 4.2.1 Loại bỏ dữ liệu ngoại lai

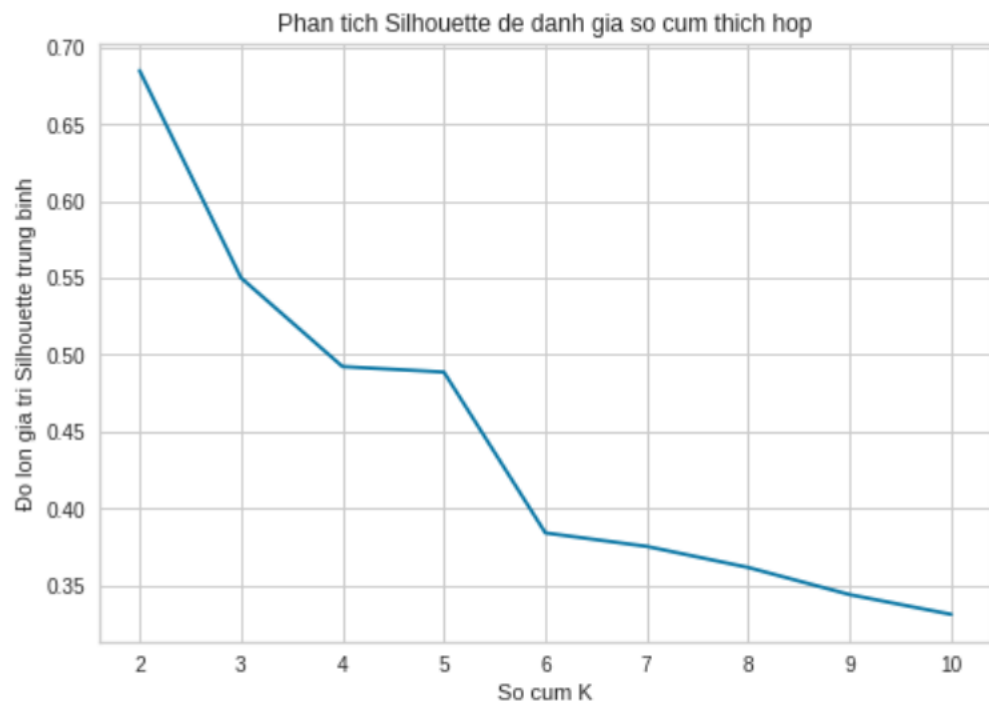
Với việc sử dụng bộ dữ liệu Iris, ta tiến hành sử dụng quy tắc  $1.5IQR$  để loại bỏ các điểm outliers. Các điểm bị loại bỏ là những bông hoa có các chiều dài, chiều rộng của cánh hoa hoặc nhụy hoa dài hoặc ngắn bất thường so với các điểm khác trong bộ dữ liệu. Sau khi tiến hành loại bỏ, bộ dữ liệu còn 142 điểm.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 142 entries, 0 to 149
Data columns (total 5 columns):
#   Column          Non-Null Count  Dtype
---  -
0   sepal_length    142 non-null   float64
1   sepal_width     142 non-null   float64
2   petal_length    142 non-null   float64
3   petal_width     142 non-null   float64
4   species         142 non-null   object
dtypes: float64(4), object(1)
memory usage: 11.7+ KB
```

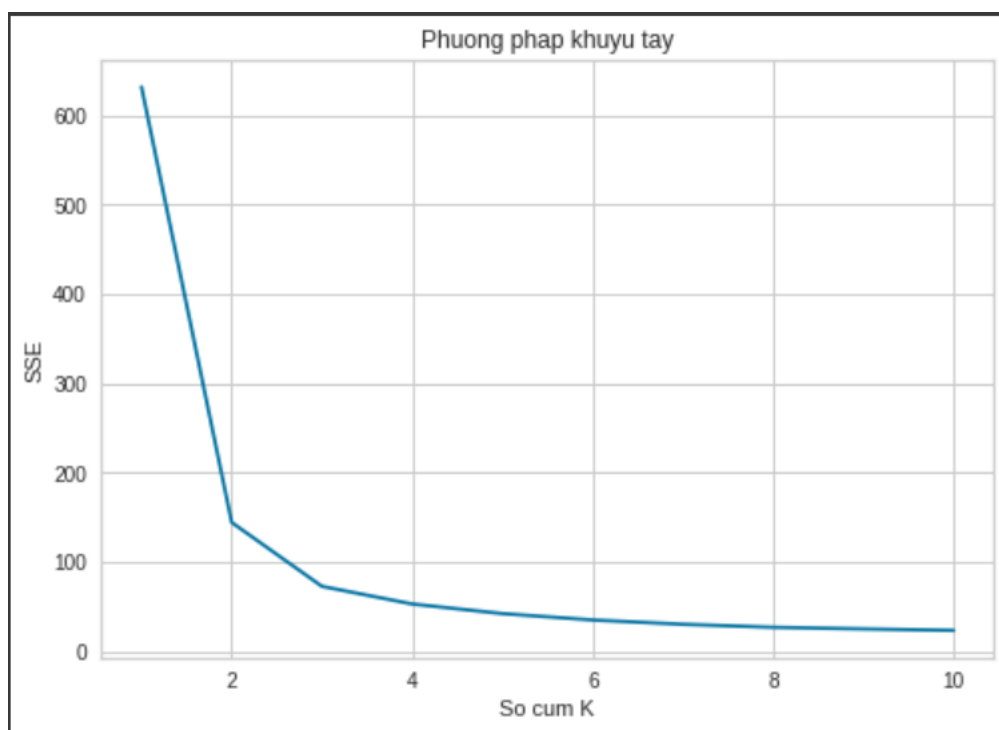
Hình 16: Bộ dữ liệu sau khi loại bỏ dữ liệu ngoại lai

### 4.2.2 Phân tích Silhoutte và Phương pháp khuỷu tay

Sau khi tiến hành đo chỉ số Silhoutte cho bộ dữ liệu sau khi loại bỏ outliers đồng thời chạy phương pháp khuỷu tay, ta chọn ra số cụm K phù hợp với bộ dữ liệu là 3 cụm.



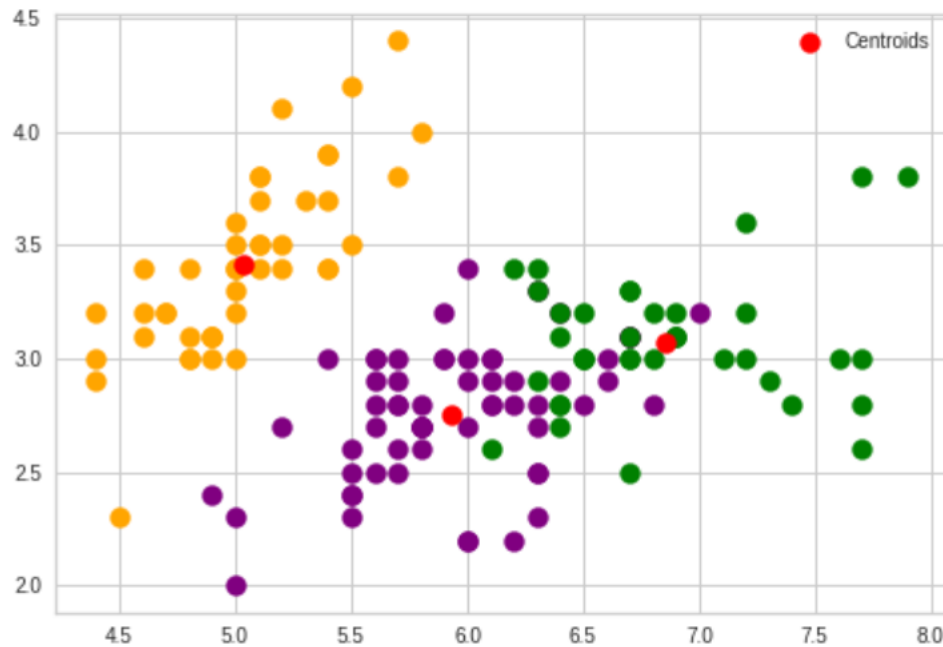
Hình 17: Kết quả chạy phân tích Silhouette



Hình 18: Kết quả chạy phương pháp khuỷu tay

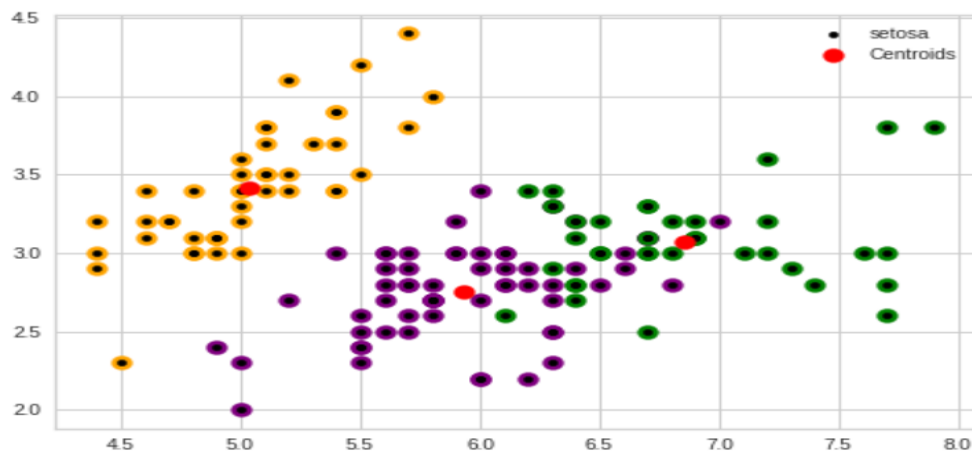
### 4.2.3 Kết quả cuối cùng

Sau khi tiến hành chạy thuật toán K-Means với cách khởi tạo điểm theo K-Means++ và chạy vòng lặp 600 lần thì kết quả phân cụm là:



Hình 19: Kết quả chạy thuật toán K-Means

Nếu như ta kiểm tra xem với nhãn đúng của các điểm dữ liệu trong 1 cụm (xem các điểm đó thuộc chủng hoa nào) và tiến hành đánh dấu các điểm đó trong kết quả phân cụm thì ta được hình ảnh.



Hình 20: So sánh với nhãn đúng

$\Rightarrow$  Từ đây ta có thể đưa ra nhận xét rằng các điểm dữ liệu trong cùng 1 chủng hoa phần lớn sẽ thuộc cùng 1 cụm, do đó ta có thể kết luận rằng các bông hoa trong cùng 1 chủng thì sẽ có các đặc tính về độ dài, độ rộng cánh hoa và nhụy hoa tương đồng nhau.

## 5 Thuật toán phân cụm theo mô hình xác suất thống kê

### 5.1 Phương pháp luận

#### 5.1.1 Vấn đề đặt ra

Về mặt trực quan, các phương pháp được thảo luận ở trên, bao gồm các phương pháp như phương pháp kết nối đơn, kết nối toàn phần, kết nối trung bình, phương pháp Ward hay là phương pháp K-means, cho ta một cách phân cụm hợp lý dựa vào các quan sát trong bộ dữ liệu. Tuy vậy, các cách phân cụm đó không cho ta biết được các quan sát đó được hình thành như thế nào, hay cụ thể hơn là ta không biết được phân phối của bộ dữ liệu. Các nghiên cứu đã chỉ ra rằng, hầu hết các phương pháp phân cụm nâng cao đều được xây dựng, phân tích và đánh giá dựa trên một mô hình xác suất thống kê toán học nào đó. Do đó, trong phần này, chúng ta sẽ xem xét một mô hình xác suất thống kê hay được sử dụng để phân tích phân phối của bộ dữ liệu, đồng thời chỉ ra phương pháp phân cụm đối với mô hình toán học này.

#### 5.1.2 Mô hình Gaussian hỗn hợp

Như đã nói ở trên, chúng ta sẽ xem xét một mô hình toán học biểu diễn phân phối của bộ dữ liệu. Ta sẽ xem xét mô hình toán học sau. Xét  $X$  là biến ngẫu nhiên  $D$  chiều biểu diễn dữ liệu. Biến ngẫu nhiên  $X$  có hàm mật độ là:

$$f_{Mix} = \sum_{k=1}^K p_k f(x; \mu_k; \Sigma_k) \quad (12)$$

trong đó:

- $p_k \in [0, 1]$  là xác suất của một điểm dữ liệu thuộc vào phân cụm  $k$ .
- $f(x; \mu_k; \Sigma_k)$  là hàm mật độ xác suất của biến ngẫu nhiên biểu diễn dữ liệu thuộc phân cụm  $k$ .

Ta có thể nói phân phối  $f_{Mix}$  là hỗn hợp của  $K$  phân phối  $f(x; \mu_1; \Sigma_1), \dots, f(x; \mu_k; \Sigma_k)$  bởi các quan sát của biến ngẫu nhiên  $X$  đều được tạo bởi các phân phối thành phần  $f(x; \mu_k; \Sigma_k)$  với xác suất  $p_k$ .

Ở trong công thức 12, ta sẽ phải xác định các phân phối thành phần  $f(x; \mu_k; \Sigma_k)$ . Sẽ không có một quy chuẩn chung nào để xác định các phân phối thành phần và thông thường ta sẽ xác

định các phân phối thành phần này dựa vào đặc điểm của bộ dữ liệu. Một trong những mô hình hay được sử dụng cho các phân phối thành phần là phân phối chuẩn nhiều chiều  $\mathcal{N}(\mu, \Sigma)$ , với  $\mu$  và  $\Sigma$  lần lượt là kì vọng và ma trận hiệp phương sai. Trong bài báo cáo này, chúng em sẽ xét các phân phối thành phần là các phân phối chuẩn nhiều chiều. Cụ thể,  $f(x, \mu_k, \Sigma_k) \sim \mathcal{N}(\mu_k, \Sigma_k)$ . Khi đó hàm mật độ  $f_{Mix}(x)$  sẽ có dạng:

$$\begin{aligned} f_{Mix} &= \sum_{k=1}^K p_k f(x; \mu_k; \Sigma_k) \\ &= \sum_{k=1}^K p_k \frac{1}{(2\pi)^{D/2} |\Sigma_k|^{1/2}} \exp\left(-\frac{1}{2} (x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k)\right) \end{aligned}$$

Quay trở lại bài toán phân cụm, bây giờ ta có  $N$  điểm dữ liệu  $\{x_j\}_{j=1}^N$  và nhiệm vụ của ta bây giờ là phải phân cụm các điểm dữ liệu đó. Vậy áp dụng mô hình toán học ở trên, với số phân cụm  $K$  cố định, ta sẽ đi tìm các hệ số  $\{p_i\}_{i=1}^K, \{\mu_i\}_{i=1}^K, \{\Sigma_i\}_{i=1}^K$  để cực đại hóa hàm hợp lí sau:

$$L\left(\{p_i\}_{i=1}^K, \{\mu_i\}_{i=1}^K, \{\Sigma_i\}_{i=1}^K\right) = \prod_{j=1}^N f_{Mix}\left(x_j \mid \{p_i\}_{i=1}^K, \{\mu_i\}_{i=1}^K, \{\Sigma_i\}_{i=1}^K\right) = \prod_{j=1}^N \sum_{k=1}^K p_k f(x_j; \mu_k; \Sigma_k)$$

Để đơn giản về mặt kí hiệu, ta sẽ đặt  $\theta = \left(\{p_i\}_{i=1}^K, \{\mu_i\}_{i=1}^K, \{\Sigma_i\}_{i=1}^K\right)$

Vậy khi đó ta sẽ phát biểu bài toán như sau. Tìm  $\theta$  để cực đại hóa hàm:

$$L(\theta) = \prod_{j=1}^N \sum_{k=1}^K p_k f(x_j; \mu_k; \Sigma_k) \quad (\text{P})$$

với điều kiện

$$\begin{cases} \sum_{i=1}^K p_i = 1 \\ p_i \geq 0 \quad \forall i = \{1, 2, \dots, K\} \end{cases}$$

Ta có vài nhận xét sau:

- Như vậy, đối với một tập dữ liệu và với số phân cụm cố định, thay vì ta lựa chọn các phương pháp phù hợp (kết nối đơn, kết nối trung bình, k-means,...) để phân cụm dữ liệu, ta sẽ chuyển sang lựa chọn mô hình. Tức là bây giờ ta sẽ quan tâm đến việc lựa chọn các phân phối thành phần sao cho nó hợp lí với bộ dữ liệu của bài toán.

- Mô hình toán học xác suất thống kê như trên có sự liên kết rất chặt chẽ với các phương pháp phân cụm như k-means. Cụ thể, các nhà khoa học [6] đã chứng minh được rằng nếu các phân phối thành phần là các phân phối chuẩn nhiều chiều và có các ma trận hiệp phương sai  $\Sigma_i = \eta \mathbb{I}$  trong đó  $\mathbb{I}$  là ma trận đơn vị, thì kết quả phân cụm của phương pháp xác suất thống kê sẽ tương đương với kết quả khi ta thực hiện phương pháp k-means.
- Tuy vậy, hiện nay vẫn chưa tìm ra được mô hình xác suất thống kê nào cho ra kết quả tương tự với các thuật toán như kết nối đơn, kết nối toàn phần hay kết nối trung bình. Cái khó nằm ở chỗ ta phải tìm ra được các phân phối thành phần phù hợp với các phương pháp phân cụm kể trên.

Bài toán (P) là bài toán tối ưu phi tuyến có ràng buộc và là một bài toán khó có lời giải chính xác. Do đó, ta sẽ sử dụng thuật toán cực đại hóa kì vọng (Expectation Maximization) để giải bài toán (P). Ở phần tiếp theo ta sẽ nói cụ thể hơn về thuật toán cực đại hóa kì vọng này.

### 5.1.3 Thuật toán cực đại hóa kì vọng

Thuật toán cực đại hóa kì vọng (Expectation Maximization) là một kỹ thuật được dùng rộng rãi trong thống kê và học máy để giải bài toán tìm hợp lý cực đại hoặc hậu nghiệm cực đại (MAP) của một mô hình xác suất có các biến ẩn. Thuật toán cực đại hóa có tên gọi thân thuộc là thuật toán EM. Sở dĩ được gọi như vậy một phần là do thuật toán này bao gồm việc thực hiện liên tiếp tại mỗi vòng lặp 2 quá trình (E): tính kì vọng của hàm hợp lý của giá trị các ẩn biến dựa theo ước lượng đang có về các tham số của mô hình và (M): ước lượng tham số của mô hình để cực đại hóa giá trị của hàm tính được ở (E). Các giá trị tìm được ở (E) và (M) tại mỗi vòng lặp sẽ được dùng cho việc tính toán ở vòng lặp kế tiếp.

Áp dụng vào việc giải bài toán (P), thuật toán cực đại hóa kì vọng sẽ có đầu vào và đầu ra lần lượt là:

- **Đầu vào:** Số phân cụm  $K$ ,  $N$  điểm dữ liệu, phân phối tiên nghiệm  $f_{Mix}(x)$ .
- **Đầu ra:** Bộ hệ số  $\hat{\theta} = \left( \{\hat{p}_i\}_{i=1}^K, \{\hat{\mu}_i\}_{i=1}^K, \{\hat{\Sigma}_i\}_{i=1}^K \right)$  tối ưu

Ngoài tham số  $\theta$  cần tìm, ta sẽ có các kí hiệu sau phục vụ cho thuật toán cực đại hóa kì vọng:

- Biến ngẫu nhiên  $X$  tuân theo phân phối có hàm mật độ là  $f_{Mix}(x)$ .
- Ta xét  $\mathbf{x} = (x_1, x_2, \dots, x_N)$  là một vector chứa  $N$  quan sát độc lập của biến ngẫu nhiên  $X$ .

- Ta xét một biến ngẫu nhiên ẩn  $Z$  thỏa mãn các tính chất sau:

$$\begin{aligned} & - X|(Z=i) \sim \mathcal{N}(\mu_i, \Sigma_i) \quad \forall i = \{1, 2, \dots, K\} \\ & - P(Z=i) = p_i \quad \forall i = \{1, 2, \dots, K\} \end{aligned}$$

- Cùng với đó ta xét  $\mathbf{z} = (z_1, z_2, \dots, z_N)$  là một vector chứa  $N$  quan sát độc lập của biến ngẫu nhiên  $Z$ . Ý nghĩa của vector quan sát  $\mathbf{z}$  là nếu  $z_j = m$  thì quan sát  $x_j$  sẽ thuộc phân cụm thứ  $m$ .

Với kí hiệu như trên, ta sẽ định nghĩa **hàm hợp lí không đầy đủ**:

$$L(\theta, X) = P(X|\theta) = f_{Mix}(X|\theta)$$

Khi đó, nếu  $\mathbf{x}$  là vector quan sát của biến ngẫu nhiên  $X$ , hàm hợp lí không đầy đủ của ta sẽ có dạng:

$$L(\theta, \mathbf{x}) = P(\mathbf{x}|\theta) = \prod_{i=1}^N f_{Mix}(x_i|\theta) = \prod_{i=1}^N \sum_{j=1}^K p_j f(x_i, \mu_j, \Sigma_j)$$

Vậy đối với mô hình toán học xác suất thống kê, nhiệm vụ của ta là tìm hệ số  $\hat{\theta}$  cực đại hóa hàm hợp lí không đầy đủ.

Ngoài ra, ta định nghĩa thêm **hàm hợp lí đầy đủ**:

$$L(\theta, X, Z) = P(X, Z|\theta) = \prod_{j=1}^K [f(X, \mu_j, \Sigma_j) p_j]^{\mathcal{I}(Z=j)}$$

trong đó,  $\mathcal{I}(Z=j) = 1$  nếu  $Z=j$  và  $\mathcal{I}(Z=j) = 0$  nếu  $Z \neq j$

Khi đó, nếu  $\mathbf{x}$  là vector quan sát của biến ngẫu nhiên  $X$ ,  $\mathbf{z}$  là vector quan sát của biến ngẫu nhiên  $Z$ , hàm hợp lí đầy đủ của ta sẽ có dạng:

$$L(\theta, \mathbf{x}, \mathbf{z}) = P(\mathbf{x}, \mathbf{z}|\theta) = \prod_{i=1}^N \prod_{j=1}^K [f(x_i, \mu_j, \Sigma_j) p_j]^{\mathbb{I}(z_i=j)}$$

Lấy  $\log$  cả hai vế của hàm hợp lí đầy đủ ta sẽ có dạng:

$$\log L(\theta; \mathbf{x}; \mathbf{z}) = \sum_{i=1}^N \sum_{j=1}^K \mathcal{I}(z_i=j) \left[ \log(p_j) - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2} (x_i - \mu_j)^T \Sigma_j^{-1} (x_i - \mu_j) - \frac{D}{2} \log 2\pi \right]$$

Mệnh đề sau đây sẽ là nền tảng toán học của thuật toán cực đại hóa kì vọng.



**Mệnh đề 5.1** Xét  $Q(\theta, \theta') = \mathbb{E}_{Z|X, \theta'}[\log L(\theta, X, Z)]$ . Khi đó nếu  $Q(\theta, \theta') > Q(\theta', \theta')$  thì ta sẽ có  $L(\theta, X) > L(\theta', X)$

**Chứng minh:**

Ta có:  $P(X, Z|\theta) = P(Z|\theta)P(X|Z, \theta)$  theo công thức xác suất.

$$\Rightarrow \log P(X|\theta) = \log P(X, Z|\theta) - \log P(Z|\theta)$$

$$\Rightarrow P(Z = i | X, \theta') \log P(X | \theta) = P(Z = i | X, \theta') \log P(X, Z | \theta) - P(Z = i | X, \theta') \log P(Z | X, \theta)$$

Lấy tổng  $i$  từ 1 đến  $K$  ta sẽ có:

$$\log P(X | \theta) = \sum_{i=1}^K P(Z = i | X, \theta') \log P(X, Z | \theta) - \sum_{i=1}^K P(Z = i | X, \theta') \log P(Z = i | X, \theta)$$

Đặt  $H(\theta, \theta') = -\sum_{i=1}^K P(Z = i | X, \theta') \log P(Z = i | X, \theta)$  và chú ý rằng ta có:  $Q(\theta, \theta') = \sum_{i=1}^K P(Z = i | X, \theta') \log P(X, Z | \theta) \Rightarrow \log P(X|\theta) = Q(\theta, \theta') + H(\theta, \theta') \quad \forall \theta$  bất kỳ.

$$\Rightarrow \log P(X | \theta) - \log P(X | \theta') = Q(\theta | \theta') - Q(\theta' | \theta') + H(\theta | \theta') - H(\theta' | \theta')$$

Mặt khác ta lại có:  $H(\theta | \theta') - H(\theta' | \theta') = D_{KL}(P(Z | X, \theta') \| P(Z | X, \theta)) \geq 0$

Trong đó  $D_{KL}$  là khoảng cách **Kullback–Leibler**, là một phép đo cách một phân phối xác suất khác biệt so với cái còn lại, phân phối xác suất tham chiếu.

Vậy ta sẽ có:  $\log P(X | \theta) - \log P(X | \theta') \geq Q(\theta | \theta') - Q(\theta' | \theta')$ .  $\square$

Vậy dựa vào **mệnh đề 5.1** ở trên, ta không nhất thiết là phải đi tìm cực đại của hàm hợp lí không đầy đủ mà thay vào đó, ở mỗi vòng lặp, ta sẽ đi tìm giá trị cực đại của hàm  $Q(\theta, \theta')$  trong đó  $\theta'$  là giá trị tham số tìm được ở vòng lặp thứ  $t$ .

Hàm  $Q(\theta, \theta')$  sẽ được tính như sau:

$$\begin{aligned} Q(\theta, \theta') &= \mathbb{E}_{Z|X, \theta'}[\log L(\theta, \mathbf{x}, \mathbf{z})] = \mathbb{E}_{Z|X, \theta'} \left[ \log \prod_{i=1}^N L(\theta, x_i, z_i) \right] \\ &= \sum_{i=1}^N \mathbb{E}_{Z|X, \theta'} [\log L(\theta, x_i, z_i)] = \sum_{i=1}^N \sum_{j=1}^K P(z_i = j | X = x_i, \theta') \log L(\theta, x_i, z_i = j) \\ &= \sum_{i=1}^N \sum_{j=1}^K P(z_i = j | X = x_i, \theta') \left[ \log(p_j) - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2} (x_i - \mu_j)^T (\Sigma_j)^{-1} (x_i - \mu_j) - \frac{D}{2} \log 2\pi \right] \end{aligned}$$

Ngoài ra theo công thức Bayes, ta có:

$$\begin{aligned}\widehat{z}_{ji} &= P(z_i = j | X = x_i, \theta^t) = \frac{P(z_i = j) P(X = x_i, \theta^t | z_i = j)}{\sum_{j=1}^K P(z_i = j) P(X = x_i, \theta^t | z_i = j)} \\ &= \frac{p_j^{(t)} f(x_i, \mu_j^{(t)}, \Sigma_j^{(t)})}{\sum_{j=1}^K p_j^{(t)} f(x_i, \mu_j^{(t)}, \Sigma_j^{(t)})}\end{aligned}\quad (13)$$

$$\Rightarrow Q(\theta, \theta^t) = \sum_{i=1}^N \sum_{j=1}^K \widehat{z}_{ji} \left[ \log(p_j) - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2} (x_i - \mu_j)^T (\Sigma_j)^{-1} (x_i - \mu_j) - \frac{D}{2} \log 2\pi \right]$$

Như vậy ta đã tính xong hàm  $Q(\theta, \theta^t)$ . Chú ý rằng các hệ số  $\widehat{z}_{ji}$  ta tính được dựa vào tham số  $\theta^t$  và hàm  $Q(\theta, \theta^t)$  có các biến là  $(\{p_i\}_{i=1}^K, \{\mu_i\}_{i=1}^K, \{\Sigma_i\}_{i=1}^K)$ .

Sau khi đã tính hàm  $Q(\theta, \theta^t)$ , nhiệm vụ của ta bây giờ sẽ là tìm  $\theta$  là điểm cực đại của hàm  $Q$ . Cụ thể, ta sẽ phải đi giải bài toán tối ưu sau:

$$\sum_{i=1}^N \sum_{j=1}^K \widehat{z}_{ji} \left[ \log(p_j) - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2} (x_i - \mu_j)^T (\Sigma_j)^{-1} (x_i - \mu_j) - \frac{D}{2} \log 2\pi \right] \quad (P')$$

với điều kiện

$$\begin{cases} \sum_{i=1}^K p_i = 1 \\ p_i \geq 0 \quad \forall i = \{1, 2, \dots, K\} \end{cases}$$

Bài toán  $(P')$  là bài toán tối ưu phi tuyến có ràng buộc. Để giải bài toán  $(P')$  ta sẽ sử dụng phương pháp nhân tử Lagrange. Và theo [9], điểm cực đại của bài toán  $(P')$  là:

$$\begin{cases} \widehat{p}_k = \frac{n_k}{N} \\ \widehat{\mu}_k = \frac{\sum_{i=1}^N \widehat{z}_{ik} x_i}{n_k} \\ n_k = \sum_{i=1}^N \widehat{z}_{ik} \\ \widehat{\Sigma}_k = \frac{\sum_{i=1}^N \widehat{z}_{ik} (x_i - \widehat{\mu}_k)(x_i - \widehat{\mu}_k)^T}{\sum_{i=1}^N \widehat{z}_{ik}} \end{cases} \quad (14)$$

Công thức 14 sẽ được tính toán lần lượt như sau. Đầu tiên ta sẽ tính các hệ số  $n_k$ , sau đó ta sẽ tính các hệ số  $\widehat{p}_k$ . Tiếp theo ta sẽ tính toán các hệ số  $\widehat{\mu}_k$ . Cuối cùng, ta sẽ đi tính các ma trận hiệp phương sai  $\widehat{\Sigma}_k$ .

Do chỉ cập nhật các hệ số như trong công thức 13 và trong công thức 14, thuật toán cực đại hóa kì vọng sẽ không tốn quá nhiều tài nguyên tính toán như các phương pháp sử dụng vòng lặp như phương pháp hướng giảm hay phương pháp Newton-Raphson. Mã giả của thuật toán cực đại hóa kì vọng sẽ được miêu tả trong thuật toán 4.

---

**Algorithm 4** Mã giả thuật toán cực đại hóa kì vọng

---

- 1: **Input:** Số phân cụm  $K$ , phân phối tiên nghiệm  $f_{Mix}$ , số vòng lặp tối đa  $T$ ,  $t = 0$ .
  - 2: Thực hiện việc phân cụm ban đầu bằng các phương pháp như K-mean hay liên kết đơn để khởi tạo các hệ số  $\{p_j\}_{j=1}^K$ .
  - 3: Khởi tạo các giá trị ban đầu  $\{\mu_j\}_{j=1}^K; \{\Sigma_j\}_{j=1}^K$
  - 4: **while**  $t \leq T$  **do**
  - 5:     Thực hiện bước  $E$ : Cập nhật các hệ số  $\hat{z}_{ij}$  như trong công thức 13.
  - 6:     Thực hiện bước  $M$ : Cập nhật các hệ số  $\left(\{\hat{p}_i\}_{i=1}^K, \{\hat{\mu}_i\}_{i=1}^K, \{\hat{\Sigma}_i\}_{i=1}^K\right)$  như trong công thức 14.
  - 7:      $t := t + 1$
  - 8: **end while**
  - 9: **Output:** Bộ hệ số tối ưu  $\hat{\theta} = \left(\{\hat{p}_i\}_{i=1}^K, \{\hat{\mu}_i\}_{i=1}^K, \{\hat{\Sigma}_i\}_{i=1}^K\right)$ .
- 

Như vậy, với số phân cụm cố định, dựa vào thuật toán cực đại hóa kì vọng ta đã xác định được các bộ hệ số tối ưu. Tuy nhiên, trong thực tế, việc xác định số phân cụm là một điều khó khăn. Do thuật toán cực đại hóa kì vọng không cho ta cách xác định số phân cụm, ta sẽ cần một phương pháp có thể giúp ta xác định được số phân cụm hợp lí.

#### 5.1.4 Xác định số phân cụm dựa vào tiêu chuẩn AIC và BIC

Để xác định số phân cụm, ta sẽ dựa vào tiêu chuẩn đánh giá sau:

$$L_{\text{total}} = -2 \log(L_{\text{max}}) + \text{Penalty}$$

Trong đó:

- $L_{\text{max}} = L\left(\{\hat{p}_i\}_{i=1}^K, \{\hat{\mu}_i\}_{i=1}^K, \{\hat{\Sigma}_i\}_{i=1}^K\right)$  với  $\left(\{\hat{p}_i\}_{i=1}^K, \{\hat{\mu}_i\}_{i=1}^K, \{\hat{\Sigma}_i\}_{i=1}^K\right)$  là bộ hệ số tối ưu với  $K$  phân cụm.
- Penalty là một hàm phạt phụ thuộc vào độ phức tạp của mô hình (thông thường phụ thuộc vào số biến).

Dựa vào tiêu chuẩn trên, ta sẽ đi tìm số phân cụm  $K$  sao cho  $L_{\text{total}}$  là thấp nhất.

Có rất nhiều cách để xác định hàm Penalty, sau đây chúng em sẽ xác định hàm Penalty thông qua số biến của mô hình. Cụ thể, đối với mô hình của ta, tổng số biến sẽ được tính như sau: Giả sử mô hình được phân thành  $K$  cụm, mỗi cụm cần xác định các vector kì vọng  $\mu_k \in \mathbb{R}^D$  và các ma trận hiệp phương sai  $\Sigma_k \in \mathbb{R}^{D \times D}$ .

$\Rightarrow$  Tổng cộng lại đối với vector kì vọng ta cần xác định  $K \times D$  biến, còn đối với ma trận hiệp phương sai, ta cần xác định  $K \times \frac{D(D+1)}{2}$  biến (do ma trận hiệp phương sai là ma trận đối xứng).

Ngoài ra ta cũng cần xác định xác suất một điểm dữ liệu thuộc phân cụm  $k$  ( $p_k$ )  $\Rightarrow$  Ta cần xác định thêm  $K - 1$  biến.

Vậy tổng cộng lại, với  $K$  cụm, mô hình của chúng ta sẽ có tất cả  $K - 1 + K \times D + K \times \frac{D(D+1)}{2} = \frac{K}{2}(D + 1)(D + 2) - 1$  biến.

Dựa vào số biến của mô hình, ta sẽ có 2 tiêu chuẩn đánh giá xác định số phân cụm:

- Tiêu chuẩn Akaike (AIC): Penalty =  $2 \times \text{Số điểm dữ liệu} \times \text{Số biến của mô hình}$

$$\Rightarrow \text{AIC} = -2\log(L_{\max}) + 2N \left( \frac{K}{2}(D + 1)(D + 2) - 1 \right)$$

- Tiêu chuẩn Bayesian (BIC): Penalty =  $2 \times \log(\text{Số điểm dữ liệu}) \times \text{Số biến của mô hình}$

$$\Rightarrow \text{BIC} = -2\log(L_{\max}) + 2\log(N) \left( \frac{K}{2}(D + 1)(D + 2) - 1 \right)$$

**Nhận xét:** tiêu chuẩn BIC và tiêu chuẩn AIC khá giống nhau. Tuy nhiên, tiêu chuẩn BIC sẽ phù hợp hơn đối với các bộ dữ liệu lớn còn tiêu chuẩn AIC sẽ phù hợp với các bộ dữ liệu nhỏ hơn.

**Chú ý:** Dựa vào công thức tính số biến, ta thấy phần lớn các hệ số cần tìm nằm ở ma trận hiệp phương sai. Do đó, để cho mô hình đơn giản hơn mà vẫn giữ được độ chính xác cao, ta hay giả thiết các ma trận hiệp phương sai tuân theo một dạng cụ thể nào đấy. Bảng dưới đây sẽ cho ta thấy các dạng giả định hay dùng của ma trận hiệp phương sai, đồng thời cho ta biết số biến cũng như tiêu chuẩn BIC dùng cho mô hình.

Dạng giả định cho $\Sigma_k$	Tổng hệ số	Tiêu chuẩn BIC
$\Sigma_k = \eta \mathbb{I}$	$K(D + 1)$	$-2\ln(L_{\max}) + 2\ln(N)K(D + 1)$
$\Sigma_k = \eta_k \mathbb{I}$	$K(D + 2) - 1$	$-2\ln(L_{\max}) + 2\ln(N)(K(D + 2) - 1)$
$\Sigma_k = \eta_k \text{Diag}(\lambda_1, \lambda_2, \dots, \lambda_D)$	$K(D + 2) + D - 1$	$-2\ln(L_{\max}) + 2\ln(N)(K(D + 2) + D - 1)$

## 5.2 Kết quả thực nghiệm

Ta sẽ sử dụng bộ dữ liệu Iris để chạy kết quả thực nghiệm cho thuật toán phân cụm theo phương pháp xác suất thống kê. Để chạy thuật toán, chúng ta sẽ sử dụng thư viện scipy của python.

Đầu tiên, ta sẽ chạy dữ liệu với số phân cụm là  $K = 3$ . Sau khi chạy ta được kết quả như hình 22. Chú ý rằng bộ dữ liệu Iris của chúng ta mỗi điểm dữ liệu sẽ có 4 chiều bao gồm các trường dữ liệu như *sepal length*, *sepal width*, *petal length*, *petal width*. Do đó khi thực hiện phương pháp phân cụm theo mô hình xác suất thống kê, ta cũng có các bộ hệ số  $\mu_k \in \mathbb{R}^4$  và  $\Sigma_k \in \mathbb{R}^{4 \times 4}$ . Tuy nhiên, để có hình dung trực quan hơn, nhóm bọn em đã chiếu xuống không gian 2 chiều. Cụ thể, bọn em đã chiếu xuống chiều tương ứng với 2 trường dữ liệu là *petal length* và *petal width*. Các hệ số ma trận hiệp phương sai và kì vọng tối ưu sẽ được hiển thị trong hình 21.

```
Các ma trận hiệp phương sai tối ưu là:
[[[0.121765  0.097232  0.016028  0.010124 ]
  [0.097232  0.140817  0.011464  0.009112 ]
  [0.016028  0.011464  0.029557  0.005948 ]
  [0.010124  0.009112  0.005948  0.010885 ]]]

[[[0.38744093 0.09223276 0.30244302 0.06087397]
  [0.09223276 0.11040914 0.08385112 0.05574334]
  [0.30244302 0.08385112 0.32589574 0.07276776]
  [0.06087397 0.05574334 0.07276776 0.08484505]]]

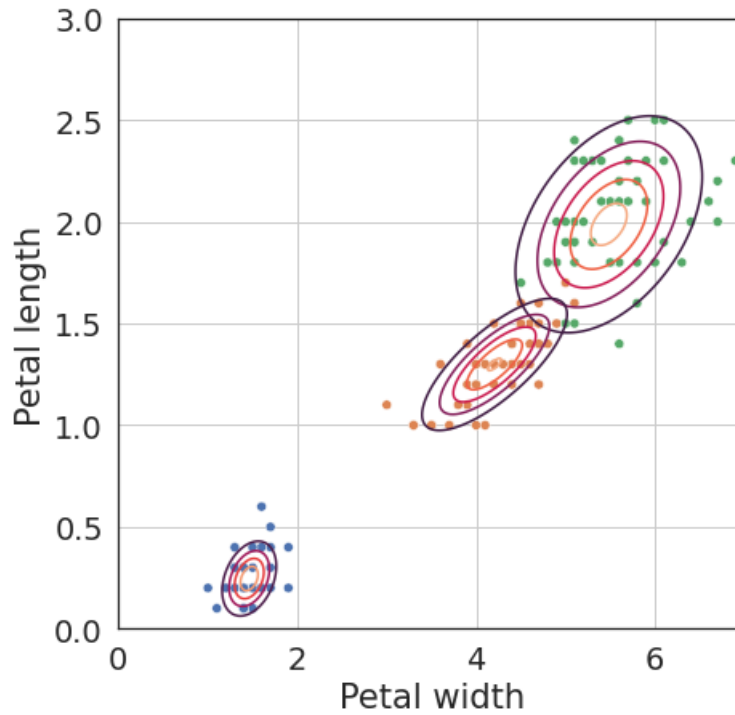
[[[0.2755171  0.09662295 0.18547072 0.05478901]
  [0.09662295 0.09255152 0.09103431 0.04299899]
  [0.18547072 0.09103431 0.20235849 0.06171383]
  [0.05478901 0.04299899 0.06171383 0.03233775]]]]

-----
Các kì vọng tối ưu là:
[[[5.006      3.428      1.462      0.246      ]
  [6.54639415 2.94946365 5.48364578 1.98726565]
  [5.9170732  2.77804839 4.20540364 1.29848217]]]
```

Hình 21: Các hệ số ma trận hiệp phương sai và kì vọng tối ưu với số phân cụm  $K=3$

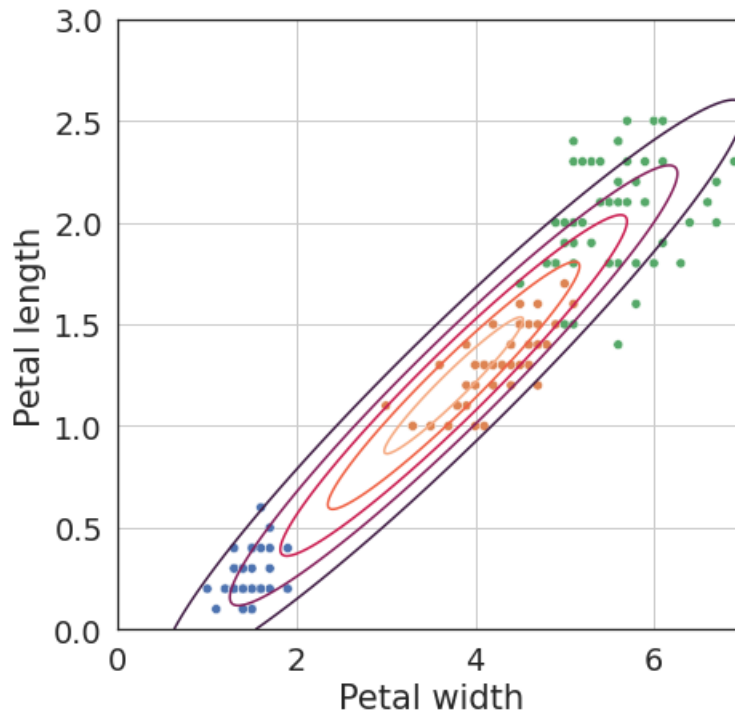
**Nhận xét:** Để ý rằng các đường đồng mức của hàm mật độ phân phối chuẩn 2 chiều sẽ là các hình ellipsoid. Nhìn vào hình 22 ta có thể thấy, thuật toán cho ta 3 phân phối chuẩn nhiều chiều phân cụm được rõ ràng bộ dữ liệu Iris. Các hình ellipsoid trên hình 22, không những cho ta biết tâm của từng cụm dữ liệu mà còn cho ta biết độ rải rác của cụm dữ liệu. Đây là một lợi thế nổi trội của phương pháp xác suất thống kê so với phương pháp K-means.

Để tìm số phân cụm hợp lí, ta sẽ áp dụng phương pháp xác suất thống kê cho  $K = 1, 2, 3, 4, 5$  và quan sát chỉ số AIC ứng với từng mô hình.

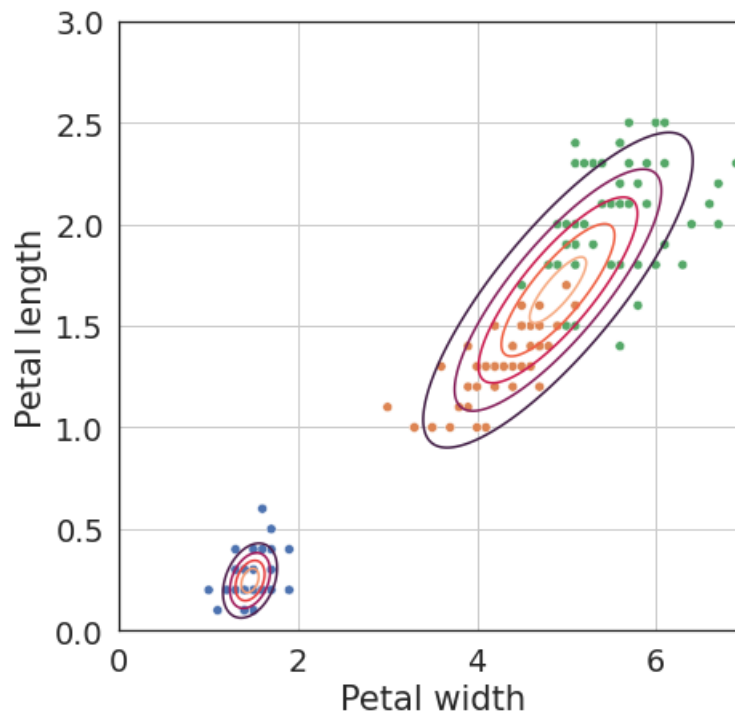


Hình 22: Thuật toán phân cụm xác suất thống kê với  $K = 3$ ,  $AIC = 448.39$

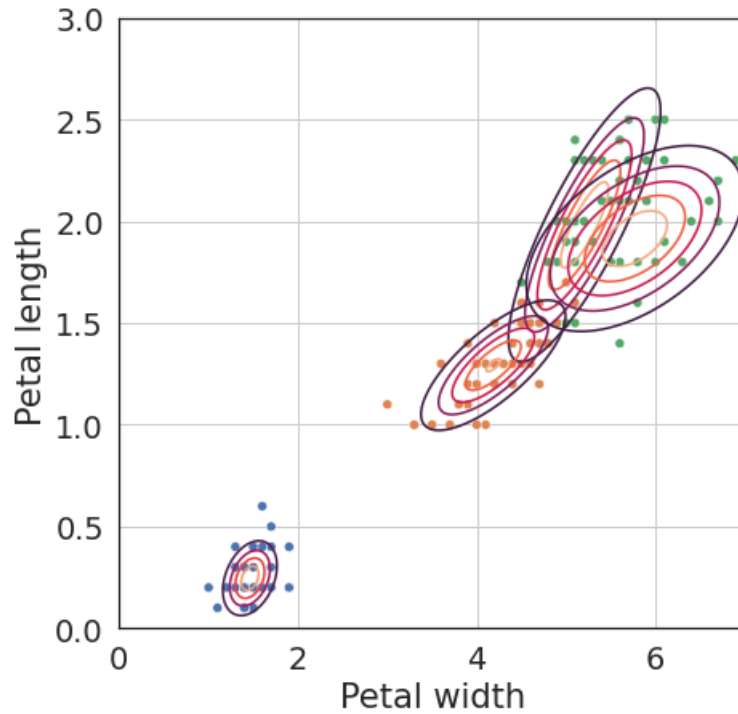
**Nhận xét:** Dựa vào các hình 22, 23, 24, 25, 26 ta thấy được với số phân cụm  $K = 3$  thì chỉ số AIC của ta là thấp nhất so với các số phân cụm  $K$  khác. Do đó, ta có thể chọn  $K = 3$  là số phân cụm của mô hình. Điều này phù hợp với bộ dữ liệu, bởi bộ dữ liệu Iris có tất cả 3 loài hoa khác nhau tương ứng với 3 cụm dữ liệu.



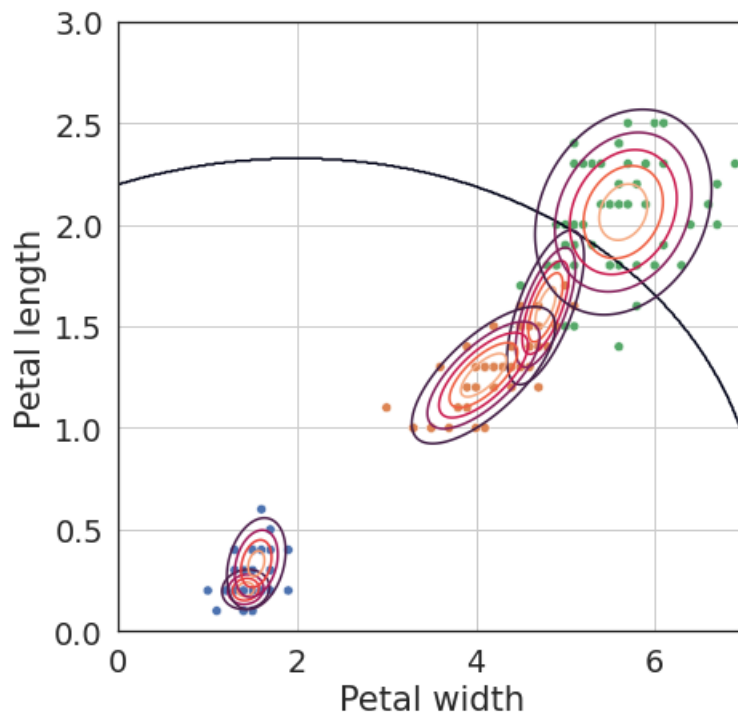
Hình 23: Thuật toán phân cụm xác suất thống kê với  $K = 1$ , AIC = 787.82



Hình 24: Thuật toán phân cụm xác suất thống kê với  $K = 2$ , AIC = 486.70



Hình 25: Thuật toán phân cụm xác suất thống kê với  $K = 4$ , AIC = 449.59



Hình 26: Thuật toán phân cụm xác suất thống kê với  $K = 5$ , AIC = 452.02



## 6 Thuật toán phân cụm theo phổ (Spectral Clustering)

### 6.1 Giới thiệu

Trong chương này sẽ trình bày về thuật toán phân cụm mới là phân cụm phổ. Chương này được tổ chức như sau: Hai phần đầu tiên được dành để giới thiệu từng bước về các đối tượng toán học được sử dụng bởi phân cụm quang phổ: đồ thị tương tự trong Phần 2 và đồ thị Laplace trong Phần 3. Các thuật toán phân cụm phổ sẽ được trình bày trong Phần 4. Phần 5 đưa ra quan một quan điểm về lát cắt trong đồ thị để chứng minh hiệu quả của thuật toán. Trong phần 6 sẽ thảo luận về các vấn đề khi sử dụng thuật toán phân cụm phổ, và cuối cùng Phần 7 đưa ra kết quả khi áp dụng vào vấn đề thực tế.

### 6.2 Đồ thị tương tự

Cho một tập điểm dữ liệu  $x_1, x_2, \dots, x_n$  và kí hiệu  $s_{ij} \geq 0$  là độ tương tự giữa của tất cả các cặp điểm dữ liệu  $x_i$  và  $x_j$ , mục tiêu trực quan của phân cụm là chia các điểm dữ liệu thành nhiều nhóm sao cho các điểm trong cùng một nhóm tương tự nhau và các điểm trong các nhóm khác nhau thì có độ khác nhau lớn. Nếu chúng ta không biết gì nhiều ngoại trừ đã biết thông tin về độ tương tự giữa các điểm dữ liệu thì một cách hay để biểu diễn dữ liệu là dưới dạng đồ thị tương tự  $G = (V, E)$ . Mỗi đỉnh  $v_i$  trong đồ thị này đại diện cho một điểm dữ liệu  $x_i$ . Hai đỉnh được nối với nhau nếu độ tương tự giữa chúng lớn hơn một ngưỡng nào đó, và cạnh của nó được đánh một trọng số  $s_{ij}$ .

Vấn đề phân cụm bây giờ có thể được định dạng lại bằng cách sử dụng đồ thị tương tự: chúng ta muốn tìm một phân hoạch của đồ thị sao cho các cạnh giữa các nhóm khác nhau có trọng số rất thấp (có nghĩa là các điểm trong các cụm khác nhau không giống nhau) và các cạnh trong một nhóm có trọng số cao (có nghĩa là các điểm trong cùng một cụm tương tự nhau). Để công thức hóa vấn đề này, trước tiên chúng ta sẽ đi vào một số ký hiệu đồ thị cơ bản và thảo luận ngắn gọn về loại đồ thị sẽ nghiên cứu.

#### 6.2.1 Ký hiệu đồ thị

Cho  $G = (V, E)$  là một đồ thị vô hướng với tập đỉnh là  $V = v_1, \dots, v_n$ . Trong đó, ta giả sử rằng  $G$  là đồ thị có trọng số, mỗi cạnh nối giữa 2 đỉnh  $v_i$  và  $v_j$  được đánh trọng số không âm  $w_{ij} \geq 0$ . Ma trận trọng số của đồ thị là  $W = (w_{ij})_{i,j=1,\dots,n}$ . Nếu  $w_{ij} = 0$ , nghĩa là các đỉnh  $v_i$  và  $v_j$  không được nối với nhau. Vì  $G$  là đồ thị vô hướng nên  $W$  là ma trận đối xứng. Bậc của đỉnh

$v_i \in V$  được định nghĩa như sau:

$$d_i = \sum_{j=1}^n w_{ij}$$

Ma trận bậc  $D$  được định nghĩa như là ma trận đường chéo với các bậc  $d_1, \dots, d_n$  nằm trên đường chéo chính. Cho một tập các đỉnh  $A \subset V$ , ta kí hiệu phần bù của  $A$  là  $\bar{A}$ . Chúng ta định nghĩa  $\mathbf{1}_A = (f_1, \dots, f_n)' \in \mathbb{R}^n$  là một vector với các phần tử  $f_i = 1$  nếu  $v_i \in A$  và  $f_i = 0$  trường hợp còn lại. Cho hai tập  $A, B \subset V$ , ta định nghĩa

$$W(A, B) := \sum_{i \in A, j \in B} w_{ij}$$

Ta xem xét hai cách khác nhau để đo "kích thước" của một tập hợp con  $A \subset V$ :

$$|A| := \text{số đỉnh nằm trong tập } A,$$

$$\text{vol}(A) := \sum_{i \in A} d_i$$

Tập  $A$  được gọi là liên thông nếu hai đỉnh bất kì luôn tồn tại một đường đi sao cho tất cả các đỉnh nằm trên nó đều thuộc  $A$ . Một tập  $A$  được gọi là thành phần liên thông nếu nó liên thông và không có cạnh nối giữa các đỉnh trong  $A$  và  $\bar{A}$ . Các tập đỉnh khác rỗng  $A_1, \dots, A_k$  là một cách phân chia đồ thị nếu  $A_i \cap A_j = \emptyset$  và  $A_1 \cup A_2 \dots \cup A_k = V$ .

### 6.2.2 Các loại đồ thị tương tự

Xây dựng đồ thị tương tự cho phân cụm phổ không phải là nhiệm vụ dễ dàng, do có rất ít cơ sở lý thuyết. Có một số cách xây dựng phổ biến để biến đổi một tập hợp nhất định  $x_1, \dots, x_n$  của các điểm dữ liệu có độ tương tự theo từng cặp  $s_{ij}$ . Khi xây dựng các biểu đồ tương tự, mục tiêu là mô hình hóa các mối quan hệ lân cận cục bộ giữa các điểm dữ liệu.

- **Đồ thị  $\varepsilon$  lân cận:** Ở đây chúng ta kết nối tất cả các điểm có khoảng cách theo chiều cặp nhỏ hơn  $\varepsilon$ . Vì khoảng cách giữa tất cả các điểm được kết nối gần như có cùng một tỷ lệ (tối đa là  $\varepsilon$ ), trọng số các cạnh sẽ không kết hợp thêm thông tin về dữ liệu vào biểu đồ. Do đó, đồ thị  $\varepsilon$  lân cận thường được coi là như một đồ thị không có trọng số.
- **Đồ thị lân cận  $k$  gần nhất:** Ở đây mục đích là nối đỉnh  $v_i$  với đỉnh  $v_j$  nếu  $v_j$  nằm trong  $k$  lân cận gần nhất của  $v_i$ . Tuy nhiên, định nghĩa này dẫn đến một đồ thị có hướng, vì mỗi quan hệ lân cận không đối xứng. Có hai cách để làm cho biểu đồ này vô hướng. Cách đầu tiên đơn giản là bỏ qua hướng của các cạnh, tức là chúng ta nối  $v_i$  và  $v_j$  với một cạnh vô

hướng nếu  $v_i$  nằm trong số  $k$  lân cận gần nhất của  $v_j$  hoặc nếu  $v_j$  nằm trong  $k$  lân cận gần nhất của  $v_i$ . Đồ thị kết quả thường được gọi là đồ thị lân cận  $k$  gần nhất. Lựa chọn thứ hai là nối các đỉnh  $v_i$  và  $v_j$  nếu cả  $v_i$  nằm trong số  $k$  lân cận gần nhất của  $v_j$  và  $v_j$  nằm trong  $k$  lân cận gần nhất của  $v_i$ . Đồ thị kết quả được gọi là đồ thị lân cận tương hỗ  $k$  gần nhất. Trong cả hai trường hợp, sau khi nối các đỉnh thích hợp, chúng ta đánh trọng số các cạnh bằng độ tương tự của các đỉnh đầu mút.

- **Đồ thị liên thông toàn phần:** Ở đây chúng ta chỉ cần kết nối tất cả các điểm có độ tương tự dương với nhau và chúng ta đánh trọng số tất cả các cạnh bằng  $s_{ij}$ . Vì đồ thị phải đại diện cho các mối quan hệ vùng lân cận cục bộ, cấu trúc này thường chỉ được chọn nếu bản thân hàm tương tự mô hình được các vùng lân cận địa phương.

### 6.3 Đồ thị Laplace và đặc điểm

Công cụ chính cho phân cụm phổ là các ma trận đồ thị Laplace. Tồn tại cả một lĩnh vực dành cho nghiên cứu các ma trận này, gọi là lý thuyết đồ thị phổ [3]. Trong phần này chúng ta định nghĩa các ma trận đồ thị Laplace khác nhau. Chú ý rằng trong tài liệu, không có quy tắc duy nhất gọi tên các ma trận đồ thị Laplace. Thông thường, mỗi tác giả chỉ gọi ma trận của “anh ta” là ma trận Laplace. Do đó cần thận trọng khi đọc tài liệu về ma trận đồ thị Laplace.

Sau đây chúng ta giả sử rằng  $G = (V, E)$  là đồ thị vô hướng có trọng số với ma trận trọng số là  $W$ , trong đó  $w_{ij} \geq 0$ . Khi sử dụng vector riêng của một ma trận, chúng ta sẽ không cần giả sử rằng chúng được chuẩn hóa. Ví dụ, vector hằng  $\mathbf{1}$  và bội  $a\mathbf{1}$  với  $a \neq 0$  sẽ được coi như các vector riêng giống nhau. Các giá trị riêng sẽ luôn được sắp xếp tăng dần tương ứng với số bội. “ $k$  vectơ riêng đầu tiên” được quy cho những vector riêng tương ứng với  $k$  giá trị riêng nhỏ nhất.

#### 6.3.1 Ma trận Laplace không chuẩn hóa

Ma trận của đồ thị Laplace không chuẩn hóa được xác định bởi Mohar[2]:

$$L = D - W$$

Mệnh đề sau đây trình bày các đặc điểm quan trọng cho việc phân cụm quang phổ.

**Mệnh đề 6.1 (Đặc điểm ma trận  $L$ )** Ma trận thỏa mãn một số tính chất sau:

1. Với mọi vector  $f \in \mathbb{R}^n$  ta có:

$$f'Lf = \frac{1}{2} \sum_{i,j=1}^n w_{ij} (f_i - f_j)^2$$

2.  $L$  là ma trận đối xứng nửa xác định dương

3. Giá trị riêng nhỏ nhất của  $L$  là 0 ứng với vector riêng là  $\mathbf{1}$

4.  $L$  có  $n$  giá trị riêng thực không âm

Lưu ý rằng đồ thị chưa chuẩn hóa Laplacian không phụ thuộc vào các phần tử đường chéo của ma trận trọng số  $W$ . Đặc biệt, các cạnh khuyên trong đồ thị không thay đổi ma trận đồ thị Laplace. Ta có thể xem nhiều đặc điểm của trị riêng và vector riêng của ma trận trong Mohar [2]. Một đặc điểm quan trọng cho phân cụm phổ được trình bày như sau:

**Mệnh đề 6.2 (Số thành phần liên thông và phổ của  $L$ )** Cho  $G$  là đồ thị vô hướng có trọng số không âm. Bội  $k$  của trị riêng 0 của  $L$  bằng với số thành phần liên thông  $A_1, \dots, A_k$  trong đồ thị. Không gian riêng của trị riêng 0 có cơ sở là  $\{\mathbf{1}_{A_1}, \dots, \mathbf{1}_{A_k}\}$ .

### 6.3.2 Ma trận Laplace chuẩn hóa

Có hai cách định nghĩa ma trận của đồ thị Laplace chuẩn hóa trong các tài liệu tham khảo. Cả hai ma trận này quan hệ mật thiết với nhau và được định nghĩa như sau

$$L_{\text{sym}} := D^{-1/2}LD^{-1/2} = I - D^{-1/2}WD^{-1/2}$$

$$L_{\text{rw}} := D^{-1}L = I - D^{-1}W.$$

Chúng ta ký hiệu ma trận đầu tiên là  $L_{\text{sym}}$  bởi nó là một ma trận đối xứng, ma trận thứ hai là  $L_{\text{rw}}$  bởi nó liên hệ mật thiết với phương pháp bước đi ngẫu nhiên (random walk). Sau đây ta sẽ tóm tắt một vài đặc điểm quan trọng của hai ma trận  $L_{\text{sym}}$  và  $L_{\text{rw}}$ .

**Mệnh đề 6.3 (Đặc điểm của  $L_{\text{sym}}$  và  $L_{\text{rw}}$ )** Ma trận  $L_{\text{sym}}$  và  $L_{\text{rw}}$  thỏa mãn một số tính chất sau:

1. Với mọi vector  $f \in \mathbb{R}^n$ , ta có:

$$f'L_{\text{sym}}f = \frac{1}{2} \sum_{i,j=1}^n w_{ij} \left( \frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}} \right)^2$$

2.  $\lambda$  là một trị riêng của  $L_{\text{rw}}$  với vector riêng  $u$  nếu và chỉ nếu  $L_{\text{sym}}$  trị riêng là  $\lambda$  với vector riêng là  $D^{\frac{1}{2}}u$ .
3.  $0$  là vector riêng của  $L_{\text{rw}}$  với vector riêng là  $\mathbf{1}$ .  $0$  là trị riêng của  $L_{\text{sym}}$  với vector riêng là  $D^{\frac{1}{2}}\mathbf{1}$
4. Ma trận  $L_{\text{sym}}$  và  $L_{\text{rw}}$  là ma trận nửa xác định dương và có  $n$  giá trị riêng không âm.

Như trường hợp của ma trận Laplace không chuẩn hóa, bội của giá trị riêng  $0$  của ma trận chuẩn hóa Laplace có liên quan đến số lượng các thành phần liên thông:

**Mệnh đề 6.4** Cho  $G$  là đồ thị vô hướng có trọng số không âm. Bội  $k$  của trị riêng  $0$  của cả  $L_{\text{sym}}$  và  $L_{\text{rw}}$  bằng với số thành phần liên thông  $A_1, \dots, A_k$  trong đồ thị. Đối với  $L_{\text{rw}}$ , không gian riêng của trị riêng  $0$  có cơ sở là  $\{\mathbf{1}_{A_1}, \dots, \mathbf{1}_{A_k}\}$ .

## 6.4 Thuật toán phân cụm phổ

Bây giờ chúng ta đưa ra hai thuật toán phân cụm phổ phổ biến nhất: phân cụm phổ không chuẩn hóa, phân cụm phổ chuẩn hóa theo [12]. Ngoài ra, còn tồn tại một thuật toán phân cụm chuẩn hóa khác dựa theo quan điểm random walk, có thể xem [10].

### 6.4.1 Phân cụm phổ không chuẩn hóa

---

**Algorithm 5** Thuật toán Phân cụm phổ không chuẩn hóa

---

- 1: **Đầu vào:** Ma trận tương tự  $S = (s_{ij})_{i,j=1\dots n} \in \mathbb{R}^{n \times n}$ , số cụm  $k$  cần xây dựng.
  - 2: Xây dựng đồ thị tương tự đã được giới ở phần trên.
  - 3: Tính ma trận không chuẩn hóa  $L$ .
  - 4: Tính  $k$  vectơ riêng đầu tiên  $u_1, \dots, u_k$  của ma trận  $L$
  - 5: Đặt  $U \in \mathbb{R}^{n \times k}$  là ma trận gồm các cột là các vectơ  $u_1, \dots, u_k$
  - 6: Với  $i = 1, \dots, n$  đặt  $y_i \in \mathbb{R}^k$  là vector ứng với hàng thứ  $i$  của  $U$
  - 7: Phân cụm các điểm  $(y_i)_{i=1,\dots,n}$  trong  $\mathbb{R}^k$  với thuật toán k-means thành các cụm  $C_1, \dots, C_k$
  - 8: **Đầu ra:** Các cụm  $A_1, \dots, A_k$  với  $A_i = \{j \mid y_j \in C_i\}$
- 

Có hai phiên bản khác nhau của phân cụm phổ chuẩn hóa, phụ thuộc đồ thị Laplace chuẩn hóa được sử dụng. Tên của hai thuật toán được đặt theo hai bài báo phổ biến [12]. [10]. Tiếp theo, ta đi vào thuật toán chuẩn hóa theo Shi and Malik [12]

### 6.4.2 Phân cụm phổ chuẩn hóa theo Shi and Malik

---

**Algorithm 6** Thuật toán Phân cụm phổ chuẩn hóa
 

---

- 1: **Đầu vào:** Ma trận tương tự  $S = (s_{ij})_{i,j=1\dots n} \in \mathbb{R}^{n \times n}$ , số cụm  $k$  cần xây dựng.
  - 2: Xây dựng đồ thị tương tự đã được giới ở phần trên.
  - 3: Tính ma trận không chuẩn hóa  $L_{rw}$ .
  - 4: Tính  $k$  vectơ riêng đầu tiên  $u_1, \dots, u_k$  của ma trận  $L_{rw}$ .
  - 5: Đặt  $U \in \mathbb{R}^{n \times k}$  là ma trận gồm các cột là các vectơ  $u_1, \dots, u_k$ .
  - 6: Với  $i = 1, \dots, n$  đặt  $y_i \in \mathbb{R}^k$  là vector ứng với hàng thứ  $i$  của  $U$ .
  - 7: Phân cụm các điểm  $(y_i)_{i=1,\dots,n}$  trong  $\mathbb{R}^k$  với thuật toán k-means thành các cụm  $C_1, \dots, C_k$ .
  - 8: **Đầu ra:** Các cụm  $A_1, \dots, A_k$  với  $A_i = \{j \mid y_j \in C_i\}$ .
- 

## 6.5 Lát cắt đồ thị

Đối với dữ liệu được đưa ra dưới dạng một đồ tương tự, vấn đề này có thể được biểu diễn lại như sau: chúng ta muốn tìm một phân hoạch của biểu đồ sao cho các cạnh giữa các nhóm khác nhau có trọng số rất thấp (có nghĩa là các điểm trong các cụm khác nhau không giống nhau) và các cạnh trong một nhóm có trọng số cao (có nghĩa là các điểm trong cùng một cụm tương tự nhau). Trong phần này, chúng ta sẽ thấy cách phân cụm quang phổ có thể được suy ra như một phép gần đúng cho các bài toán phân vùng đồ thị như vậy.

Cho một đồ thị tương tự với ma trận trọng số  $W$ , cách đơn giản và trực tiếp nhất để xây dựng một phân chia của đồ thị là giải bài toán min-cut. Để xác định nó, ta nhắc lại kí hiệu  $W(A, B) := \sum_{i \in A, j \in B} w_{ij}$  và  $\bar{A}$  là phần bù của  $A$ . Cho  $k$  tập hợp con nhất định, cách tiếp cận min-cut chỉ đơn giản là chọn phân hoạch đồ thị thành  $k$  tập con  $A_1, \dots, A_k$  sao cho tối thiểu

$$\text{cut}(A_1, \dots, A_k) := \frac{1}{2} \sum_{i=1}^k W(A_i, \bar{A}_i)$$

Thực tế, đặc biệt đối với  $k = 2$ , mincut là một vấn đề tương đối dễ dàng và có thể được giải quyết một cách hiệu quả, có thể xem [13]. Tuy nhiên, trong thực tế nó thường không dẫn đến các phân vùng ưng ý. Vấn đề là trong nhiều trường hợp, lời giải của min-cut chỉ đơn giản là tách một đỉnh riêng lẻ khỏi phần còn lại của đồ thị. Tất nhiên, đây không phải là điều chúng ta muốn đạt được khi phân cụm, vì các cụm nên có số điểm lớn hợp lý. Một cách để giải quyết vấn đề này là yêu cầu rõ ràng rằng các bộ  $A_1, \dots, A_k$  là "lớn hợp lý". Hai hàm mục tiêu phổ biến nhất để mã hóa điều này là RatioCut (được giới thiệu [5]) và cắt chuẩn hóa Ncut (được giới thiệu

[12]). Định nghĩa của hai cắt được định nghĩa:

$$\begin{aligned} \text{RatioCut}(A_1, \dots, A_k) &:= \frac{1}{2} \sum_{i=1}^k \frac{W(A_i, \bar{A}_i)}{|A_i|} = \sum_{i=1}^k \frac{\text{cut}(A_i, \bar{A}_i)}{|A_i|} \\ \text{Ncut}(A_1, \dots, A_k) &:= \frac{1}{2} \sum_{i=1}^k \frac{W(A_i, \bar{A}_i)}{\text{vol}(A_i)} = \sum_{i=1}^k \frac{\text{cut}(A_i, \bar{A}_i)}{\text{vol}(A_i)}. \end{aligned} \quad (15)$$

Cả hai hàm mục tiêu đều cố gắng đạt được là các cụm “cân bằng”, được đo bằng số đỉnh hoặc trọng số cạnh, tương ứng. Thật không may, việc đưa ra các điều kiện cân bằng làm cho vấn đề đơn giản để giải quyết vấn đề mincut trở thành bài toán NP-hard. Phân cụm phổ là một cách để giải quyết các phiên nói lỏng của những vấn đề đó. Chúng ta sẽ chỉ ra rằng Ncut nói lỏng dẫn đến phân cụm phổ chuẩn hóa, trong khi RatioCut nói lỏng đến phân cụm phổ không chuẩn hóa.

### 6.5.1 Xấp xỉ RatioCut

Mục tiêu của chúng ta là giải quyết vấn đề tối ưu sau:

$$\min_{A \subset V} \text{RatioCut}(A, \bar{A})$$

Đầu tiên chúng ta viết lại vấn đề bằng một hình thức thuận tiện hơn. Cho một phân hoạch  $V$  thành  $k$  tập  $A_1, \dots, A_k$ , chúng ta xác định  $k$  vectơ chỉ số  $h_j = (h_{1,j}, \dots, h_{n,j})'$  như sau

$$h_{i,j} = \begin{cases} 1/\sqrt{|A_j|} & \text{if } v_i \in A_j \\ 0 & \text{otherwise} \end{cases} \quad (i = 1, \dots, n; j = 1, \dots, k). \quad (16)$$

Sau đó ta đặt ma trận  $H \in \mathbb{R}^{n \times k}$  là ma trận chứa  $k$  vectơ chỉ số đó như là cột của ma trận. Quan sát các cột trong  $H$  là ma trận trực chuẩn với nhau, tức là  $H'H = I$ . Tương tự như các phép tính ở phần trước, chúng ta có thể thấy rằng

$$h_i' L h_i = \frac{\text{cut}(A_i, \bar{A}_i)}{|A_i|}. \quad (17)$$

Hơn nữa, ta có thể thấy rằng

$$h_i' L h_i = (H' L H)_{ii} \quad (18)$$

Kết hợp những thứ đó lại với nhau, chúng ta sẽ có được

$$\text{RatioCut}(A_1, \dots, A_k) = \sum_{i=1}^k h_i' L h_i = \sum_{i=1}^k (H' L H)_{ii} = \text{Tr}(H' L H) \quad (19)$$

trong đó  $\text{Tr}$  là vết của ma trận. Vì vậy, vấn đề tối thiểu hàm  $\text{RatioCut}(A_1, \dots, A_k)$  có thể được viết lại thành

$$\min_{A_1, \dots, A_k} \text{Tr}(H' LH) \text{ với điều kiện } H'H = I, H \text{ được định nghĩa như 16}$$

Đây là bài toán tối ưu rời rạc với mỗi phần tử trong vector  $h_i$  chỉ nhận được hai giá trị, hiển nhiên đây vẫn là một bài toán NP-hard. Việc nối lỏng của bài toán này là loại bỏ điều kiện rời rạc và thay vào đó cho phép điều  $h_{i,j}$  được nhận giá trị  $\mathbb{R}$  bất kì. Bài toán được nối lỏng trở thành:

$$\min_{A_1, \dots, A_k} \text{Tr}(H' LH) \text{ với điều kiện } H'H = I$$

Đây là dạng tiêu chuẩn của một bài toán tối thiểu hàm vết của ma trận, và theo định lý Rayleigh-Ritz cho chúng ta biết rằng lời giải là chọn  $H$  là ma trận chứa  $k$  vector riêng đầu tiên của  $L$  dưới dạng như là cột của ma trận. Chúng ta có thể thấy rằng ma trận  $H$  thực chất là ma trận  $U$  được sử dụng trong thuật toán phân cụm phổ không chuẩn hóa như được mô tả trong phần trước. Bây giờ, chúng ta cần chuyển đổi lại ma trận nghiệm có giá trị thực thành một phân vùng rời rạc. Như trên, cách tiêu chuẩn là sử dụng thuật toán k-mean trên các hàng của  $U$ . Điều này dẫn đến thuật toán phân cụm phổ không chuẩn hóa chung như đã trình bày.

### 6.5.2 Xấp xỉ Ncut

Các kỹ thuật tương tự như các kỹ thuật được sử dụng cho RatioCut có thể được sử dụng để tìm ra cách phân cụm chuẩn hóa như nối lỏng hàm Ncut. Ta định nghĩa vector chỉ số  $h_j = (h_{1,j}, \dots, h_{n,j})'$  bởi:

$$h_{i,j} = \begin{cases} 1/\sqrt{\text{vol}(A_j)} & \text{if } v_i \in A_j \\ 0 & \text{otherwise} \end{cases} \quad (i = 1, \dots, n; j = 1, \dots, k). \quad (20)$$

Ta đặt  $H$  như là ma trận với  $k$  cột là các vector  $h_i$  như trên. Ta thấy rằng  $H'H = I$ ,  $h_i' D h_i = 1$ ,  $h_i' L h_i = \text{cut}(A_i, \bar{A}_i) / \text{vol}(A_i)$ . Vì vậy, ta có thể viết lại bài toán tối thiểu hàm Ncut như sau:

$$\min_{A_1, \dots, A_k} \text{Tr}(H' LH) \text{ với điều kiện } H'DH = I, H \text{ như trong 20}.$$

Nối lỏng điều kiện rời rạc, thay  $T = D^{1/2}H$  ta được bài toán nối lỏng:

$$\min_{T \in \mathbb{R}^{n \times k}} \text{Tr}(T' D^{-1/2} L D^{-1/2} T) \text{ subject to } T'T = I.$$



Đây là bài toán tối thiểu hóa vết ma trận được giải bằng ma trận  $T$  chứa  $k$  vector trị riêng đầu tiên của  $L_{\text{sym}}$ . Thay ngược lại  $H = D^{-1/2}T$  và sử dụng Mệnh đề 4 ta chỉ ra được  $H$  là ma trận gồm các cột là  $k$  vector riêng đầu tiên của ma trận  $L_{\text{TW}}$ . Điều này dẫn đến thuật phân cụm phổ theo Shi and Malik [12] đã trình bày ở trên.

Tất nhiên, cách nối lỏng trên không phải là cách duy nhất. Lí do việc nối lỏng trên được ưa thích không phải do nó đưa ra được cách phân cụm tốt nhất. Tính phổ biến của nó chủ yếu là do nó dẫn đến lớp bài toán đại số tuyến tính chuẩn rất đơn giản để giải.

## 6.6 Thảo luận

Trong phần này, chúng ta sẽ thảo luận ngắn gọn về một số vấn đề nảy sinh khi thực sự triển khai phân cụm phổ và một số cách giải quyết đối những vấn đề này.

### 6.6.1 Xây dựng đồ thị tương tự

Chúng ta minh họa vấn đề lựa chọn xây dựng bằng cách sử dụng ví dụ trình bày trong Hình 27. Với phân phối cơ bản, chúng ta chọn một phân phối trên  $\mathbb{R}^2$  với ba cụm: hai "mặt trăng" và một Gaussian. Mật độ của mặt trăng đáy được chọn lớn hơn mật độ của mặt trăng trên cùng. Hình phía trên bên trái trong Hình là một mẫu được rút ra từ bản phân phối này. Ba hình tiếp theo mô tả các đồ thị tương tự khác nhau trên mẫu này.

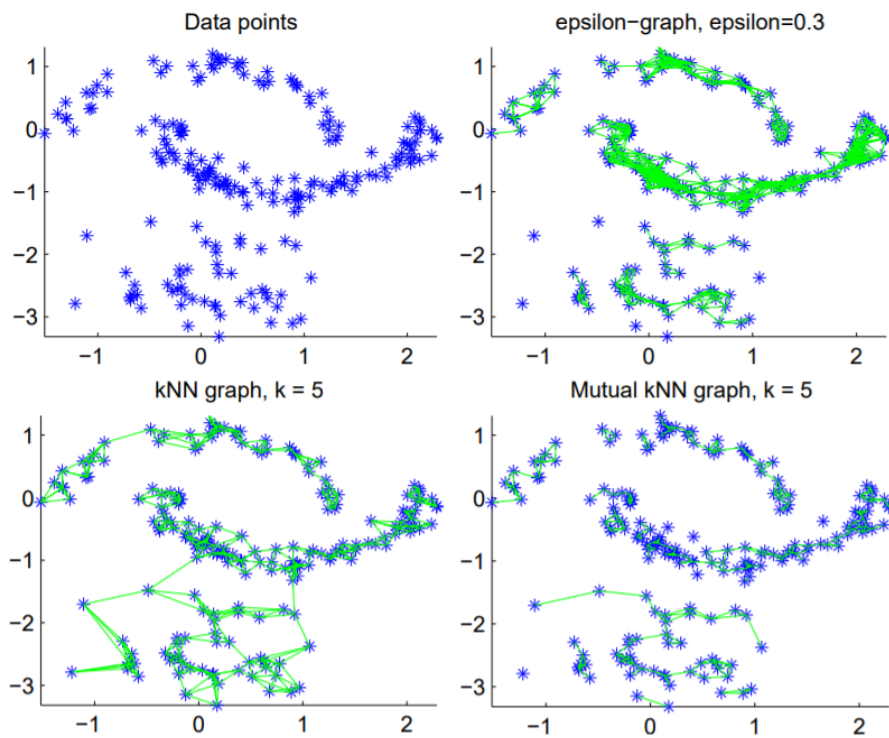
Trong đồ thị  $\varepsilon$ -lân cận, chúng ta có thể thấy rằng rất khó để chọn một tham số hữu ích  $\varepsilon$ . Với  $\varepsilon = 0.3$  như trong hình, các điểm trên mặt trăng giữa đã được nối dày đặc với nhau, trong khi các điểm trong Gaussian hầu như không được kết nối với nhau. Vấn đề này luôn xảy ra nếu chúng ta có dữ liệu "ở các tỷ lệ khác nhau", tức là khoảng cách giữa các điểm dữ liệu là khác nhau ở các vùng khác nhau của không gian.

Mặt khác, đồ thị lân cận  $k$ -gần nhất có thể kết nối các điểm "trên các tỷ lệ khác nhau". Chúng ta có thể thấy rằng các điểm trong mặt trăng mật độ thấp được kết nối với các điểm trong mặt trăng mật độ cao. Đây là thuộc tính chung khá hữu ích của đồ thị  $k$ -lân cận gần nhất. Chúng ta cũng có thể thấy rằng đồ thị lân cận  $k$ -gần nhất có thể chia thành nhiều thành phần không kết nối nếu có các vùng mật độ cao cách xa nhau một cách hợp lý. Ví dụ như hai mặt trăng ở trong hình.

Đồ thị lân cận tương hỗ  $k$ -gần nhất có đặc tính là nó có xu hướng kết nối các điểm trong các vùng có mật độ không đổi, nhưng không kết nối các vùng có mật độ khác nhau với nhau. Vì vậy, lân cận gần nhau nhất  $k$  có thể được coi là "ở giữa" đồ thị lân cận  $\varepsilon$  và đồ thị lân cận  $k$ -gần nhất. Nó có thể hoạt động trên các quy mô khác nhau, nhưng không trộn lẫn các quy mô đó với

nhau. Do đó, biểu đồ láng giềng gần nhau nhất  $k$  có vẻ đặc biệt phù hợp nếu chúng ta muốn phát hiện các cụm có mật độ khác nhau.

Đồ thị liên thông đầy đủ rất thường được sử dụng liên quan đến hàm tương tự Gaussian  $s(x_i, x_j) = \exp\left(-\|x_i - x_j\|^2 / (2\sigma^2)\right)$ . Ở đây, tham số  $\sigma$  đóng một vai trò tương tự như tham số  $\varepsilon$  trong  $\varepsilon$ -lân cận. Các điểm trong vùng lân cận địa phương được kết nối với trọng số tương đối cao, trong khi các cạnh giữa các điểm ở xa có trọng số dương, nhưng không đáng kể. Tuy nhiên, ma trận tương tự kết quả không phải là thưa thớt. Điều này khó cho các thuật toán tìm trị riêng của ma trận Laplace.



Hình 27: Các loại đồ thị tương tự khác nhau.

Theo một gợi ý chung, chúng nên làm việc với đồ thị  $k$  lân cận gần nhất là lựa chọn đầu tiên. Nó tạo ra ma trận  $W$  thưa để việc tính toán trị riêng của ma trận Laplace, và theo kinh nghiệm thì kết quả sau khi phân cụm ít bị ảnh hưởng bởi các lựa chọn tham số không phù hợp hơn so với các đồ thị khác.

### 6.6.2 *Số cụm*

Một công cụ được thiết kế đặc biệt để chọn số cụm là " heuristic eigengap" cho các thuật toán đã trình bày. Ở đây mục tiêu là chọn số  $k$  sao cho tất cả các giá trị riêng  $\lambda_1, \dots, \lambda_k$  rất nhỏ, nhưng  $\lambda_{k+1}$  là tương đối lớn. Điều này đặc biệt đúng nếu đồ thị có  $k$  thành phần liên thông.

### 6.6.3 *Lựa chọn thuật toán*

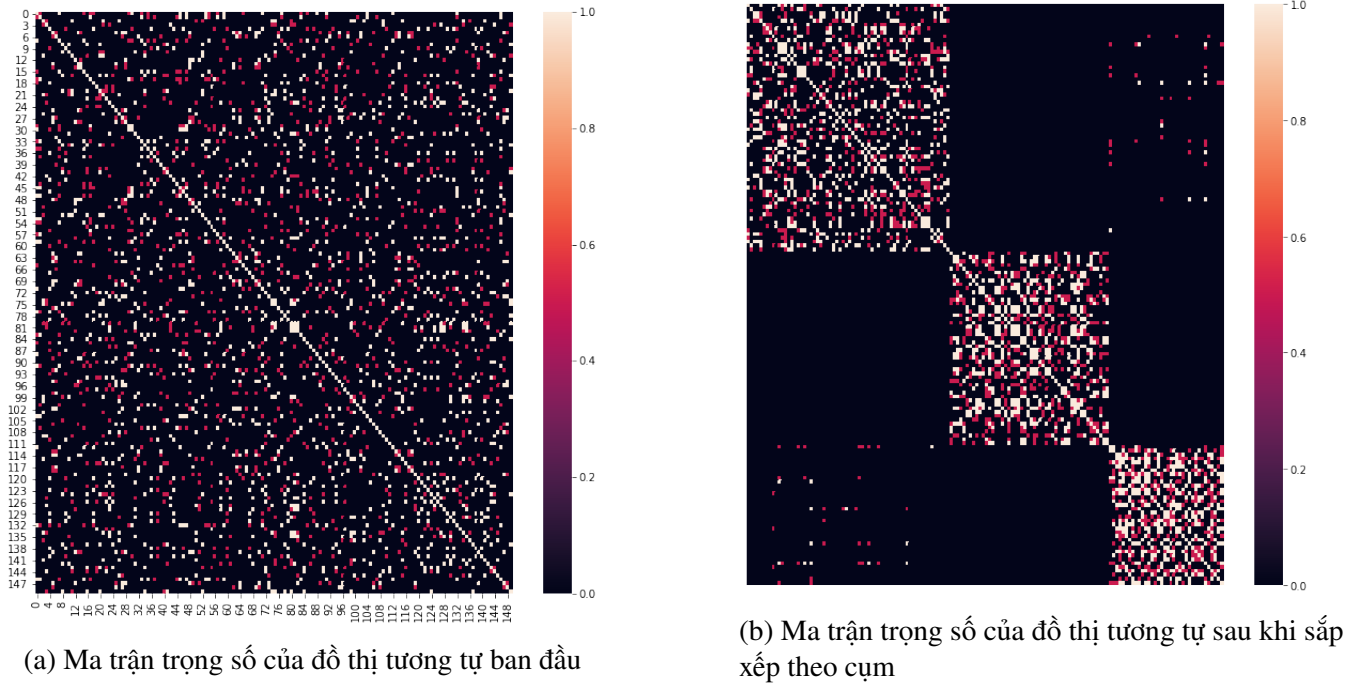
Hai thuật toán phân cụm không chuẩn hóa và chuẩn hóa sẽ đưa kết quả giống nhau nếu bậc của mọi đỉnh đều xấp xỉ nhau, đặc biệt nếu xây dựng đồ thị tương tự theo  $k$  lân cận gần nhất. Tuy nhiên, trong trường hợp bậc của đỉnh là khác nhau, thông thường, người ta muốn tối thiểu độ tương tự giữa các điểm khác cụm trong khi muốn tối đa độ tương tự các đỉnh trong cùng một cụm. Tối ưu RatioCut chỉ thực hiện yêu cầu thứ nhất khi tối ưu Ncut sẽ tối ưu được cả hai điều kiện. Vì vậy, người ta thường ưa thích sử dụng thuật toán phân cụm chuẩn hóa hơn.

## 6.7 *Thực nghiệm*

Ta sử dụng liệu về đặc điểm các loại hoa iris đã được trình ở trong các chương trước để thử nghiệm kết quả phân cụm của thuật toán phân cụm theo phổ.

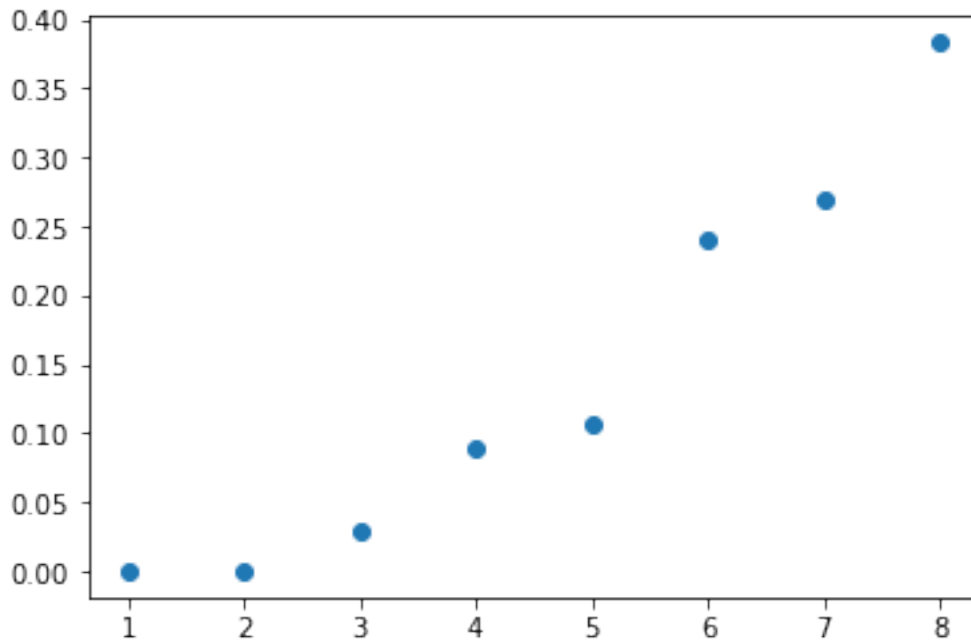
Trong thực nghiệm này, ta sẽ xây dựng đồ thị tương tự bằng đồ thị  $k$  lân cận gần nhất với  $k = 15$  và thuật toán phân cụm phổ chuẩn hóa. Để biết chi tiết hơn về thuật toán và tham số khác, có trực tiếp sử dụng tham khảo phần code.

Sau khi áp dụng xây dựng đồ thị tương tự dựa trên  $k$  lân cận gần nhất, ta thu được ma trận trọng số được mô tả như trong Hình 28a. Ta thấy xây ma trận trọng số tương đối thưa, vì vậy, các thuật toán tìm trị riêng với ma trận thưa sẽ hoạt động tương đối tốt.



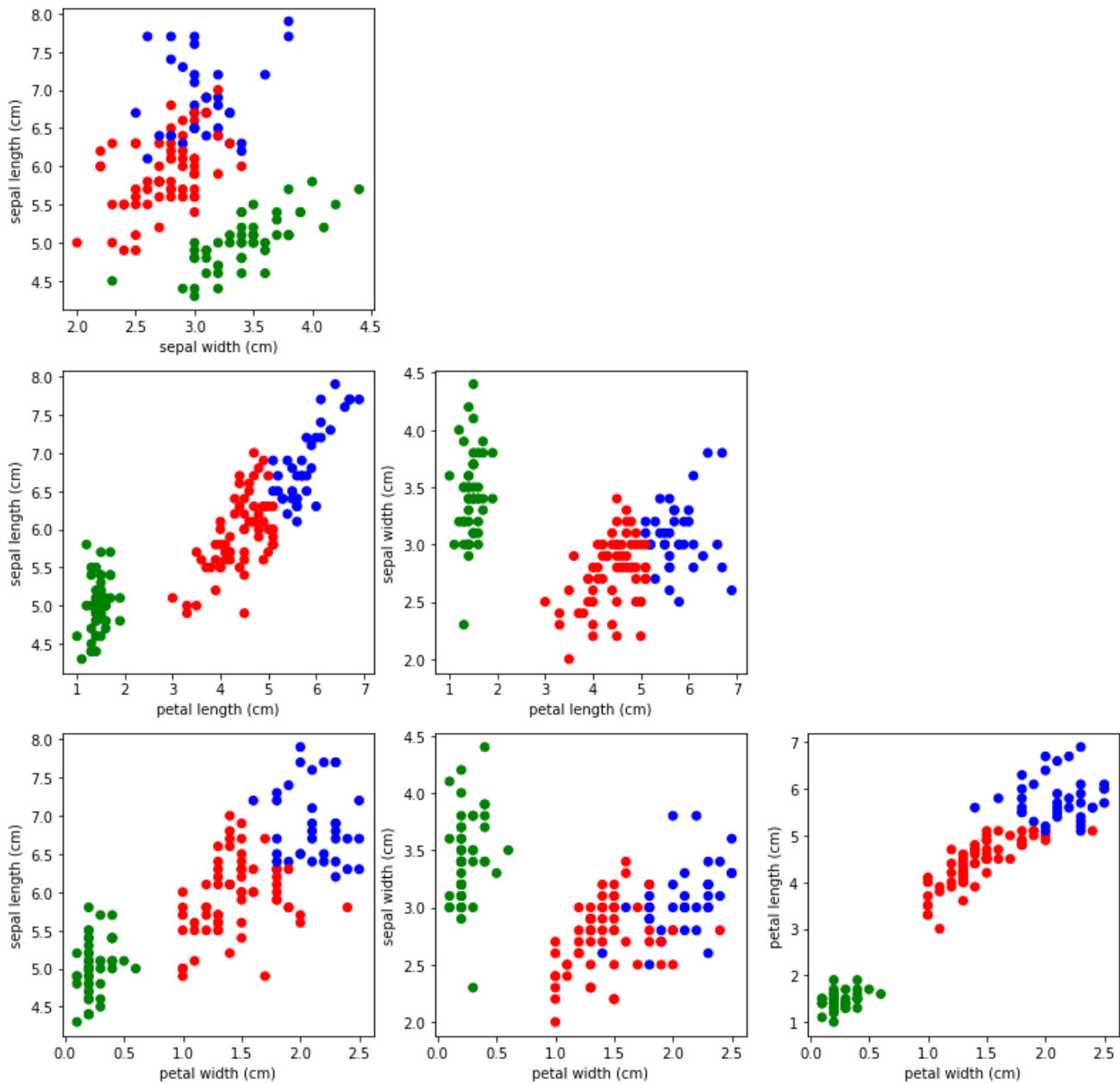
Hình 28: Ma trận trọng số của đồ thị tương tự.

Để chọn số cụm, quan sát Hình 29 ta thấy trị riêng thứ 4 khá lớn so với 3 trị riêng đầu tiên. Vì vậy, ta chọn số cụm cần phân là 3. Điều này cũng đúng với thực tế là có tổng cộng 3 loại hoa.



Hình 29: Các giá trị riêng của ma trận Laplace chuẩn hóa

Sau khi chọn số cụm, ta áp dụng thuật toán phân cụm theo phổ phân thành 3 cụm được mô tả như trong hình 30. Kết quả được mô tả trong các trong không gian hai chiều, với mỗi chiều là từng cặp thuộc tính của hoa, các điểm khác màu nhau tương ứng với các điểm thuộc các cụm khác nhau. Chỉ số silhouette thu được khi áp dụng thuật toán này là 0.5541608580282844. Ta sắp xếp lại các cùng cụm gần nhau thu được ma trận trọng số như trong Hình 28b. Trong Hình 28b ta có thể thấy các điểm trong một cụm độ tương tự nhau khá lớn trong khi các cụm khác nhau thì độ tương tự là tương đối thấp.



Hình 30: Các điểm dữ liệu dữ liệu sau khi được phân cụm với màu khác nhau tương ứng cụm khác nhau.

## 7 Chia tỷ lệ đa chiều (Multidimensional scaling)

### 7.1 Giới thiệu

Mục đích chung của chia tỷ lệ đa chiều là tìm ra cấu hình của các điểm trong không gian, thường là Euclide, trong đó mỗi điểm đại diện cho một trong các đối tượng hoặc cá thể và khoảng cách giữa các cặp điểm trong cấu hình “khớp” nhất có thể có sự khác biệt ban đầu giữa các cặp đối tượng hoặc cá thể.

MDS đề cập đến một tập hợp các kỹ thuật “ordination” có liên quan được sử dụng trong trực quan hóa thông tin, cụ thể là để hiển thị thông tin chứa trong ma trận khoảng cách. Nó là một dạng giảm kích thước phi tuyến tính.

Có thể sắp xếp  $N$  đối tượng trong một hệ tọa độ chiều thấp chỉ bằng cách sử dụng thứ tự hạng của  $N(N-1)/2$  độ tương đồng ban đầu (khoảng cách), chứ không phải độ lớn của chúng khi chỉ sử dụng thông tin thứ tự này để thu được biểu diễn hình học, quá trình này được gọi là *Non-metric multidimensional scaling*. Nếu độ lớn thực tế của độ tương tự (khoảng cách) giữa các điểm ban đầu được sử dụng để thu được biểu diễn hình học theo  $q$  chiều, thì quá trình này được gọi là *Metric multidimensional scaling*.

### 7.2 Thuật toán cơ bản

Cho  $N$  đối tượng, có  $M = N(N-1)/2$  độ tương tự giữa các các cặp đối tượng. Giả sử không có độ tương tự là bằng nhau, độ tương tự được sắp xếp như sau:

$$s_{i_1 k_1} < s_{i_2 k_2} < \dots < s_{i_M k_M} \quad (21)$$

Chúng ta muốn một các biểu diễn  $N$  đối tượng trong không gian  $q$  chiều sao cho khoảng cách giữa các cặp điểm là  $d_{ik}^{(q)}$  thỏa mãn điều kiện:

$$d_{i_1 k_1}^{(q)} > d_{i_2 k_2}^{(q)} > \dots > d_{i_M k_M}^{(q)} \quad (22)$$

Tuy nhiên trong thực tế, với giá trị  $q$  bất kỳ nào đó có khả năng không tìm ra được các biểu diễn các điểm đạt “khớp” hoàn hảo. Vì vậy, [7] đã đề xuất một thước đo về mức độ mà một biểu diễn hình học không thể khớp hoàn hảo. Phép đo này, stress, được định nghĩa như sau:

$$\text{Stress}(q) = \left\{ \frac{\sum_{i < k} \sum_{ik} \left( d_{ik}^{(q)} - \hat{d}_{ik}^{(q)} \right)^2}{\sum_{i < k} \sum_{ik} \left[ d_{ik}^{(q)} \right]^2} \right\}^{1/2} \quad (23)$$

Trong đó  $\hat{d}_{i_1 k_1}^{(q)}$  thỏa mãn:

$$\hat{d}_{i_1 k_1}^{(q)} \geq \hat{d}_{i_2 k_2}^{(q)} \geq \dots > \hat{d}_{i_M k_M}^{(q)} \quad (24)$$

$\hat{d}_{ik}^{(q)}$  được sử dụng để đánh giá sự đơn điệu của  $d_{ik}^{(q)}$ , không nhất thiết phải thỏa mãn tính chất khoảng cách.

Ý tưởng chính là tìm ra một biểu diễn đối tượng thành các điểm trong không gian  $q$  chiều sao cho hàm stress là nhỏ nhất có thể (Kruskal [7]) để đưa ra đánh giá dựa trên kết quả của hàm stress như bảng 8

Bảng 8: Đánh giá MDS theo stress

Stress (%)	Goodness of fit
20	Poor
10	Fair
5	Good
2.5	Excellent
0	Perfect

Để tìm ra một cấu hình  $q$  chiều biểu diễn các đối tượng ban đầu, Kruskal [8] đã đề xuất thuật toán như mô tả trong thuật toán 7:

**Algorithm 7** Thuật toán MDS

---

- 1: **Input:**  $N$  đối tượng cần biểu diễn, số chiều không gian  $q$  mà đối tượng muốn biểu diễn.
- 2: Tính  $N(N-1)/2$  độ tương tự giữa các cặp đối tượng. Sắp xếp chúng theo thứ tự tăng dần
- 3: Khởi tạo một cấu hình biểu diễn các đối tượng thành các điểm  $\{(x_{i1}, \dots, x_{iq})\}_1^N$
- 4: **while** Stress hội tụ **do**
- 5:     Tính khoảng cách giữa các cặp điểm  $d_{ik}$ . Tìm  $\hat{d}_{ik}$

$$\min f(\hat{d}) = \sum_{i < k} (d_{ik} - \hat{d}_{ik})^2 \quad (25)$$

$$\text{vdk} \quad \hat{d}_{i_1 k_1} \leq \dots \leq \hat{d}_{i_N k_N}$$

- 6:     Cập nhật liên tọa độ các điểm khi nào  $\|J\| \approx 0$  :

$$x_{ij} = x_{ij} - \frac{\alpha}{\text{mag}(J)} J_{ij} \quad (26)$$

Trong đó

$$J_{ij} = \frac{\partial \text{stress}(q)}{\partial x_{ij}}, \text{mag}(J) = \sqrt{\frac{\sum_{i < j} J_{ij}^2}{\sum_{i < j} x_{ij}^2}}, \alpha > 0$$

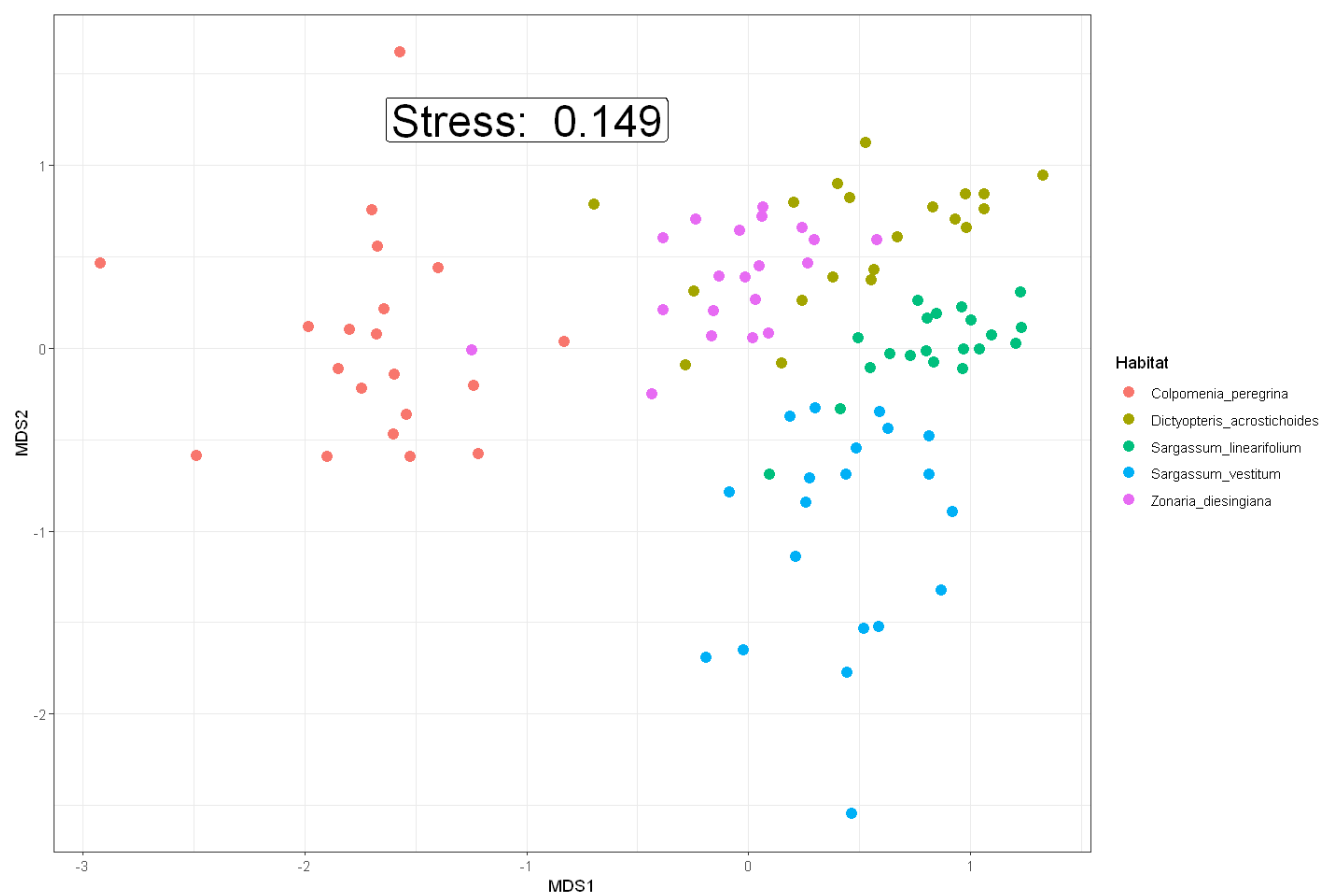
- 7: **end while**
  - 8: **Output:** Tọa độ các điểm trong không gian  $q$  chiều
- 

### 7.3 Thực nghiệm

Ta xem xét một ví dụ nghiên cứu muốn đối chiếu đặc tính kiếm ăn của động vật ăn cỏ biển với năm loài tảo macro [11]. Hai mươi bản sao chép mỗi cá thể của bảy loài tảo macro đã được thu thập từ Cảng Sydney, và sự phong phú của bảy loài giáp xác ăn cỏ được ghi lại từ mỗi lần sao chép. Kết quả đạt được được mô tả như trong hình 31. Ta có thể thấy  $\text{stress} = 0.149$  trong biểu diễn này không quá xấu.

Trong hình 31, các mẫu từ mỗi môi trường có nhiều khả năng giống với các mẫu từ cùng một môi trường hơn là các mẫu từ các sinh cảnh khác (được hình dung dưới dạng các cụm có màu sắc khác nhau). Điều này có nghĩa là thành phần loài của động vật ăn cỏ khác nhau giữa các chuỗi thức ăn của chúng. Nhớ rằng đây không phải là một kiểm định thống kê, đây chỉ đơn là một hình ảnh trực quan giúp ta phân biệt các mẫu.





Hình 31: Kết quả biểu diễn bởi MDS

## Tài liệu

- [1] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. Technical report, Stanford, 2006.
- [2] Mohar. B. The laplacian spectrum of graphs, in: Graph theory, combinatorics, and applications, y. alavi, g. chartrand, o. ollermann, and a. schwenk, eds, 1991.
- [3] Fan RK Chung and Fan Chung Graham. *Spectral graph theory*. Number 92. American Mathematical Soc., 1997.
- [4] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. Rock: A robust clustering algorithm for categorical attributes. *Information Systems*, 25(5):345–366, 2000.
- [5] Lars Hagen and Andrew B Kahng. New spectral methods for ratio cut partitioning and clustering. *IEEE transactions on computer-aided design of integrated circuits and systems*, 11(9):1074–1085, 1992.
- [6] R.A. Johnson and D.W. Wichern. *Applied Multivariate Statistical Analysis: Pearson New International Edition PDF eBook*. Pearson Education, 2013.
- [7] Joseph B Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.
- [8] Joseph B Kruskal. Nonmetric multidimensional scaling: a numerical method. *Psychometrika*, 29(2):115–129, 1964.
- [9] Kevin P. Murphy. *Machine learning : a probabilistic perspective*. MIT Press, Cambridge, Mass. [u.a.], 2013.
- [10] Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856, 2002.
- [11] Alistair GB Poore, Megan J Watson, Rocky de Nys, James K Lowry, and Peter D Steinberg. Patterns of host use among alga-and sponge-associated amphipods. *Marine Ecology Progress Series*, 208:183–196, 2000.
- [12] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000.
- [13] Mechthild Stoer and Frank Wagner. A simple min-cut algorithm. *Journal of the ACM (JACM)*, 44(4):585–591, 1997.

- [14] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch: an efficient data clustering method for very large databases. In *SIGMOD '96*, 1996.