



ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY



VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC
SCHOOL OF APPLIED MATHEMATICS AND INFORMATICS

Phân tích số liệu – MI4020

Clustering, distance methods, and ordination

Sinh viên: <i>Nguyễn Văn Nghiêm</i>	- 20206206
<i>Nguyễn Hoàng Sơn</i>	- 20206165
<i>Đặng Sỹ Tiến</i>	- 20200537
<i>Tạ Duy Hải</i>	- 20206197
<i>Nguyễn Thị Ngọc Lan</i>	- 20185372

Giảng viên: *ThS. Lê Xuân Lý*

Viện: *Toán ứng dụng và Tin học*
Đại học Bách khoa Hà Nội

Hà Nội, Ngày 03 tháng 03 năm 2023

Mục lục

Mở đầu	1
1 Giới thiệu	3
2 Các phép đo tương tự	5
2.1 Khoảng cách và hệ số tương tự cho các cặp quan sát	5
2.2 Sự tương tự và các thước đo liên kết cho các cặp biến	9
2.3 Kết luận về sự tương tự	10
3 Phân cụm phân cấp	12
3.1 Giới thiệu	12
3.1.1 Phương pháp phân cụm theo cấp	12
3.1.2 Biểu diễn kết quả	12
3.2 Các phương pháp phân cụm tổng gộp theo cấp	13
3.2.1 Các loại kết nối	13
3.2.2 Phân cụm theo mức của Ward	16
3.2.3 BIRCH	19
3.3 Kết quả thực nghiệm	21
3.4 Tổng kết	23
4 Phân cụm không phân cấp	24
4.1 Phương pháp K-means	24
4.1.1 Giới thiệu bài toán	24
4.1.2 Thuật toán K-means	24
4.1.3 Nhược điểm của phương pháp K-means	27
4.2 Cải tiến thuật toán K-means và thuật toán K-means++	29
4.2.1 Cải tiến thuật toán K-means	29
4.2.2 Thuật toán K-means++	30
4.2.3 Phương pháp Elbow (Elbow method)	32
4.2.4 Đánh giá Silhouette	33
4.3 Kết quả thực nghiệm	34
5 Phân cụm dựa theo mô hình xác suất thống kê	43
5.1 Mô hình Gausian hỗn hợp	43
5.1.1 Nhắc lại về phân phối Gaussian đa chiều	43
5.1.2 Mô hình Gaussian hỗn hợp	43
5.2 Thuật toán cực đại hóa kỳ vọng EM (Expectation Maximization)	45
5.3 Xác định số phân cụm dựa vào tiêu chuẩn AIC và BIC	48
5.4 Kết quả thực nghiệm	48
5.5 Tổng kết	50
6 Chia tỉ lệ nhiều chiều	51
6.1 Dùng khoảng cách trong chia tỉ lệ nhiều chiều và cách làm cỗ điển	51
6.2 Dùng thứ hạng trong chia tỉ lệ nhiều chiều và cách tiếp cận của Kruskal	58
6.3 Ví dụ	63
6.4 Tổng kết	66

Mở đầu

Bài báo cáo này là kết quả sau một thời gian dài nghiên cứu về chủ đề Clustering trong học phần *Phân tích số liệu - MI4020* và cũng là kết quả đánh dấu sự kết thúc của quá trình học tập và nghiên cứu trong học phần này. 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, những bài học và những câu hỏi ý nghĩa 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.

Trong quá trình làm bài báo cáo này, nhóm đã bầu ra thành viên Nguyễn Văn Nghiêm làm trưởng nhóm, chịu trách nhiệm phân công, đôn thúc và kiểm tra tiến độ làm việc của tất cả các thành viên. Và các thành viên trong nhóm đều đã cố gắng nỗ lực vững kiến thức chủ đề và đều đã hoàn thành hết công việc được giao. Không những vậy, các thành viên đều tích cực tìm hiểu thêm nhiều thuật toán khác không nằm trong giáo trình. Do vậy, đánh giá mức độ đóng góp của tất cả các thành viên đều là **Tích cực**.

Nội dung được phân công cho các thành viên:

1. Giới thiệu: Đặng Sỹ Tiến
2. Các phép đo tương tự: Đặng Sỹ Tiến
3. Phân cụm phân cấp: Nguyễn Hoàng Sơn
4. Phân cụm không phân cấp: Nguyễn Văn Nghiêm
5. Phân cụm dựa theo mô hình xác suất thống kê: Nguyễn Thị Ngọc Lan
6. Chia tỉ lệ nhiều chiều: Tạ Duy Hải

Do thời gian tìm hiểu và nghiên cứu cũng như kiến thức của bản thân 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!

Hà Nội, Ngày 03 tháng 03 năm 2023

Tạ Duy Hải

Nguyễn Thị Ngọc Lan

Nguyễn Văn Nghiêm

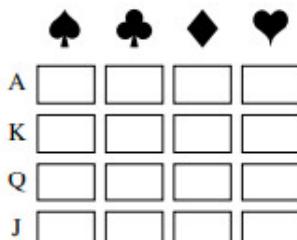
Nguyễn Hoàng Sơn

Đặng Sỹ Tiến

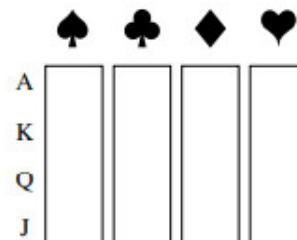
1 Giới thiệu

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.

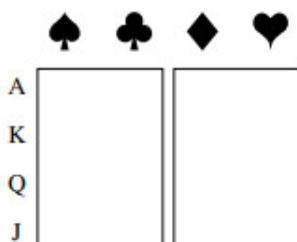
Để minh họa bản chất của khó khăn trong việc xác định một nhóm tự nhiên, hãy xem xét việc sắp xếp 16 lá bài có mặt trong một bộ bài bình thường thành các cụm có các đối tượng giống nhau. Một số nhóm được minh họa trong hình:



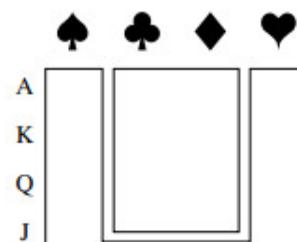
(a) Individual cards



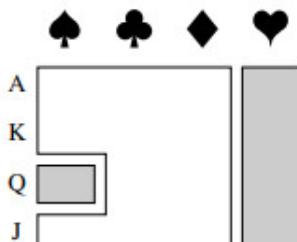
(b) Individual suits



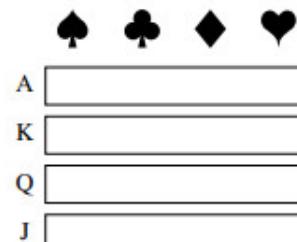
(c) Black and red suits



(d) Major and minor suits (bridge)



(e) Hearts plus queen of spades and other suits (hearts)



(f) Like face cards

Hình 1: Nhóm các lá bài hình người

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 Striling loại 2:

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

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

$n \backslash k$	0	1	2	3	4	5	6	7	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			
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

Hình 2

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),... 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 phép đo tương tự

Hầu hết các biện pháp tương tự với 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 khác biệt" 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 (rồi rạc, liên tục, nhị phân), thang đo lường (danh 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. Nhớ lại rằng khoảng cách Euclidean (đường thẳng) giữa hai quan sát p -chiều:

$$\begin{aligned} x' &= [x_1, x_2, \dots, x_p] \text{ và } y' = [y_1, y_2, \dots, y_p] \\ d(x, y) &= \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_p - y_p)^2} \\ &= \sqrt{(x - y)'(x - y)} \end{aligned} \tag{1}$$

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

$$d(x, y) = \sqrt{(x - y)'A(x - y)} \tag{2}$$

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

$$d(x, y) = \left[\sum_{i=1}^p |x_i - y_i|^m \right]^{1/m} \tag{3}$$

- 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:

$$d(x, y) = \sum_{i=1}^p \frac{|x_i - y_i|}{x_i + y_i} \tag{4}$$

- Hệ số Czekanowski:

$$d(x, y) = 1 - \frac{2 \sum_{i=1}^p \min(x_i, y_i)}{\sum_{i=1}^p (x_i + y_i)} \tag{5}$$

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

$$\begin{aligned} d(P, Q) &= d(Q, P) \\ d(P, Q) &> 0 \text{ nếu } P \neq Q \\ d(P, Q) &= 0 \text{ nếu } P = Q \\ d(P, Q) &\leq d(P, R) + d(R, Q) \text{ (bất đẳng thức tam giác)} \end{aligned}$$

Mặt khác, hầu hết các thuật toán phân cụm sẽ chấp nhận các số đo khoảng cách được ấn định một cách chủ quan, có thể không thỏa mãn một trong các thuộc tính trên, 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 "có" 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: Bảng thể hiện giá trị của hai quan sát

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 có sự khác biệt các quan sát. Từ công thức trên, bình phương khoảng cách giữa các quan sát i và k sẽ là:

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

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: Xét một cặp hai tù nhân cùng sống hoặc cùng không sống (bị tử hình). Nếu cặp tù nhân này cùng sống, nghĩa là cặp 1-1, thì chúng ta sẽ có nhiều khai thác hơn về điểm tương đồng hoặc điểm khác biệt so với khi cặp tù nhân cùng không sống (cặp 0-0).

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:

2 Các phép đo tương tự

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

Bảng 2: Bảng dự phòng

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 khớp
$\frac{a}{p}$	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 của cặp khớp
$\frac{a}{b+c}$	Tỉ lệ các cặp không khớp với cặp 0-0 bị loại trừ

Bảng 3: Bảng cơ sở lý luận

Về mặt ý nghĩa, hệ số 1 thể hiện tỷ lệ giữa các cặp tương đồng so với tổng thể. Hệ số 2 thể hiện tỷ lệ khi tăng tỷ trọng của các cặp tương đồng lên hai lần. Hệ số 3 thể hiện tỷ lệ khi tăng tỷ trọng của các cặp không khớp lên hai lần. Vì cặp 0-0 yếu hơn so với cặp 1-1 nên ta có thể loại bỏ cặp 0-0, khi đó, hệ số 4 thể hiện tỷ lệ cặp 1-1 so với tổng thể. Tương tự với hệ số 5 nhưng tổng thể đã loại bỏ số lượng cặp 0-0. Hệ số 6 được xây dựng từ hệ số 5 nhưng tăng tỷ trọng cặp 1-1 lên hai lần. Hệ số 7 khá giống hệ số 3 nhưng đã loại bỏ cặp 0-0 khỏi tổng thể. Hệ số 8 - hệ số cuối cùng trong bảng thể hiện tỷ lệ giữa số cặp 1-1 và các cặp không khớp.

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 1 và Bảng 2. Khi đó nếu hệ số 1 của bảng 1 lớn hơn hệ số 1 của bảng 2, tương tự hệ số 2 của Bảng 1 cũng lớn hơn hệ số 2 của Bảng 2 và ngược lại, và hệ số 3 sẽ là ít nhất là lớn đối với Bảng 1 cũng như đối với Bảng 2. 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	140 lb	Xanh lá	Đen	Phải	Nữ
Cá nhân 2	73 in	185 lb	Nâu	Nâu	Phải	Nam
Cá nhân 3	67 in	165 lb	Xanh dương	Đen	Phải	Nam
Cá nhân 4	64 in	120 lb	Nâu	Nâu	Phải	Nữ
Cá nhân 5	76 in	210 lb	Nâu	Nâu	Trái	Nam

Bảng 4: Bảng thể hiện đặc điểm của các cá nhân

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

$$\begin{aligned} X_1 &= \begin{cases} 1 \text{ chiều cao } \geq 72\text{in} \\ 0 \text{ chiều cao } < 72\text{in} \end{cases} & X_4 &= \begin{cases} 1 \text{ tóc đen} \\ 0 \text{ tóc nâu} \end{cases} \\ X_2 &= \begin{cases} 1 \text{ cân nặng } \geq 150\text{lb} \\ 0 \text{ cân nặng } < 150\text{lb} \end{cases} & X_5 &= \begin{cases} 1 \text{ tay phải} \\ 0 \text{ tay trái} \end{cases} \\ X_3 &= \begin{cases} 1 \text{ mắt nâu} \\ 0 \text{ mắt khác} \end{cases} & X_6 &= \begin{cases} 1 \text{ nữ} \\ 0 \text{ nam} \end{cases} \end{aligned}$$

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_6	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		Totals
		1	0	
Cá nhân 1	1	1	2	3
	0	3	0	3
Totals		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:

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

2 Các phép đo tương tự

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:

		Individual				
		1	2	3	4	5
Individual	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	$\left(\frac{5}{6}\right)$	$\frac{2}{6}$	$\frac{2}{6}$	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 3 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		Totals
		1	0	
Biến i	1	a	b	a+b
	0	c	d	c+d
Totals		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 $r^2 = \frac{x^2}{n}$ để kiểm tra tính độc lập của hai biến phân loại. Đôi 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 [1]: (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 "mối liên kết" chặt chẽ hơn. Tiếng Hungary và tiếng Phần Lan đường như đứng riêng, và tiếng Ba Lan có một số đặc điểm của các ngôn ngữ trong mỗi nhóm con lớn hơn.

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	neljä
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	ochos	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: Bảng danh sách các từ đếm số từ 1-10 của các nước Châu Âu

2 Các phép đo tương tự

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.

	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: Bảng tần số ngôn ngữ

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 ẩn 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 ẩn tượng trực quan về các biện pháp tương tự hoặc khoảng cách để tạo thành nhóm. Ở các phần sau, chúng ta sẽ thảo luận về các phương pháp ít chủ quan hơn để phân cụm.

3 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

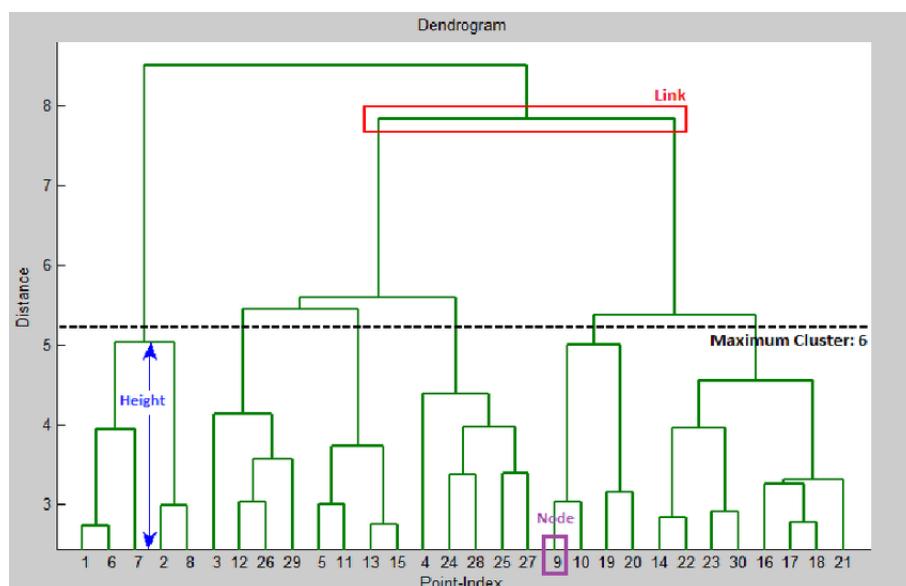
- Agglomerative clustering: Bắt đầu từ các cụm nhỏ nhất là mỗi cụm gồm 1 phần tử và gộp dần các phần tử lại cho đến khi thành 1 cụm duy nhất hay đến 1 ngưỡng cho trước.
- Divisive clustering: Bắt đầu từ cụm lớn nhất là 1 cụm chứa tất cả các phần tử và tách dần các cụm cho đến khi mỗi cụm còn 1 phần tử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

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.



Hình 3: Biểu đồ dendrogram

3.2 Các phương pháp phân cụm tổng hợp theo cấp

Sau đây, ta sẽ trình bày các loại thuật toán phân cụm theo cách tổng hợp 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 method

3.2.1 Các loại kết nối

Với phương pháp phân cụm tổng hợp 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:

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

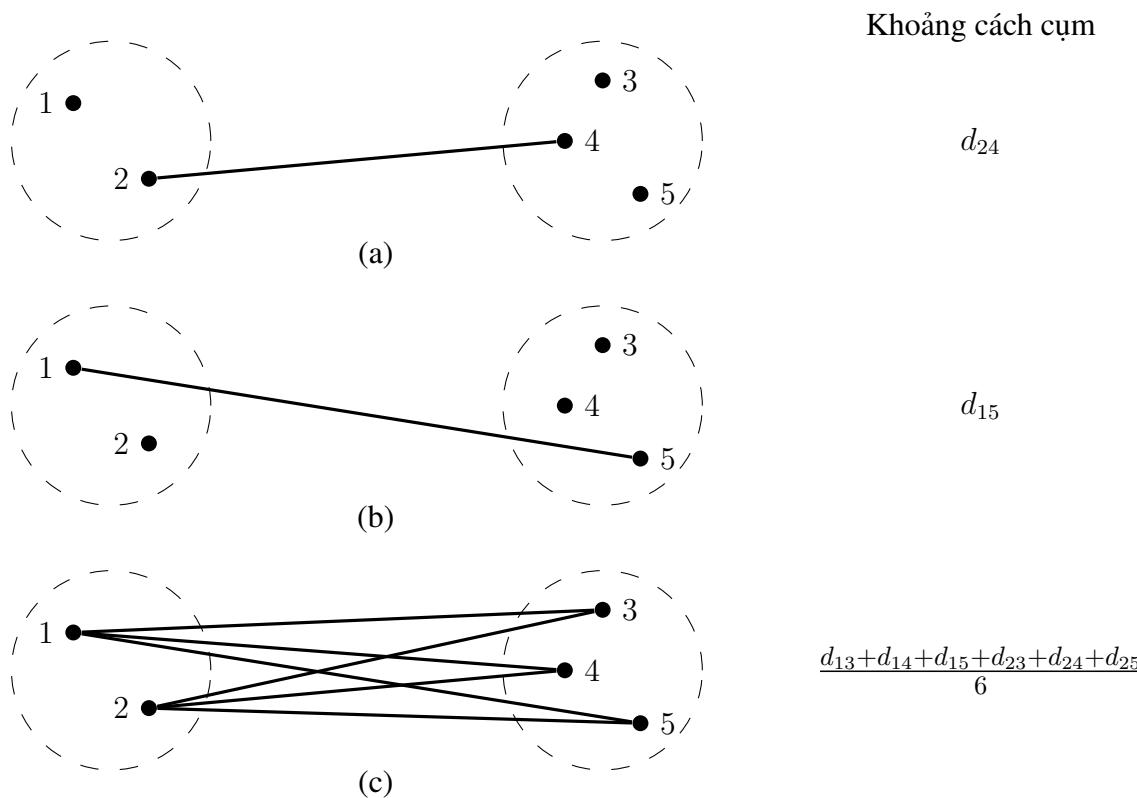
Kết nối toàn phần:

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

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



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.
-

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:

$$d_{(UV)W} = \min\{d_{UW}, d_{VW}\} \quad (6)$$

Để làm rõ hơn ta cùng xét 1 ví dụ.

3 Phân cụm phân cấp

Ví dụ

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

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

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:

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

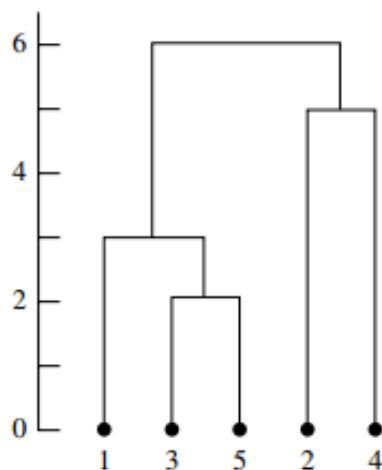
Làm tương tự, gộp 2 và 4 ta được:

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

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

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

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.



Hình 4: Biểu đồ dendrogram cho kết nối đơn

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), ...

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:

$$d_{(UV)W} = \max\{d_{UW}, d_{VW}\} \quad (7)$$

Kết nối trung bình:

$$d_{(UV)W} = \frac{|U||W|d(U, W) + |V||W|d(V, W)}{|UV||W|} \quad (8)$$

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:

$$ESS(X) = \sum_{x \in X} (x - \bar{x})'(x - \bar{x}) \quad (9)$$

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:

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

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

Cài đặt phương pháp của Ward [2]

Việc thực hiện phương pháp của Ward hoàn toàn tương tự 3 loại kết nối đã nêu:

$$d(UV, W) = \frac{|U| + |W|}{|U| + |V| + |W|} d(U, W) + \frac{|V| + |W|}{|U| + |V| + |W|} d(V, W) - \frac{|W|}{|U| + |V| + |W|} d(U, V) \quad (10)$$

3 Phân cụm phân cấp

Thuật toán Lance–Williams [3]

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:

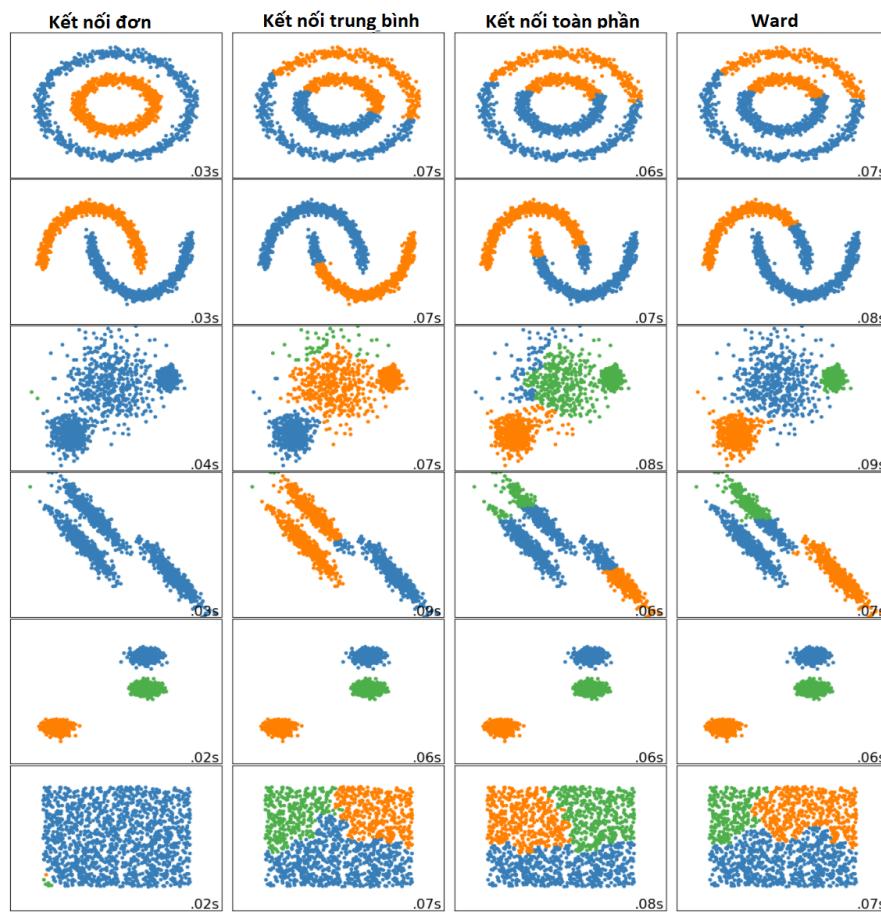
$$d(UV, W) = \alpha_i d(U, W) + \alpha_j d(V, W) + \beta d(U, V) + \gamma |d(U, W) - d(V, W)| \quad (11)$$

Phương pháp	α_i	α_j	β	γ
Kết nối đơn	0.5	0.5	0	-0.5
Kết nối toàn phần	0.5	0.5	0	0.5
Trung bình nhóm	$\frac{ U }{ U + V }$	$\frac{ V }{ U + V }$	0	0
Trung bình nhóm trọng số	0.5	0.5	0	0
Trọng tâm	$\frac{ U }{ U + V }$	$\frac{ V }{ U + V }$	$\frac{- U . V }{(U + V)^2}$	0
Ward	$\frac{ U + W }{ U + V + W }$	$\frac{ V + W }{ U + V + W }$	$\frac{- W }{ U + V + W }$	0

Bảng 7: Bảng công thức cho một số phương pháp

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 5: 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 Phân cụm phân cấp

3.2.3 BIRCH

BIRCH [4] (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 - được gọi là cây thuộc tính cụm (Clustering Feature Tree - CF Tree).

BIRCH chỉ lưu bộ ba thông tin thống kê đặc trưng (N, LS, SS) - được gọi là các đặc trưng của cụm (Clustering Feature - CF).

Một số khái niệm

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

- Bán kính: $R = (\frac{1}{N} \sum_{x \in X} (x - m_x)^2)^{\frac{1}{2}}$
- Đường kính: $D = (\frac{1}{N(N-1)} \sum_{x,y \in X} (x - y)^2)^{\frac{1}{2}}$
- Momen bậc 1: $LS = \sum_{x \in X} x$
- Momen bậc 2: $SS = \sum_{x \in X} x^2$
- Khoảng cách trung bình giữa hai cụm:

$$D_1 = \left(\frac{\sum_{i=1}^{N_1} \sum_{l=N_1+1}^{N_1+N_2} (X_i - X_l)^2}{N_1 N_2} \right)^{\frac{1}{2}}$$

- Lượng thông tin mất mát giữa hai cụm:

$$D_2 = \sum_{k=1}^{N_1+N_2} (X_k - \frac{\sum_{l=1}^{N_1+N_2} X_l}{N_1 + N_2})^2 - \sum_{i=1}^{N_1} (X_i - \frac{\sum_{l=1}^{N_1} X_l}{N_1})^2 - \sum_{j=N_1+1}^{N_1+N_2} (X_j - \frac{\sum_{l=N_1+1}^{N_1+N_2} X_l}{N_2})^2$$

Thuộc tính cụm:

Với mỗi cụm ta sẽ lưu 3 thuộc tính cụm (CF) tương ứng là số lượng đối tượng trong cụm, momen bậc 1, 2:

$$CF_X = (N_X, LS_X, SS_X) \quad (12)$$

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

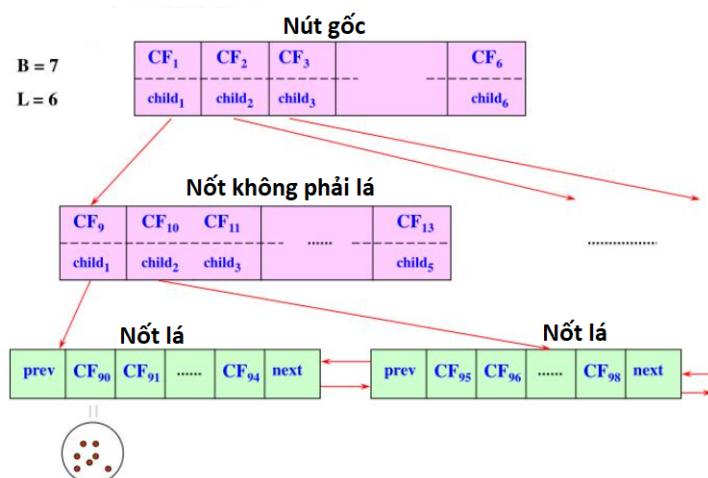
$$CF_{12} = CF_1 + CF_2 = (N_1 + N_2, LS_1 + LS_2, SS_1 + SS_2)$$

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 (CFT) được đặc trưng bởi hai yếu tố sau:

- *Hệ số phân nhánh*: Xác định số nút con tối đa \mathbf{B} của nút gốc hoặc nút trong và số mục tối đa \mathbf{L} của một nút lá.
- *Ngưỡng*: Khoảng cách tối đa \mathbf{T} giữa các cặp đối tượng trong một nút lá. Khoảng cách này thường được gọi là đường kính hoặc bán kính của cụm con được lưu tại nút lá.



Hình 6: Cây thuộc tính cụm

Algorithm 2 Thuật toán BIRCH

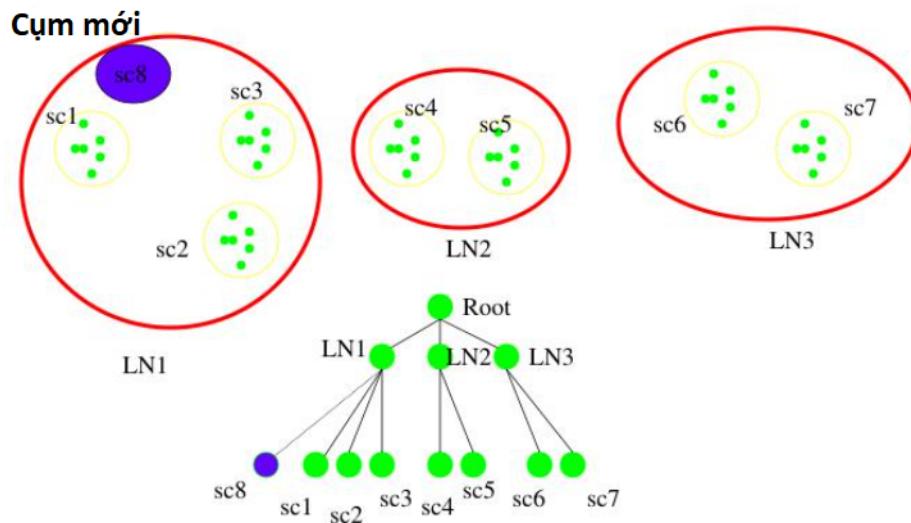
- 1: Xây dựng cây thuộc tính cụm (CFT).
- 2: (Tùy chọn) Hiệu chỉnh ngưỡng, xây dựng lại CFT (không đủ bộ nhớ, ...)
- 3: Phân cụm toàn cục.
- 4: (Tùy chọn) Tinh chỉnh các cụm.

Trong **Bước 1**, các đối tượng dữ liệu lần lượt được chèn vào CFT. Các đối tượng được chèn vào nút lá gần nó nhất (về khoảng cách). Nếu đường kính của cụm con này lớn hơn *ngưỡng T* hoặc vi phạm điều kiện về *hệ số phân nhánh* thì sẽ xảy ra tách nút cho đến khi CFT thỏa mãn cả hai điều kiện được đặt ra.

Quá trình thêm và tách nút được minh họa bởi ví dụ sau.

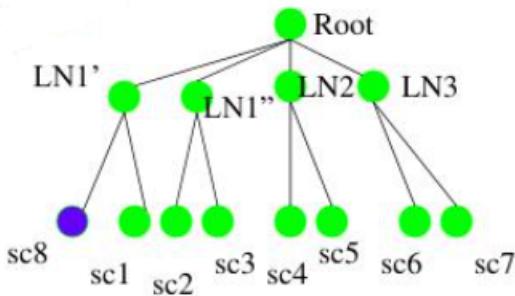
Ví dụ

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

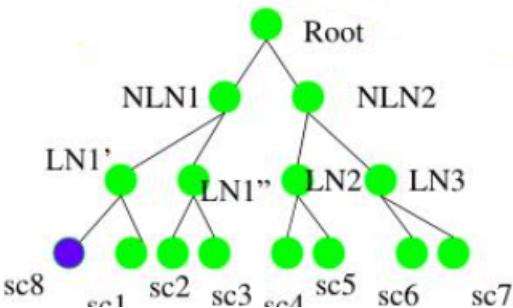


Do số phần tử tối đa là 3 nên ta tách nốt lá LN1

3 Phân cụm phân cấp



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



Trong quá trình tách nốt, các mục ở trong các nốt lá hiện tại sẽ được xem xét để tách cùng và thêm vào nốt mới với mục được tách nếu việc thêm vào không vi phạm điều kiện được đặt ra ban đầu. Ngoài ra, việc lựa chọn mục nào được tách cùng cũng phải đảm bảo tối ưu các thuộc tính của cụm.

Bước 2 đóng vai trò bù đắp giữa **Bước 1** và **Bước 3**. Trong bước này, CFT sẽ được quét lại để xây dựng một CFT nhỏ hơn, loại bỏ các điểm *outlier* và nhóm các cụm con nhỏ thành một cụm lớn hơn để giảm bớt bộ nhớ hoặc để phù hợp với thuật toán phân cụm ta áp dụng ở **Bước 3**.

Trong **Bước 3**, chúng ta sẽ áp dụng thuật toán phân cụm cho toàn bộ các mục ở trong các nốt lá khác nhau. Khoảng cách được áp dụng cho thuật toán phân cụm thường là khoảng cách **D1** hoặc **D2**.

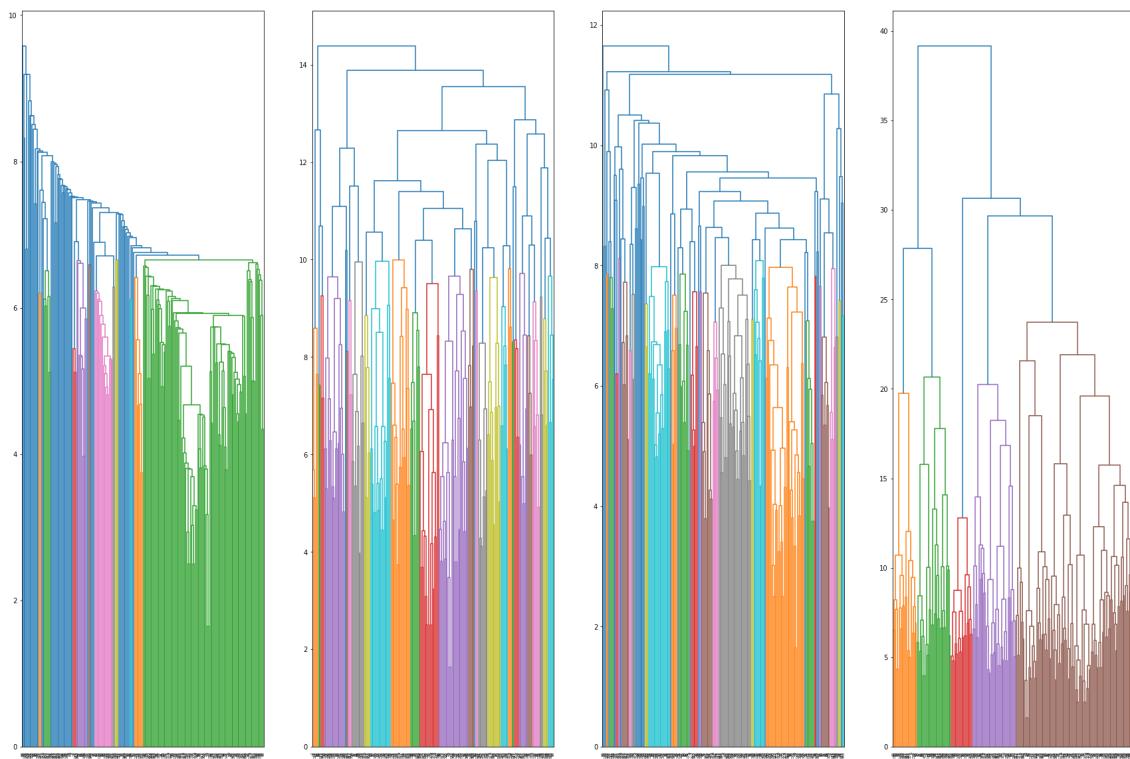
Bước 4 có vai trò gắn nhãn và phân phối lại các điểm cho các cụm dựa vào trọng tâm của nó, do có thể trong quá trình thực hiện **Bước 3** xảy ra một số sai sót dẫn đến phân cụm nhầm, có hai bản sao của điểm tại hai cụm khác nhau,....

Nhận xét

Nhìn chung, BIRCH hoạt động khá tốt đối với tập dữ liệu lớn. Tuy nhiên, hạn chế của nó là chỉ hoạt động được trên lớp bài toán phân cụm mà dữ liệu ở dạng số.

3.3 Kết quả thực nghiệm

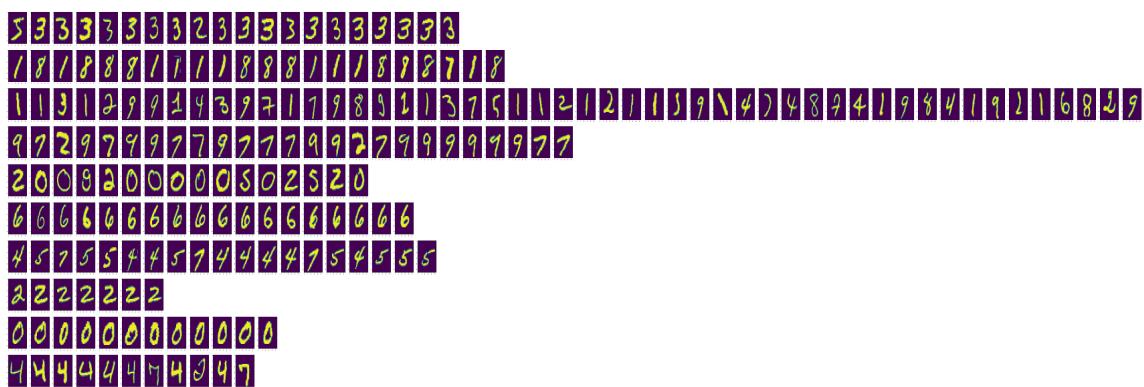
Thực hiện phân cụm trên bộ dữ liệu ảnh số MNITS. Trong ví dụ này em chỉ chọn 200 dữ liệu đầu để phân cụm.



Hình 7: Biểu đồ dendrogram của kết quả

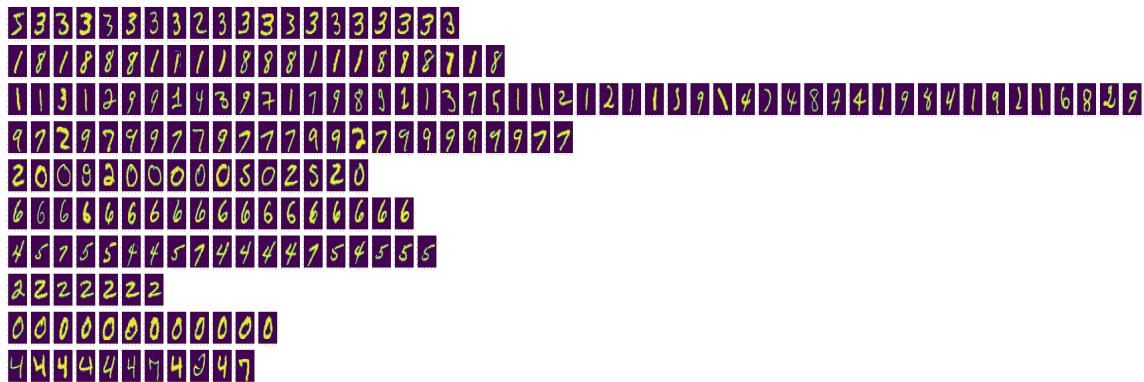
Từ trái sang phải lần lượt là kết nối đơn, kết nối toàn phần, kết nối trung bình và phương pháp Ward.

Ta có thể thấy sự khác biệt rõ ràng nhất về sự chênh lệch số lượng phần tử giữa các cụm thể hiện trong kết nối đơn. Trong kết nối trung bình và kết nối toàn phần thì chênh lệch này cũng khá lớn. Phương pháp Ward cho ta một sự phân bố đều hơn. Dưới đây là kết quả phân cụm của phương pháp Ward và phương pháp Birch.



Hình 8: Phương pháp Ward

3 Phân cụm phân cấp



Hình 9: Phương pháp Birch

Ta thu được kết quả từ hai phương pháp trong tường hợp này là gần như giống hệt nhau. Có thể nói thuật toán cũng đã hoạt động khá hiệu quả với ví dụ này, trừ một vài trường hợp cá biệt.

3.4 Tổng kết

Có rất nhiều phương pháp tổng hợp 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.

4 Phân cụm không phân cấp

Thuật toán phân cụm không phân cấp được thiết kế để phân cụm bộ dữ liệu ban đầu thành K cụm và từ K cụm đầu ra tiến hành quản lý dữ liệu. K có thể biết trước hoặc được xác định như một phần trong thủ tục phân cụm. 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.

4.1 Phương pháp K-means

4.1.1 Giới thiệu bài toán

Phương pháp K-means được giới thiệu lần đầu bởi nhà toán học Lloyd năm 1957 và được James MacQueen sử dụng lần đầu tiên vào năm 1967. Mặc dù đã tồn tại từ lâu nhưng đến thời điểm hiện tại, K-means vẫn là một trong những phương pháp phổ biến nhất.

K-means là phương pháp dựa trên phân hoạch. Phân cụm bằng cách chia không gian ra nhiều miền khác nhau và không giao nhau. Giải thuật K-means phân chia tệp dữ liệu thành k cụm và mỗi cụm có một điểm trung tâm được gọi là **centroid**.

4.1.2 Thuật toán K-means

Quá trình phân cụm K-means được thực hiện qua 3 bước chính:

1. Chọn ngẫu nhiên k điểm làm k trung tâm ban đầu (initial centroids)
2. Phân các điểm dữ liệu vào cụm có điểm centroid gần với nó nhất. Tính toán lại điểm centroid cho các cụm.
3. Lặp lại bước 2 cho đến khi thỏa mãn *điều kiện hội tụ*.

Điều kiện hội tụ:

- Không có (không đáng kể) việc phân lại các điểm dữ liệu vào các cụm khác
- Không có (không đáng kể) thay đổi các điểm centroid
- Giảm không đáng kể về tổng lỗi phân cụm:

$$SSE = \sum_{i=1}^k \sum_{x \in C_i} d(x, c_i)^2 \quad (13)$$

- C_i : cụm thứ i
- c_i : điểm centroid của cụm C_i
- $d(x, c_i)$: khoảng cách của điểm x và điểm trung tâm c_i

Trong giai đoạn chọn các điểm trung tâm ban đầu, chúng ta có thể lựa chọn ngẫu nhiên các điểm bất kì tuy nhiên để tránh việc các điểm ngẫu nhiên đó cách xa so với bộ dữ liệu dẫn đến tốc độ hội tụ chậm hơn, do đó chúng ta nên chọn ngẫu nhiên các điểm nằm trong bộ dữ liệu ban đầu làm các điểm trung tâm.

Giải thuật K-means có thể linh hoạt lựa chọn các phép đo khoảng cách để tính (Euclidean, Minkowski ...), và thường phổ biến là khoảng cách Euclidean.

4 Phân cụm không phân cấp

Thuật toán K-means:

Algorithm 3 K-means(χ , k)

- 1: Chọn ngẫu nhiên k điểm centroid để làm các điểm trung tâm ban đầu $C = \{c_1, c_2, \dots, c_k\}$
- 2: **while** not CONVERGENCE **do**
- 3: **for each** $x \in \chi$ **do**
- 4: Tính khoảng cách từ x đến các điểm centroid
- 5: Gán x vào cụm có điểm centroid gần nhất
- 6: **end for**
- 7: **for each** $C_i \in C$ **do**
- 8: Tính lại các điểm centroid $c_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$
- 9: **end for**
- 10: **end while**

Ví dụ 1

Sử dụng K-means phân cụm 7 quan sát sau thành 2 cụm:

Individual	x_1	x_2
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

Đầu tiên ta cần lựa chọn phép đo khoảng cách, trong ví dụ này lựa chọn khoảng cách Euclidean. Chọn 2 tâm cụm bắt đầu là 2 quan sát 1 và 4

	Individual	Centroid
Cluster 1	1	(1.0, 1.0)
Cluster 2	4	(5.0, 7.0)

Các quan sát còn lại được phân bổ cho cụm gần nhất.

Individual	Distance to centroid of cluster 1	Distance to centroid of cluster 2
1	0.0	7.2
2	1.1	6.1
3	3.6	3.6
4	7.2	0.0
5	4.7	2.5
6	5.3	2.1
7	4.3	2.9

Quan sát 1, 2 gần với cụm 1 nhất nên được phân vào cụm 1.

Một điểm chú ý là quan sát thứ 3 có khoảng cách cách đều so với cả 2 cụm nên việc phân bổ nó vào cụm nào sẽ do chiến lược của người dùng quyết định. Trong trường hợp này, quan sát thứ 3 được phân vào cụm 1. Các quan sát còn lại được phân vào cụm 2.

Sau khi phân bổ các điểm còn lại vào 2 cụm thu được kết quả:

	Individual	Centroid
Cluster 1	1, 2, 3	(1.8, 2.3)
Cluster 2	4, 5, 6, 7	(4.1, 5.4)

Tiếp tục bước lặp, tính toán khoảng cách giữa các quan sát với 2 tâm cụm:

Individual	Distance to centroid of cluster 1	Distance to centroid of cluster 2
1	1.5	5.4
2	0.4	4.3
3	2.1	1.8
4	5.7	1.8
5	3.2	0.7
6	3.8	0.6
7	2.8	1.1

Đến đây nhận thấy rằng chỉ có quan sát thứ 3 gần với cụm 2 nhất nhưng lại thuộc cụm 1. Do đó đưa quan sát thứ 3 sang cụm 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:

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

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

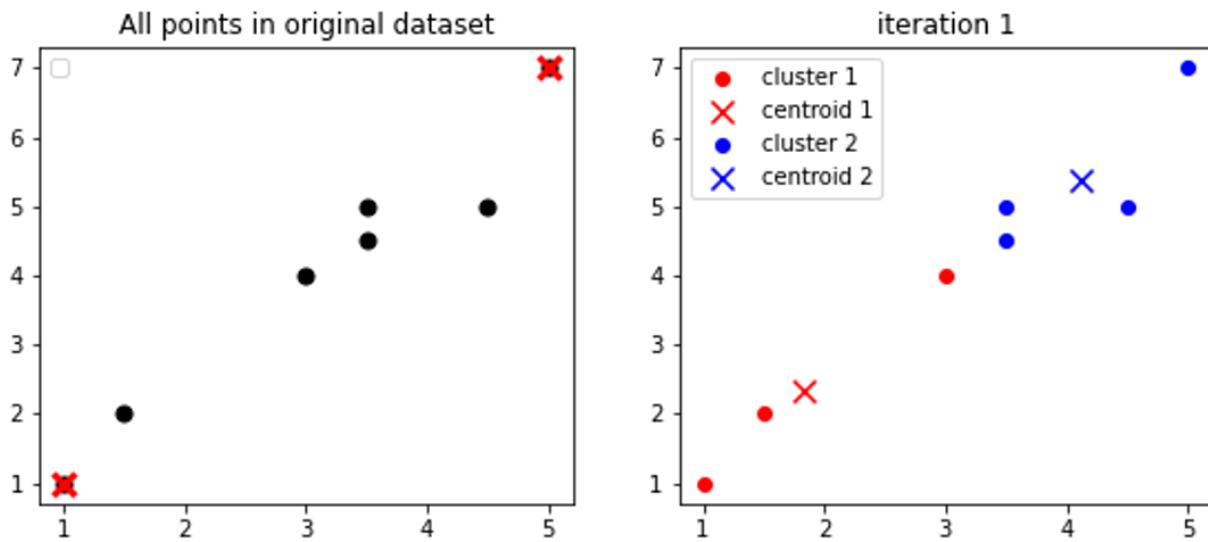
Trong đó n là số lượng điểm trong cụm trước khi cập nhật với điểm trung tâm là $\bar{x}' = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_p)$. Tính toán lại tâm cụm thu được kết quả:

	Individual	Centroid
Cluster 1	1, 2	(1.3, 1.5)
Cluster 2	3, 4, 5, 6, 7	(3.9, 5.1)

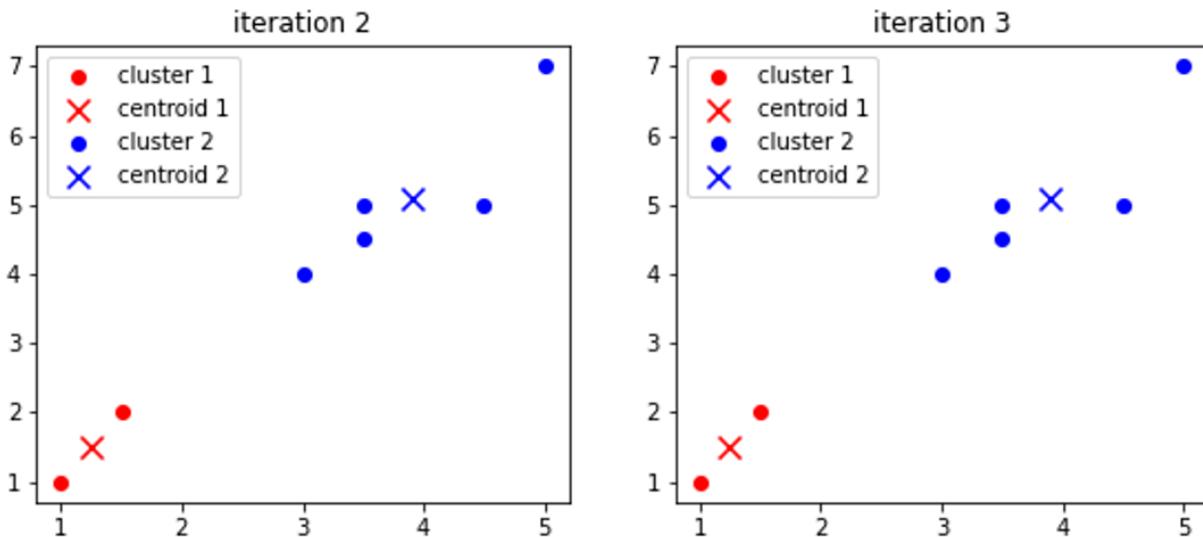
Quá trình lặp lại này vẫn sẽ được tiếp tục nhưng kết quả không thay đổi do vậy kết quả 2 cụm trên là kết quả cuối cùng của phân cụm K-means cho 7 quan sát trên.

4 Phân cụm không phân cấp

Trực quan hóa lại quá trình phân cụm trên:



Hình 10: Quá trình phân cụm K-means



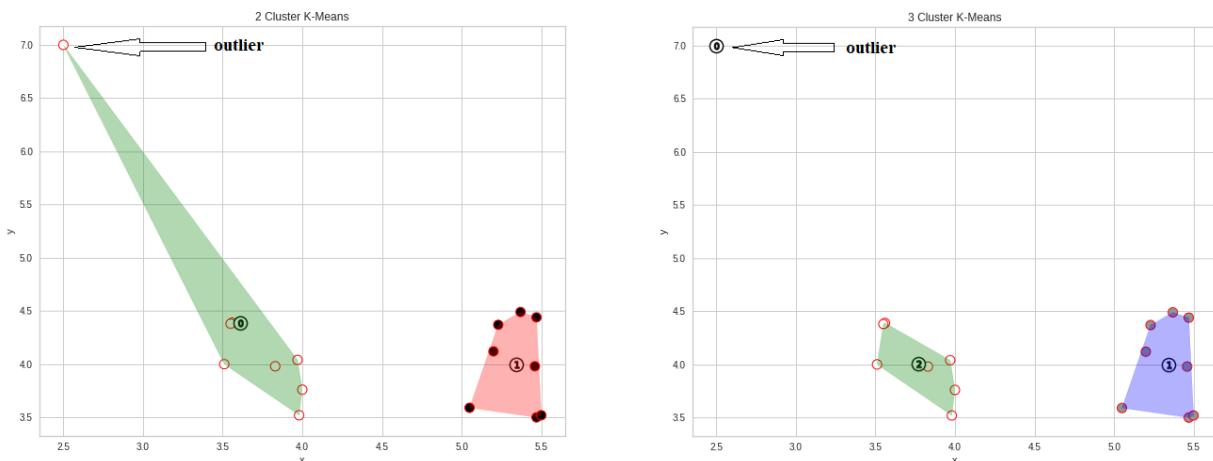
Hình 11: Quá trình phân cụm K-means

4.1.3 Nhược điểm của phương pháp K-means

K-means là một thuật toán phân cụm đơn giản, dễ hiểu mà lại hiệu quả tuy nhiên nhược điểm là ta cần phải xác định trước số cụm k trước khi tiến hành chạy thuật toán. Ngoài ra, K-means cũng có 1 số nhược điểm điển hình sau:

Ảnh hưởng của điểm ngoại lai:

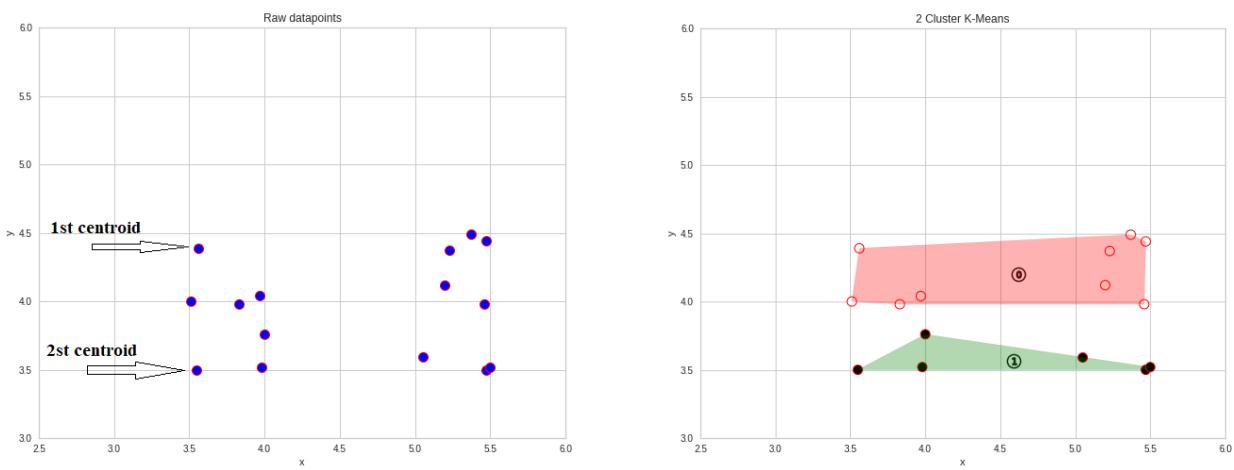
K-means rất nhạy cảm với các ngoại lai, tức là, các ngoại lai có thể ảnh hưởng đáng kể về kết quả phân cụm.



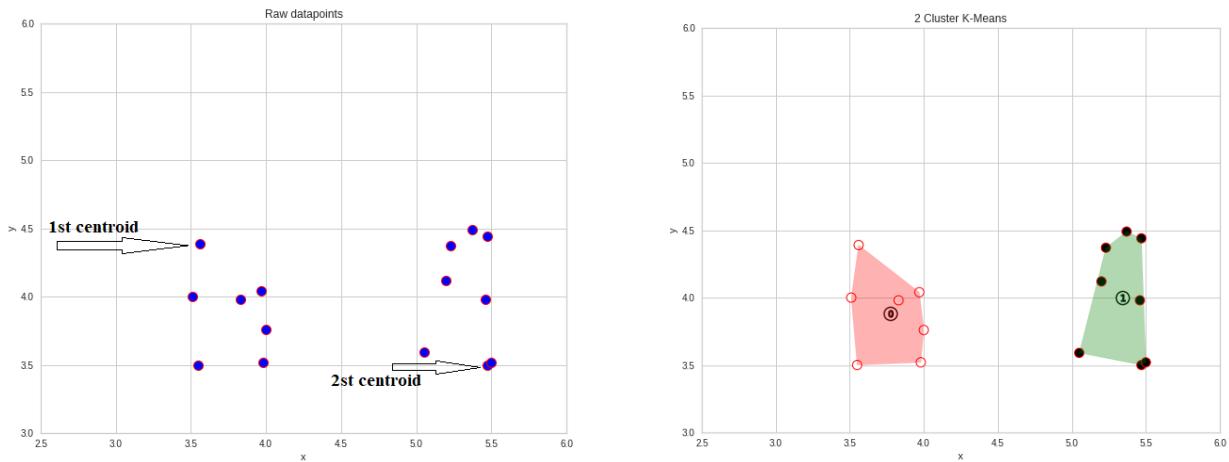
Hình 12: Undesirable clusters and Ideals clusters

Ảnh hưởng của việc khởi tạo các điểm trung tâm ban đầu:

Chất lượng của K-means phụ thuộc nhiều vào các trung tâm ban đầu. Ví dụ minh họa trường hợp chọn các điểm trung tâm ban đầu khác nhau dẫn đến kết quả khác nhau của cùng 1 tập dữ liệu:



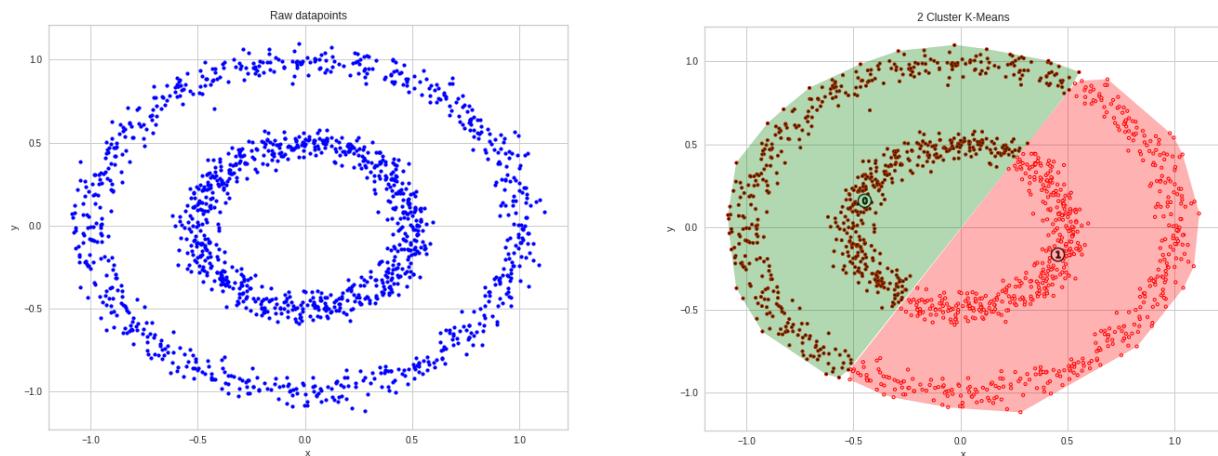
Hình 13: Random selection of seeds for case one



Hình 14: Random selection of seeds for case two

K-means khó khăn trong việc phân cụm các cụm cong (curved clusters):

Khi sử dụng khoảng cách Euclidean, K-means khó khăn phát hiện các cụm phi hình cầu (non-spherical clusters).



Hình 15: How to deal with those cases?

Đánh giá thuật toán:

- **Ưu điểm:**
 - Đơn giản, dễ hiểu, hiệu quả
 - Dễ cài đặt
- **Nhược điểm:**
 - Phải xác định được giá trị k
 - Thiết lập các điểm trung tâm khác nhau cho các cụm sẽ dẫn tới những kết quả phân cụm khác
 - Giải thuật k-means nhạy cảm (gặp lỗi) với các điểm ngoại lai (outliers)
 - K-means khó khăn trong việc phân cụm các cụm cong (curved clusters)
 - 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, do vậy ta nên chuẩn hóa dữ liệu trước khi tiến hành chạy thuật toán

4.2 Cải tiến thuật toán K-means và thuật toán K-means++

4.2.1 Cải tiến thuật toán K-means

Vấn đề ngoại lai:

Các điểm ngoại lai sẽ dẫn đến kết quả phân cụm không tốt và để giải quyết vấn đề này có thể sử dụng 2 giải pháp:

Giải pháp 1: Loại bỏ các điểm ngoại lai

Để loại bỏ các điểm ngoại lai sử dụng quy tắc 1.5IQR:

- Tứ phân vị thứ nhất Q1: Phân vị mức 25%
- Tứ phân vị thứ hai Q2: Phân vị mức 50%
- Tứ phân vị thứ ba Q3: Phân vị mức 75%
- Khoảng cách tứ phân vị: $IQR = Q3 - Q1$

Quy tắc 1.5IQR: Giá trị x được gọi là điểm ngoại lai nếu $x \notin [Q1 - 1.5IQR, Q3 + 1.5IQR]$
Ngoài ra có thể sử dụng quy tắc 1.7IQR hoặc Z-score để xác định điểm ngoại lai

Giải pháp 2: Thực hiện lấy mẫu ngẫu nhiên (a random sampling)

Thay vì phân cụm tất cả dữ liệu, chúng ta lấy một mẫu ngẫu nhiên S từ toàn bộ dữ liệu.

- S sẽ được sử dụng để phân thành k cụm. Do chỉ lấy một tập con nên khả năng các điểm ngoại lai được chọn là thấp.
- Gán các điểm còn lại của tập dữ liệu vào các cụm tùy theo đánh giá về khoảng cách (hoặc độ tương tự)

Vấn đề chọn điểm trung tâm ban đầu:

Lựa chọn các điểm trung tâm khác nhau có thể dẫn đến các kết quả, tốc độ hội tụ khác nhau. Do đó để giải quyết vấn đề này có 2 phương pháp được nêu ra:

Giải pháp 1: Thực hiện giải thuật K-means nhiều lần, mỗi lần lựa chọn tập các điểm trung tâm ban đầu khác nhau. Sau đó phân tích đưa ra kết quả các cụm tốt nhất.

Tuy nhiên thực hiện giải pháp này có thể tốn nhiều thời gian và cần có sự phân tích của con người.

Giải pháp 2: Sử dụng thuật toán K-means++ để thiết lập các điểm trung tâm cho các cụm.

Mục tiêu của thuật toán K-means++ là lựa chọn các điểm trung tâm ban đầu càng xa càng tốt, việc đó sẽ đem lại hiệu quả tối ưu trong quá trình phân cụm.

4.2.2 Thuật toán K-means++

Thuật toán K-means++ lần đầu tiên được giới thiệu trong bài báo "*k-means++: the advantages of careful seeding*" [5] của David Arthur và Sergei Vassilvitskii. Trong phương pháp K-means, các trung tâm ban đầu được chọn ngẫu nhiên từ các điểm dữ liệu, điều này có thể dẫn đến sự hội tụ kém và kết quả cuối cùng dưới mức tối ưu. K-means++ giải quyết vấn đề này bằng cách sử dụng một phương pháp cẩn thận hơn để chọn các trung tâm ban đầu.

Thuật toán bắt đầu bằng cách chọn một centroid ngẫu nhiên từ các điểm dữ liệu. Sau đó, đối với mỗi trung tâm tiếp theo, thuật toán chọn một điểm dữ liệu cách xa bất kỳ centroid hiện có, với xác suất tỷ lệ thuận với bình phương khoảng cách của nó với tâm gần nhất. Điều này đảm bảo rằng các trung tâm ban đầu được phân tách tốt, dẫn đến sự hội tụ nhanh hơn và kết quả cuối cùng tốt hơn.

Thuật toán K-means++ đã được chứng minh là cải thiện đáng kể chất lượng của phân cụm cuối cùng, cũng như giảm số lần lặp cần thiết để hội tụ. Nó đã trở thành một lựa chọn phổ biến để khởi tạo K-means và được sử dụng rộng rãi trong thực tế.

Với $D(x)$ được định nghĩa là khoảng cách ngắn nhất của quan sát x với các tâm cụm.

Thuật toán K-means++ được xây dựng như sau:

4 Phân cụm không phân cấp

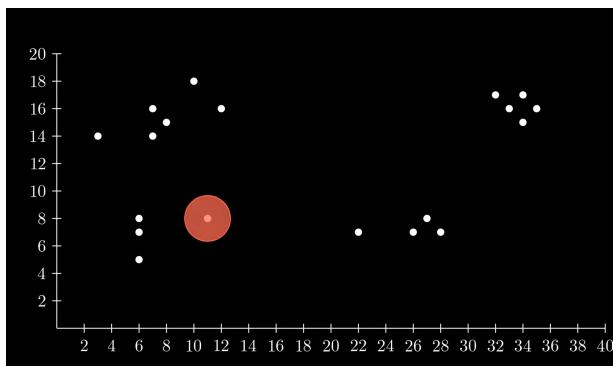
Thuật toán K-means++:

Algorithm 4 K-means $\text{++}(\chi, k)$

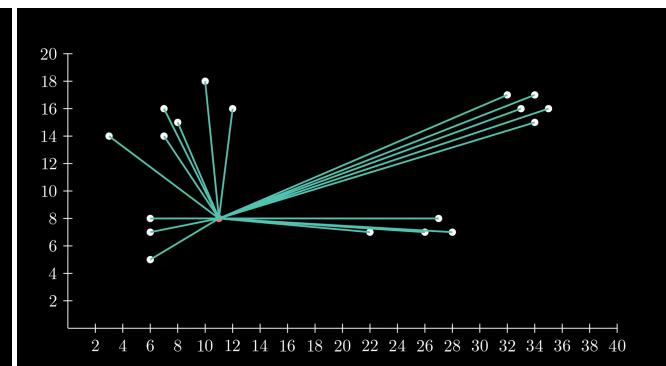
- 1: Chọn điểm centroid c_1 ngẫu nhiên từ dữ liệu ban đầu
- 2: Chọn centroid tiếp theo $c_i = x' \in \chi$ với xác suất $\frac{D(x')^2}{\sum_{x \in \chi} D(x)^2}$.
- 3: Lặp lại bước 2 cho đến khi chọn được k điểm trung tâm
- 4: Thực hiện các bước tiếp theo tương tự thuật toán K-means

Một điều đang chú ý ở đây là tại sao bước 2 lại chọn các centroid theo xác suất mà không phải là chọn luôn điểm có xác suất lớn nhất (tương ứng với $D(x)$ lớn nhất). Lý do là cách chọn điểm có $D(x)$ lớn nhất như vậy sẽ dẫn đến khả năng cao điểm được chọn chính là điểm ngoại lai (outlier), điều đó dẫn tới kết quả phân cụm không tốt. Do đó việc chọn theo xác suất sẽ giảm thiểu sự ảnh hưởng của điểm ngoại lai trong việc chọn trung tâm ban đầu. Tất nhiên nếu bộ dữ liệu đã được xử lý các điểm ngoại lai thì chiến lược chọn điểm có $D(x)$ lớn nhất sẽ đem lại hiệu quả tốt.

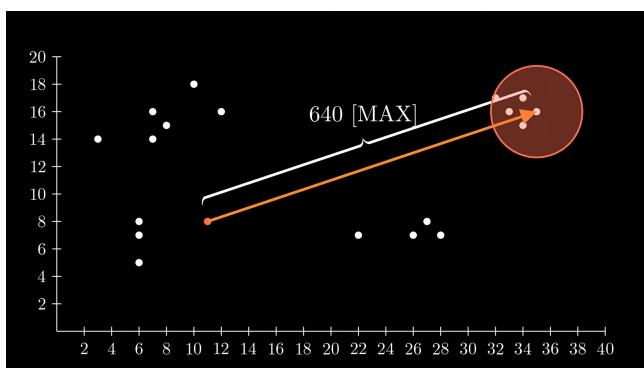
Ví dụ minh họa về cách chọn điểm centroid của K-means++ với chiến lược chọn điểm có $D(x)$ lớn nhất:



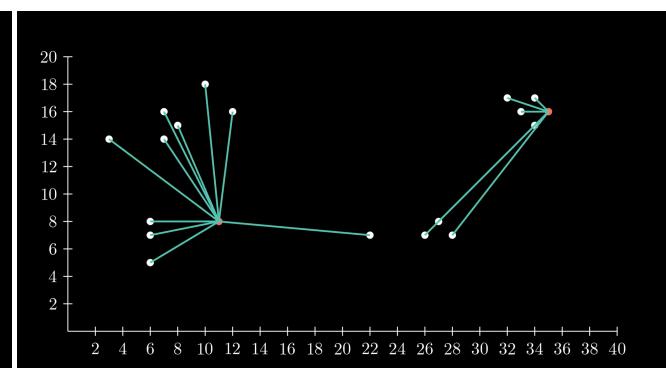
Hình 16: Select 1th centroid



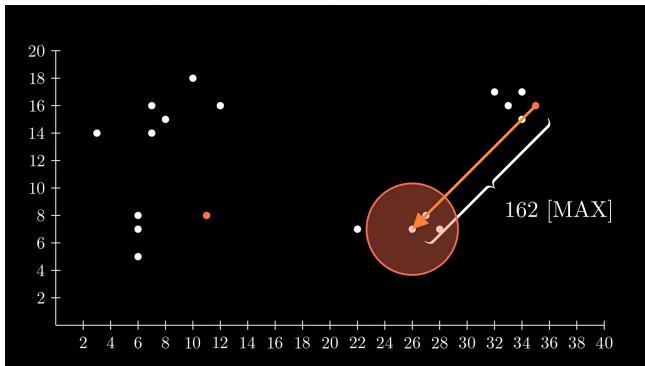
Hình 17: Calculation $D(x)$ to select 2th centroid



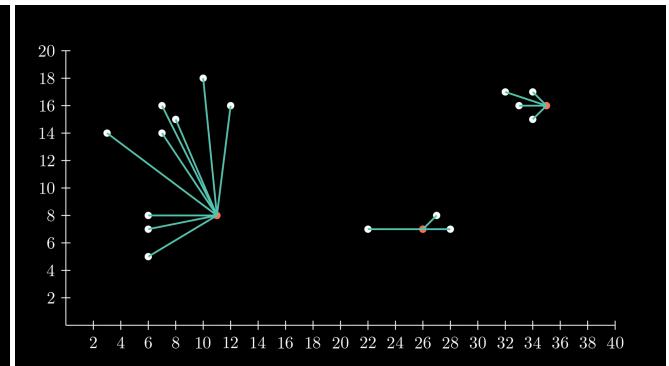
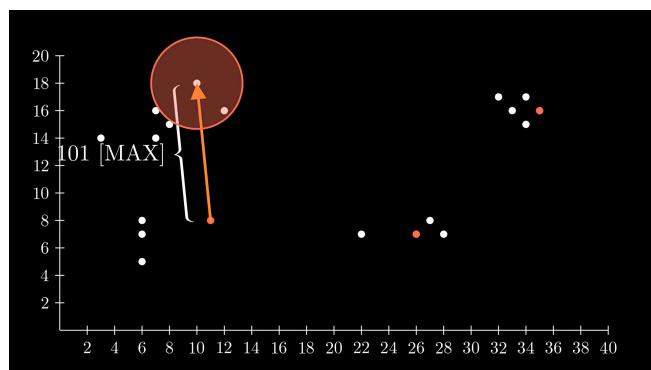
Hình 18: Select 2th centroid



Hình 19: Calculation $D(x)$ to select 3th centroid



Hình 20: Select 3th centroid

Hình 21: Calculation $D(x)$ to select 4th centroid

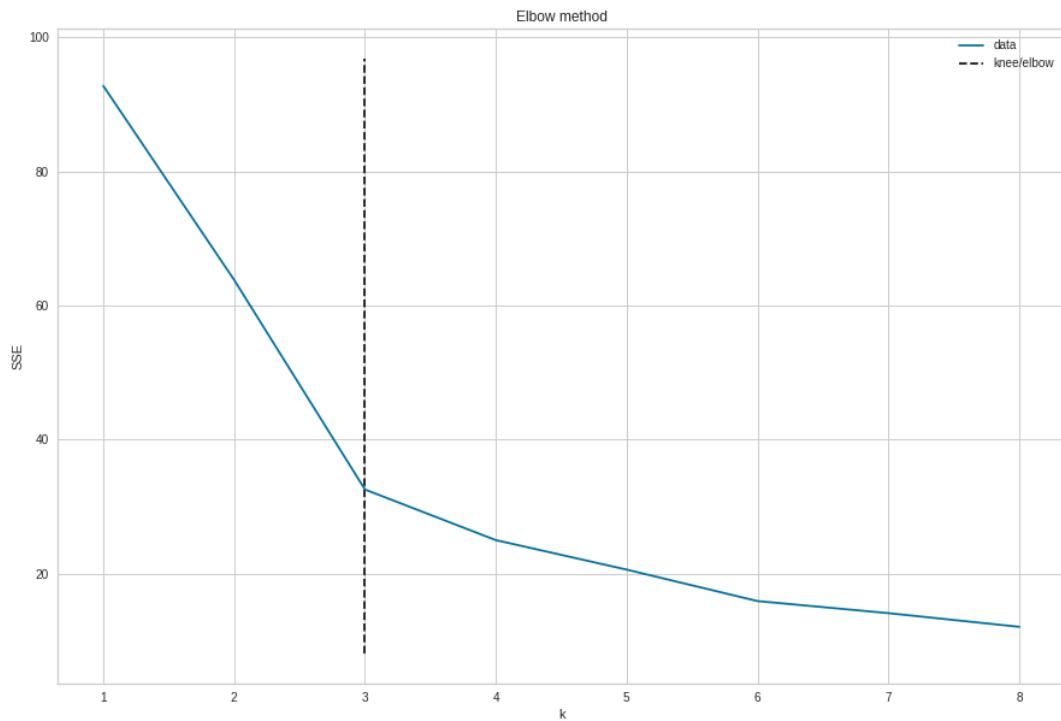
Hình 22: Select 4th centroid

Mặc dù việc khởi tạo trong K-means++ đắt hơn về mặt tính toán so với thuật toán K-means nhưng thời gian chạy để hội tụ giảm đáng kể và đã được tác giả chứng minh với tốc độ $O(\log(k))$. Điều này là do các trung tâm ban đầu được chọn có khả năng nằm trong các cụm khác nhau.

4.2.3 Phương pháp Elbow (Elbow method)

Trong thuật toán K-means chúng ta cần phải xác định trước số cụm k . Câu hỏi đặt ra là số cụm bao nhiêu cho một bộ dữ liệu cụ thể? Phương pháp Elbow là một cách giúp ta lựa chọn được số lượng các cụm phù hợp dựa vào đồ thị trực quan hóa bằng cách nhìn vào sự suy giảm của hàm tổng lỗi phân cụm SSE và lựa chọn ra điểm khuỷu tay.

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 khuỷu tay trên đồ thị đấy (điểm mà đồ thị gấp khúc nhiều nhất) sẽ cho ta cách chia cụm hợp lý theo phương pháp khuỷu tay.



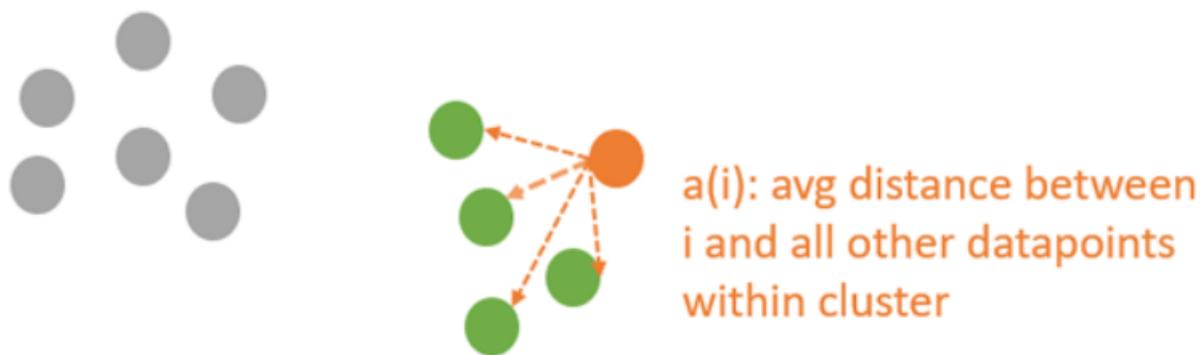
Hình 23: Minh họa phương pháp Elbow với điểm khuỷu tay tại $k = 3$

4.2.4 Đánh giá 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:

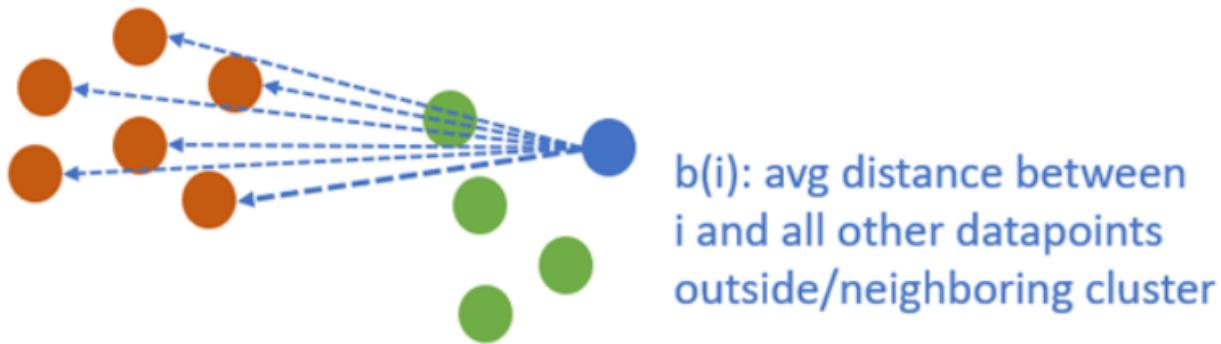
$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

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



Hình 24: Average distance between i and all other datapoints within cluster

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



Hình 25: Average distance between i and all other datapoints outside/neighboring cluster

$s(i)$ 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 S là chỉ số Silhouette trung bình.

- Nếu $S = 1$ thì 2 cụm đang xét cách xa nhau.
- Nếu $S = 0$ thì 2 cụm đang xét đang rất gần nhau.
- Nếu $S = -1$ 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 dựa trên đánh giá biểu đồ Silhouette.

4.3 Kết quả thực nghiệm

Phân cụm khách hàng (Customer Segmentation)

Sử dụng dữ liệu khách hàng để xem xét cách thức hoạt động của thuật toán này. Ví dụ này nhằm mục đích chia khách hàng thành một số nhóm và quyết định cách nhóm khách hàng thành các cụm để tăng giá trị khách hàng và doanh thu của công ty. Trường hợp sử dụng này thường được gọi là phân khúc khách hàng.

Chi tiết bộ dữ liệu xem tại: <https://www.kaggle.com/datasets/shwetabh123/mall-customers>

Thông tin bộ dữ liệu:

Thông tin của bộ dữ liệu Mall Customers:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   CustomerID      200 non-null    int64  
 1   Genre            200 non-null    object  
 2   Age              200 non-null    int64  
 3   Annual Income (k$) 200 non-null    int64  
 4   Spending Score (1-100) 200 non-null    int64  
dtypes: int64(4), object(1)
memory usage: 7.9+ KB
```

4 Phân cụm không phân cấp

Các thông số thống kê:

	CustomerID	Age	Annual Income (k\$)	Spending Score (1-100)
count	200.000000	200.000000	200.000000	200.000000
mean	100.500000	38.850000	60.560000	50.200000
std	57.879185	13.969007	26.264721	25.823522
min	1.000000	18.000000	15.000000	1.000000
25%	50.750000	28.750000	41.500000	34.750000
50%	100.500000	36.000000	61.500000	50.000000
75%	150.250000	49.000000	78.000000	73.000000
max	200.000000	70.000000	137.000000	99.000000

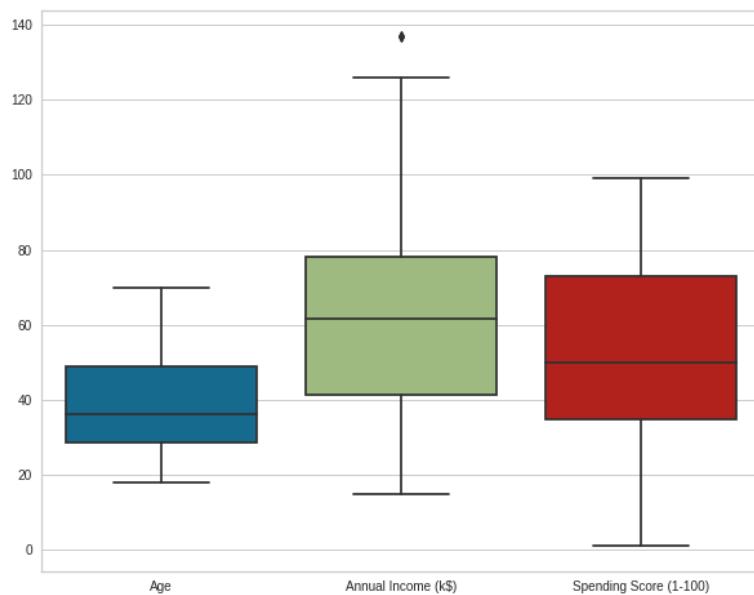
Các hàng đầu tiên của bộ dữ liệu:

	CustomerID	Genre	Age	Annual Income (k\$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40
...

Với bộ dữ liệu này, tiến hành loại bỏ 2 features "CustomerID" và "Genre".

Loại bỏ outlier:

Vẽ biểu đồ box plot:



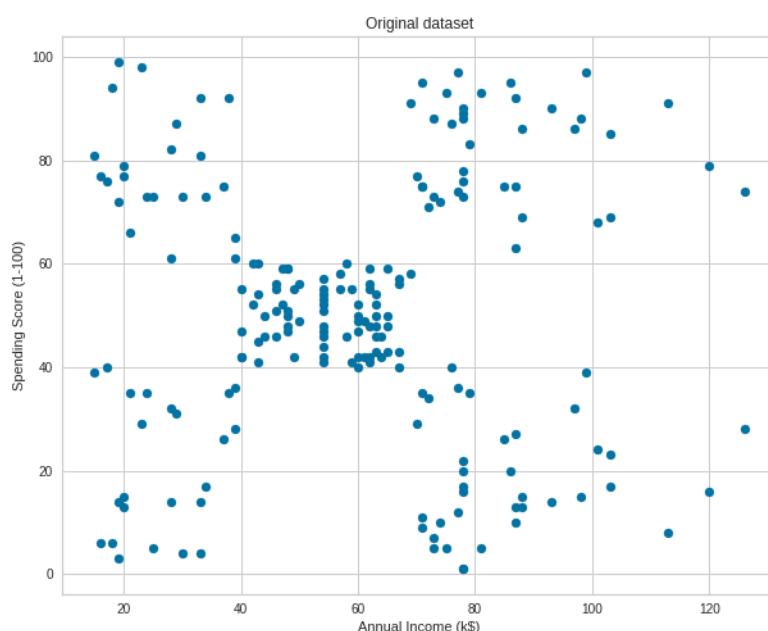
Nhận thấy có 1 số điểm ngoại lai tại "Annual Income (k\$)". Tiến hành loại bỏ bằng quy tắc 1.5IQR.

Thông tin của bộ dữ liệu sau khi loại bỏ ngoại lai còn lại 198 điểm dữ liệu:

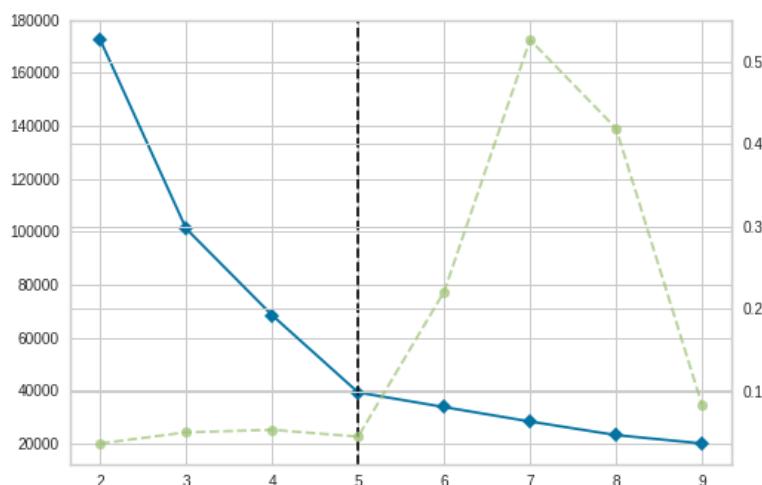
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 198 entries, 0 to 197
Data columns (total 3 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Age              198 non-null    int64  
 1   Annual Income (k$) 198 non-null    int64  
 2   Spending Score (1-100) 198 non-null   int64  
dtypes: int64(3)
memory usage: 14.3 KB
```

Phân cụm K-means với 2 features "Annual Income (k\$)" và "Spending Score (1-100)":

Biểu diễn sự phân bố ban đầu của dữ liệu:

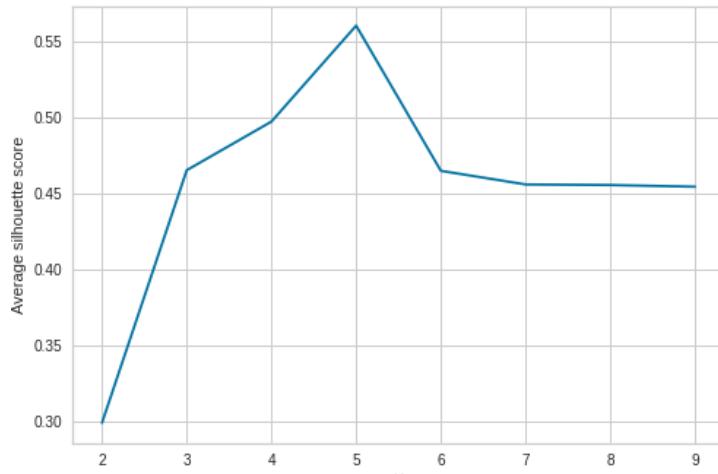


Thực hiện phương pháp Elbow để tìm số cụm tối ưu:



4 Phân cụm không phân cấp

Thực hiện phương pháp phân tích Silhouette:

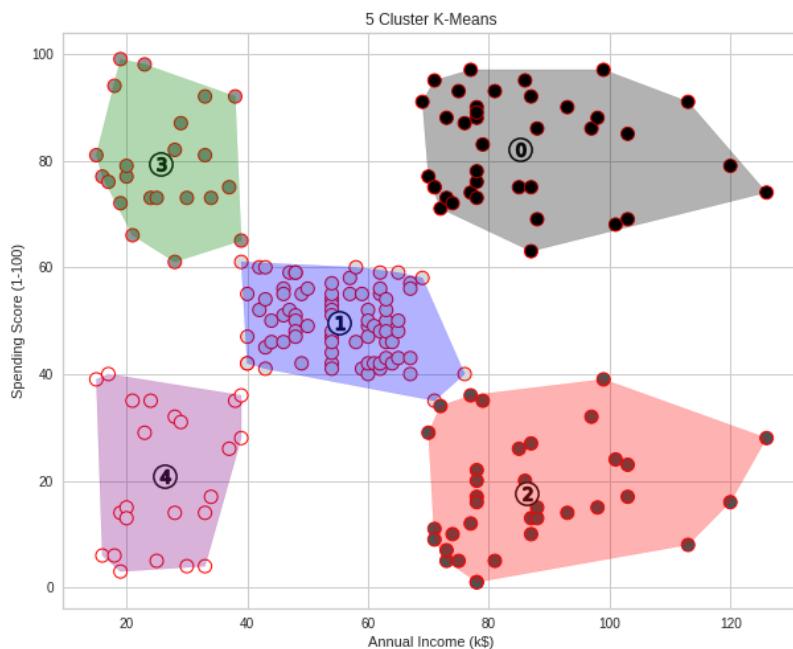


Các kết quả đều cho thấy với $k = 5$ sẽ có kết quả tối ưu do đó lựa chọn $k = 5$ tiến hành phân cụm.

Kết quả 5 tâm cụm thu được:

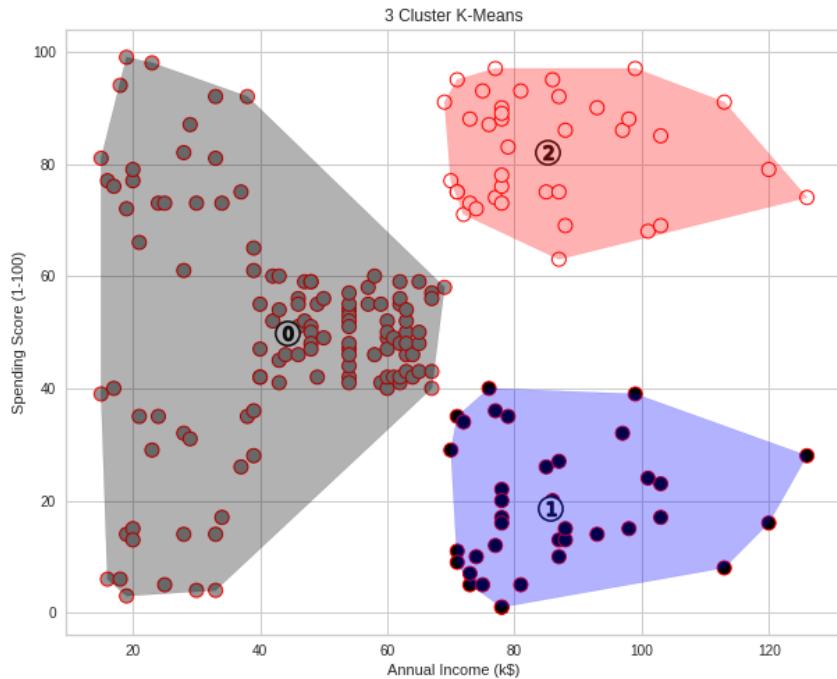
	Annual Income (k\$)	Spending Score (1-100)
0	55.087500	49.712500
1	25.727273	79.363636
2	85.210526	82.105263
3	26.304348	20.913043
4	86.342857	17.571429

Biểu diễn kết quả 5 cụm :



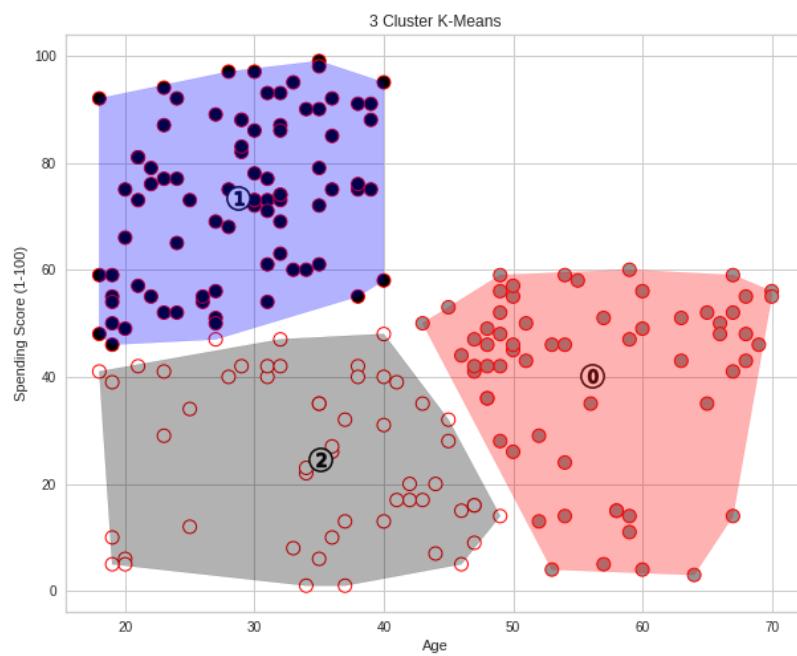
Từ các kết quả trên có thể đưa ra một số nhận xét cơ bản như các khách hàng ở cụm 0, cụm 2 và cụm 3 có xu hướng mức tiêu phù hợp với mức thu nhập. Cụm 1 có xu hướng mức chi tiêu cao nhưng mức thu nhập thấp, ngược lại với cụm 4.

Trên thực tế, các đơn vị kinh doanh trong việc phân cụm khách hàng thường được phân loại thành 3 loại: khách hàng nóng, khách hàng ấm, khách hàng lạnh. Cho nên tiếp tục tiến hành phân cụm với số cụm là $k = 3$:



Phân cụm K-means với 2 features "Age" và "Spending Score (1-100)":

Thực hiện phân cụm K-means thành 3 cụm cho 2 features "Age" và "Spending Score (1-100)" thu được kết quả:



4 Phân cụm không phân cấp

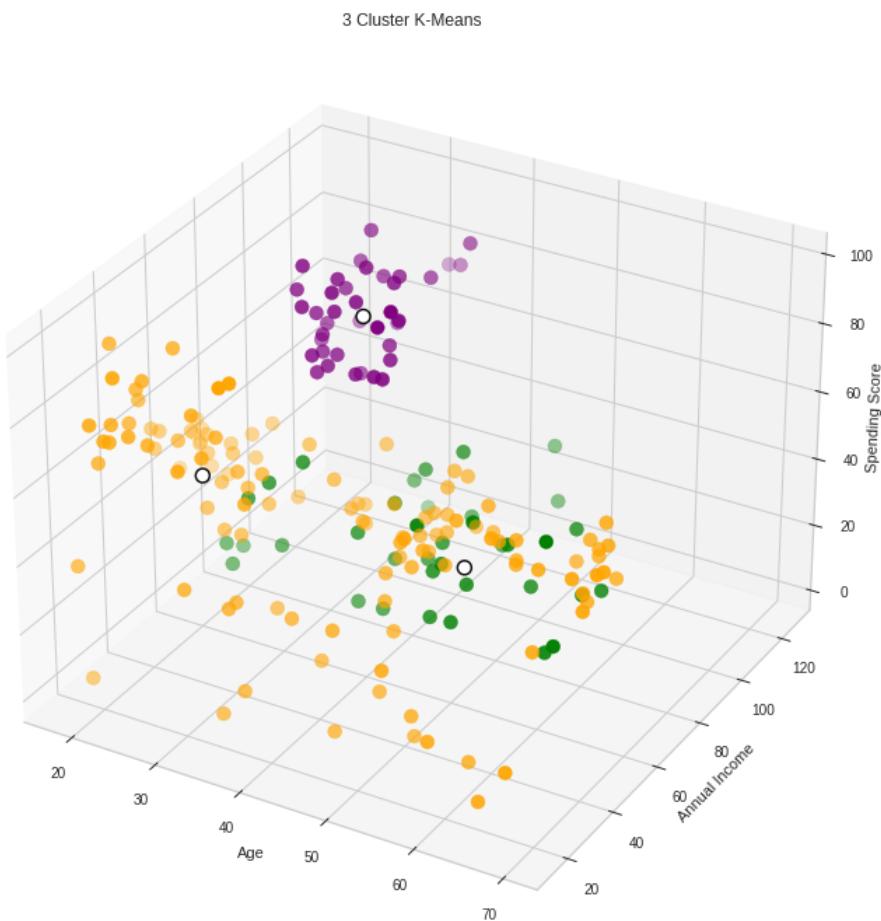
Phân cụm K-means với 3 features Phân cụm K-means với 3 features "Age", "Annual Income (k\$)" và "Spending Score (1-100)":

Thực hiện phân cụm K-means thành 3 cụm cho 3 features "Age", "Annual Income (k\$)" và "Spending Score (1-100)" thu được kết quả:

Các tâm cụm:

	Age	Annual Income (k\$)	Spending Score (1-100)
0	25.850746	42.238806	54.268657
1	50.989130	61.967391	33.967391
2	32.948718	84.794872	81.487179

Biểu diễn các cụm:



Nhìn chung, từ các kết quả phân cụm cho thấy được: nhóm khách hàng nóng thường có đặc điểm chung là độ tuổi trẻ, mức thu nhập và chi tiêu đều cao; nhóm khách hàng ấm có đặc điểm là độ tuổi trung bình, mức thu nhập từ thấp đến trung bình, mức chi tiêu đều có từ thấp đến cao; nhóm khách hàng lạnh có đặc điểm là độ tuổi cao, mức thu nhập cao nhưng mức chi tiêu thấp.

Nén ảnh (Image Compression)

Một trong những ứng dụng khác của phân cụm đó là nén ảnh. Ý tưởng nén ảnh bằng phân cụm là phân cụm tập các điểm ảnh ban đầu thành k cụm, tương ứng với k màu khác nhau. Sau đó thực hiện thay giá trị tất cả các điểm ảnh bằng chính tâm cụm của cụm mà điểm ảnh đó thuộc về. Khi đó, ảnh ban đầu của ta có rất nhiều màu khác nhau thành bức ảnh chỉ còn k màu.

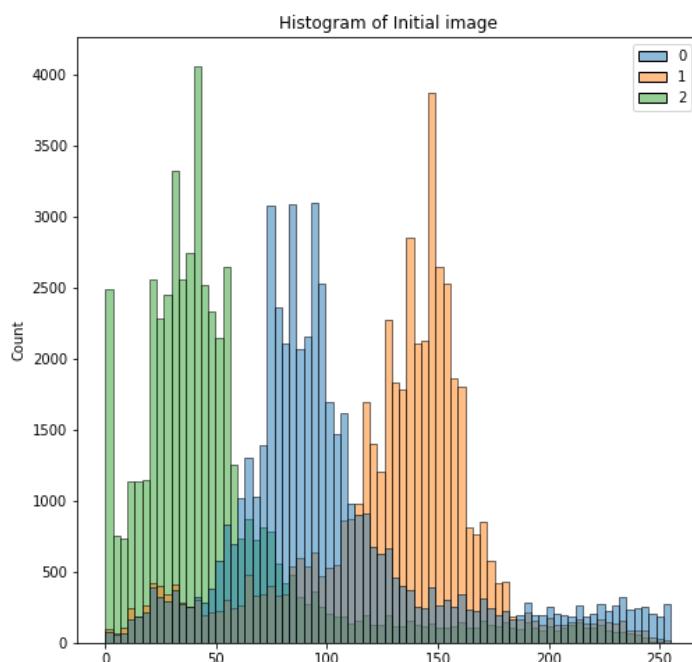
Trong phần này, chúng ta sẽ thực hiện nén bức ảnh bên dưới:



Kích thước của bức ảnh là (168, 300, 3).

Đầu tiên cần phải định hình lại bức ảnh thành ảnh 2D có kích thước $(168 \times 300, 3) = (50400, 3)$. Tức bức ảnh ban đầu có thể có tối đa 50400 màu khác nhau.

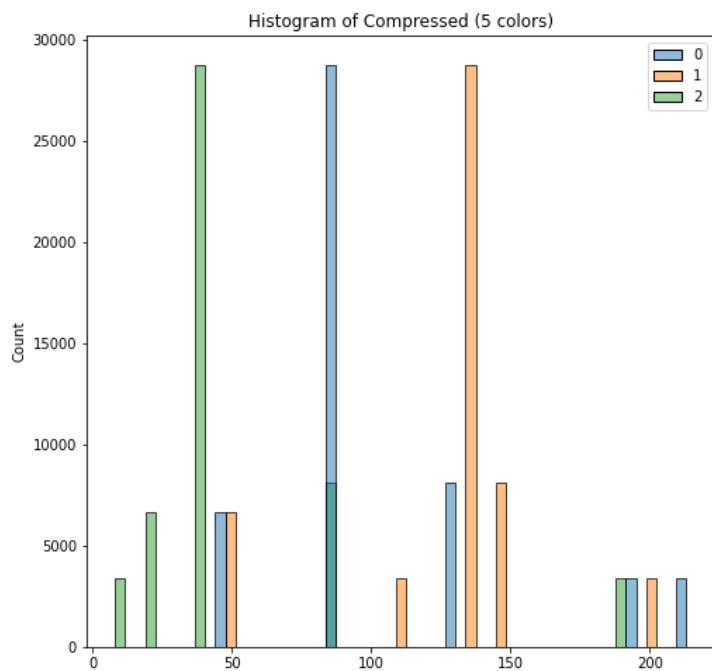
Biểu đồ histogram:



Thực hiện phân cụm K-means với $k = 5$. Và gán lại các giá trị của từng điểm bằng tâm cụm.

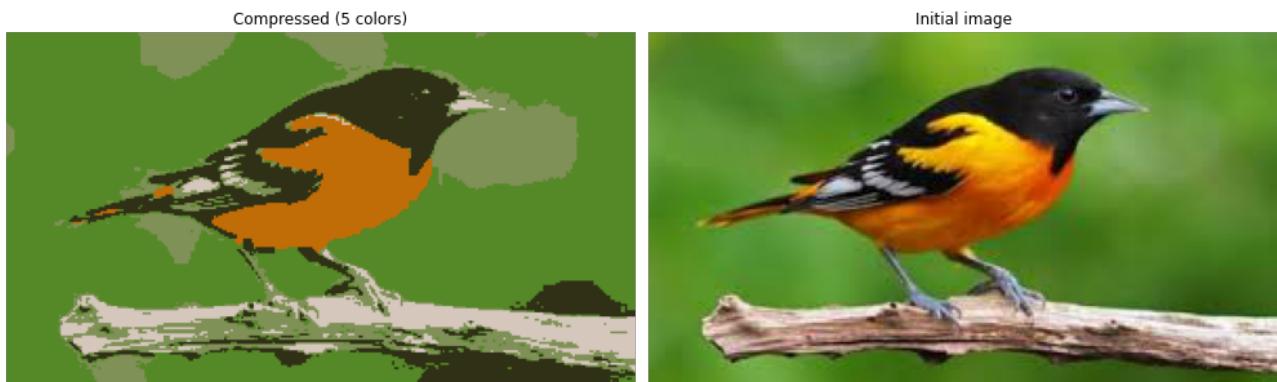
4 Phân cụm không phân cấp

Biểu đồ histogram của bức ảnh:

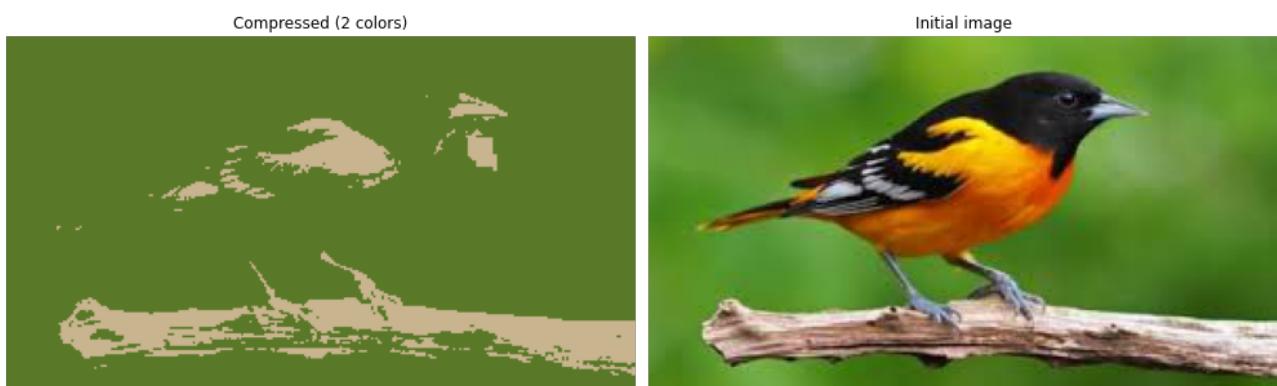


Số màu của bức ảnh giờ chỉ còn lại là 5 màu tương ứng với 5 tâm cụm.

Bức ảnh sau khi nén:



Kết quả nén ảnh với một số giá trị k khác:



Compressed (4 colors)



Initial image



Compressed (64 colors)



Initial image



5 Phân cụm dựa theo mô hình xác suất thống kê

Những phương pháp phân cụm đã được đề cập đến trong bài báo cáo này bao gồm phương pháp kết nối đơn, kết nối toàn phần, kết nối trung bình, phương pháp Ward (phương pháp K-means),

5.1 Mô hình Gaussian hỗn hợp

5.1.1 Nhắc lại về phân phối Gaussian đa chiều

Trong không gian d chiều, một biểu diễn của véc tơ $\mathbf{x} = (x_1, x_2, \dots, x_d)$ được định nghĩa là:

$$f_{\mathbf{x}}(x_1, x_2, \dots, x_d) = \frac{1}{(2\pi)^{d/2} \sqrt{|\Sigma|}} \exp \left\{ \left(-\frac{1}{2}(\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \right) \right\}$$

Hoặc chúng ta có thể viết:

$$N(\mathbf{x} | \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} \sqrt{|\Sigma|}} \exp \left\{ \left(-\frac{1}{2}(\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \right) \right\}$$

Trong đó μ là véc tơ kì vọng theo các chiều của \mathbf{x} và Σ là ma trận hiệp phương sai (covariance matrix). Ma trận hiệp phương sai của véc tơ ngẫu nhiên $\mathbf{x} = (x_1, x_2, \dots, x_d)$ có công thức như sau:

$$\text{Cov}(\mathbf{x}) = \begin{bmatrix} \text{Cov}(x_1, x_1) & \text{Cov}(x_1, x_2) & \dots & \text{Cov}(x_1, x_d) \\ \text{Cov}(x_2, x_1) & \text{Cov}(x_2, x_2) & \dots & \text{Cov}(x_2, x_d) \\ \vdots & \vdots & \ddots & \vdots \\ \text{Cov}(x_d, x_1) & \text{Cov}(x_d, x_2) & \dots & \text{Cov}(x_d, x_d) \end{bmatrix}$$

5.1.2 Mô hình Gaussian hỗn hợp

Mô hình Gaussian hỗn hợp (Gaussian Mixture Model) viết tắt là GMM là một mô hình phân cụm thuộc lớp bài toán học không giám sát mà phân phối xác xuất cú mỗi một cụm được giả định là phân phối Gaussian đa chiều. SỞ dĩ mô hình được gọi mà Mixture là vì xác suất của mỗi điểm dữ liệu không chỉ phụ thuộc vào một phân phối Gaussian duy nhất mà là kết hợp từ nhiều phân phối Gaussian khác nhau từ mỗi cụm.

Ta sẽ xem xét mô hình toán học sau: Xét X là một biến ngẫu nhiên D chiều biểu diễn dữ liệu. Biến ngẫu nhiên X có hàm mật độ là:

$$f_{\text{mix}} = \sum_{k=1}^K p_k f(x, \mu_k, \Sigma_k)$$

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 trên, 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 $N(\mu, \Sigma)$, với μ và \sum 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, $f(x, \mu_k, \sum_k) \sim N(\mu, \Sigma)$. Khi đó hàm mật độ f_{mix} sẽ có dạng:

$$f_{\text{mix}} = \sum_{k=1}^K p_k f(x, \mu_k, \Sigma_k) = \sum_{k=1}^K p_k \frac{1}{(2\pi)^{D/2} |\Sigma_k|^{1/2}} \exp \left(-\frac{1}{2} (x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k) \right)$$

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 $\{p_j\}_{j=1}^K, \{\mu_j\}_{j=1}^K, \{\Sigma_j\}_{j=1}^K$ để cực đại hóa hàm hợp lí sau:

$$L\left(\{p_j\}_{j=1}^K, \{\mu_j\}_{j=1}^K, \{\Sigma_j\}_{j=1}^K\right) = \prod_{j=1}^N f_{\text{mix}}\left(x_j | \{p_j\}_{j=1}^K, \{\mu_j\}_{j=1}^K, \{\Sigma_j\}_{j=1}^K\right) = \prod_{j=1}^N \sum_{k=1}^K p_k f(x, \mu_k, \Sigma_k)$$

Để đơn giản về mặt kí hiệu, ta sẽ đặt $\theta = \left(\{p_j\}_{j=1}^K, \{\mu_j\}_{j=1}^K, \{\Sigma_j\}_{j=1}^K\right)$ Vậy khi đó ta sẽ phát biểu bài toán như sau:

Bài toán (P): Tìm θ để cực đại hóa hàm:

$$L(\theta) = \prod_{j=1}^N \sum_{k=1}^K p_k f(x, \mu_k, \Sigma_k)$$

Với điều kiện:

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

Ta có vài nhận xét như 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.

5.2 Thuật toán cực đại hóa kỳ vọng EM (Expectation Maximization)

Thuật toán cực đại hóa kỳ 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 kỳ 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 kỳ vọng sẽ có đầu vào và đầu ra lần lượt là:

- Đầu vào: Số phân cụm K , N điểm dữ liệu, phân phối tiền nghiệm $f_{\text{mix}}(x)$.
- Đầu ra: Bộ hệ số $\hat{\theta} = \left(\{p_j\}_{j=1}^K, \{\mu_j\}_{j=1}^K, \{\Sigma_j\}_{j=1}^K\right)$ tối ưu.

Ngoài tham số θ cần tìm, ta sẽ có các kí hiệu sau phục vụ cho thuật toán cực đại hóa kỳ vọng:

- Biến ngẫu nhiên X tuân theo phân phối có hàm mật độ là $f_{\text{mix}}(x)$.
- Ta xét $x = (x_1, x_2, \dots, x_N)$ 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:
 - $X|(Z=i) \sim N(\mu_i, \Sigma_i) \forall i = \{1, 2, \dots, K\}$
 - $P(Z=i) = p_i \forall i = \{1, 2, \dots, K\}$
- Cùng với đó ta xét $z = (z_1, z_2, \dots, z_N)$ là một vector chứa N quan sát độc lập của biến ngẫu nhiên Z . Ý nghĩa của vector quan sát 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:

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

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

$$L(\theta, x) = P(x|\theta) = \prod_{j=1}^N f_{\text{mix}}(x_j|\theta) = \prod_{j=1}^N \sum_{k=1}^K p_k f(x_j, \mu_k, \Sigma_k)$$

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:

$$L(\theta, X, Z) = P(X, Z|\theta) = \prod_{j=1}^N [f(X, \mu_{z_j}, \Sigma_{z_j}) p_{z_j}]^{\mathbb{I}(Z=j)}$$

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

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

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

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

$$\log L(\theta; x; z) = \sum_{i=1}^N \sum_{j=1}^K \mathbb{I}(z_j = j) [\log(p_j) - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2} (x_j - \mu_j)^T \Sigma_j^{-1} (x_j - \mu_j) - \frac{D}{2} \log 2\pi]$$

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

Định lý 5.1. Xét $Q(\theta, \theta') = E_{Z|X, \theta'}[\log L(\theta, X, Z)]$. Khi đó nếu $Q(\theta, \theta') > Q(\theta', \theta')$ thì ta sẽ có $L(\theta, X) > L(\theta', X)$

Chứng minh:

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

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

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

Lấy tổng i từ 1 đến K ta có:

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

Đặt $H(\theta, \theta') = -\sum_{i=1}^K P(Z = i|X, \theta') \log P(Z|X, \theta)$ và chú ý rằng ta có:

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

$$\Rightarrow \log P(X|\theta) = Q(\theta, \theta') + H(\theta, \theta') \quad \forall \theta$$

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

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 độ đo Kullback - Leibler, là một phép đo cách một phân phối khác biệt so với phân phối xác suất tham chiếu.

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

Đây là điều cần chứng minh.

Vậy dựa vào định lý (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 $Q(\theta, \theta^t)$ trong đó θ^t là giá trị tham số tìm được ở vòng lặp thứ t .

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

$$\begin{aligned} Q(\theta, \theta^t) &= \mathbb{E}_{Z|X, \theta^t} [\log L(\theta, x, z)] = \mathbb{E}_{Z|X, \theta^t} \left[\log \prod_{i=1}^N L(\theta, x_i, z_i) \right] \\ &= \sum_{i=1}^N \mathbb{E}_{Z|X, \theta^t} [\log L(\theta, x_i, z_i)] = \sum_{i=1}^N \sum_{j=1}^K P(z_j = j, X = x_i, \theta^t) \log L(\theta, x_i, z_j = j) \\ &= \sum_{i=1}^N \sum_{j=1}^K P(z_j = j, X = x_i, \theta^t) \left[\log(p_j) - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2} (x_i - \mu_j)^T (\Sigma_j)^{-1} (x_i - \mu_j) - \frac{D}{2} \log 2\pi \right] \end{aligned}$$

Ngoài ra theo công thức Bayes, ta có:

$$z_{ij} = P(z_j = j|X = x_i, \theta^t) = \frac{P(z_j = j)P(X = x_i, \theta^t|z_j = j)}{\sum_{j=1}^K P(z_j = j)P(X = x_i, \theta^t|z_j = j)}$$

5 Phân cụm dựa theo mô hình xác suất thống kê

$$= \frac{p_j(t)f(x_j, \mu_j(t), \Sigma_j(t))}{\sum_{j=1}^K p_j(t)f(x_j, \mu_j(t), \Sigma_j(t))} \quad (14)$$

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

Như vậy ta đã tính xong hàm $Q(\theta, \theta^t)$. Chú ý rằng các hệ số \widehat{z}_{ij} ta tính được dựa vào tham số θ^t và hàm $Q(\theta, \theta^t)$ có các biến là $(\{p_j\}_{j=1}^K, \{\mu_j\}_{j=1}^K, \{\Sigma_j\}_{j=1}^K)$. Sau khi đã tính hàm $Q(\theta, \theta^t)$ nhiệm vụ của ta bây giờ sẽ là tìm θ là điểm cực đại của hàm Q . Cụ thể, ta sẽ phải đi giải bài toán tối ưu (P') sau:

$$\sum_{i=1}^N \sum_{j=1}^K z_{ij} \left[\log(p_j) - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2}(x_i - \mu_j)^T (\Sigma_j)^{-1} (x_i - \mu_j) - \frac{D}{2} \log 2\pi \right]$$

Với điều kiện:

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

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:

$$\begin{cases} \widehat{p}_k = \frac{n_k}{N} \\ \widehat{\mu}_k = \frac{\sum_{i=1}^N \widehat{z}_{ik} x_i}{n_k} \\ n_k = \sum_{i=1}^N \widehat{z}_{ik} \\ \widehat{\Sigma}_k = \frac{\sum_{i=1}^N \widehat{z}_{ik} (x_i - \widehat{\mu}_k)(x_i - \widehat{\mu}_k)^T}{\sum_{i=1}^N \widehat{z}_{ik}} \end{cases} \quad (15)$$

Công thức 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 $\widehat{\Sigma}_k$. Mã giả của thuật toán cực đại hóa kì vọng sẽ được miêu tả trong thuật toán (5).

Algorithm 5 Thuật toán cực đại hóa kỳ 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 hệ số $\{p_j\}_{j=1}^K$.
 - 3: Khởi tạo 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 \widehat{z}_{ij} như trong công thức (14).
 - 6: Thực hiện bước M: Cập nhật các hệ số $(\{\widehat{p}_i\}_{i=1}^K, \{\widehat{\mu}_i\}_{i=1}^K, \{\widehat{\Sigma}_i\}_{i=1}^K)$ như trong công thức (15).
 - 7: $t := t + 1$
 - 8: end while
 - 9: Output: Bộ hệ số tối ưu $\widehat{\theta} = (\{\widehat{p}_i\}_{i=1}^K, \{\widehat{\mu}_i\}_{i=1}^K, \{\widehat{\Sigma}_i\}_{i=1}^K)$
-

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 lý.

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

Để xác định số phân cụm, ta sẽ đi tìm số phân cụm K sao cho hàm L_{total} là nhỏ nhất:

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

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 số phân cụm K .
- Penalty là một hàm phạt phụ thuộc vào độ phức tạp của mô hình (thông thường phụ thuộc vào số biến của mô hình).

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 $\Sigma_k \in \mathbb{R}^{D \times D}$.

⇒ 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) ⇒ Ta cần xác định thêm $K - 1$ biến.

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

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

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

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

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

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

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

5.4 Kết quả thực nghiệm

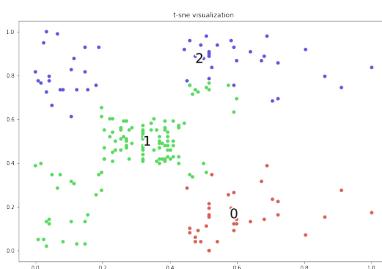
Ta sử dụng bộ dữ liệu khách hàng Mall Customers như trong thuật toán K-means.

Với số phân cụm là 3, ta sẽ được các phân cụm có giá trị trung bình và ma trận hiệp phương sai như sau:

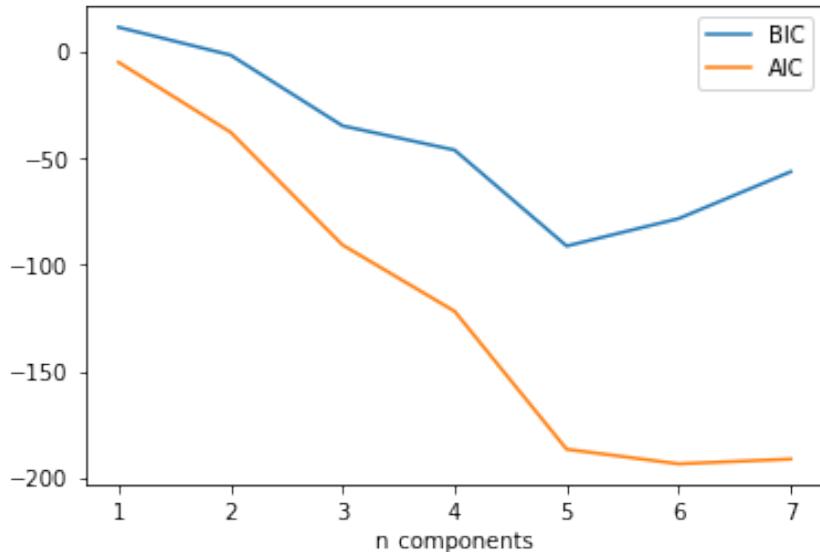
5 Phân cụm dựa theo mô hình xác suất thống kê

```
means:  
[[ 0.6010665  0.16017669]  
[ 0.30524379  0.46719907]  
[ 0.37825574  0.82961932]]  
covariances:  
[[[ 0.01827869  0.00365932]  
[ 0.00365932  0.00961065]]  
[  
[[ 0.02028609  0.01770041]  
[ 0.01770041  0.03144144]]  
[  
[[ 0.08073891  0.00439822]  
[ 0.00439822  0.0099174 ]]]
```

Hình ảnh biểu diễn phân bố của các cụm ($K = 3$):



Tính toán tiêu chuẩn AIC và BIC ta có biểu đồ sau:



Ta thấy BIC đạt giá trị thấp nhất khi số phân cụm là 5. Phân cụm lại bộ dữ liệu với số phân cụm là 5, ta được trung bình và ma trận hiệp phương sai của các ma trận như sau:

```

means:
[[ 0.60502531  0.15433196]
 [ 0.33368985  0.49394756]
 [ 0.58393969  0.82673863]
 [ 0.0829305   0.80743088]
 [ 0.09861098  0.21597752]]
covariances:
[[[ 0.01818446  0.00433814]
 [ 0.00433814  0.00873064]]

 [[ 0.00613567 -0.00231927]
 [-0.00231927  0.0051635 ]]

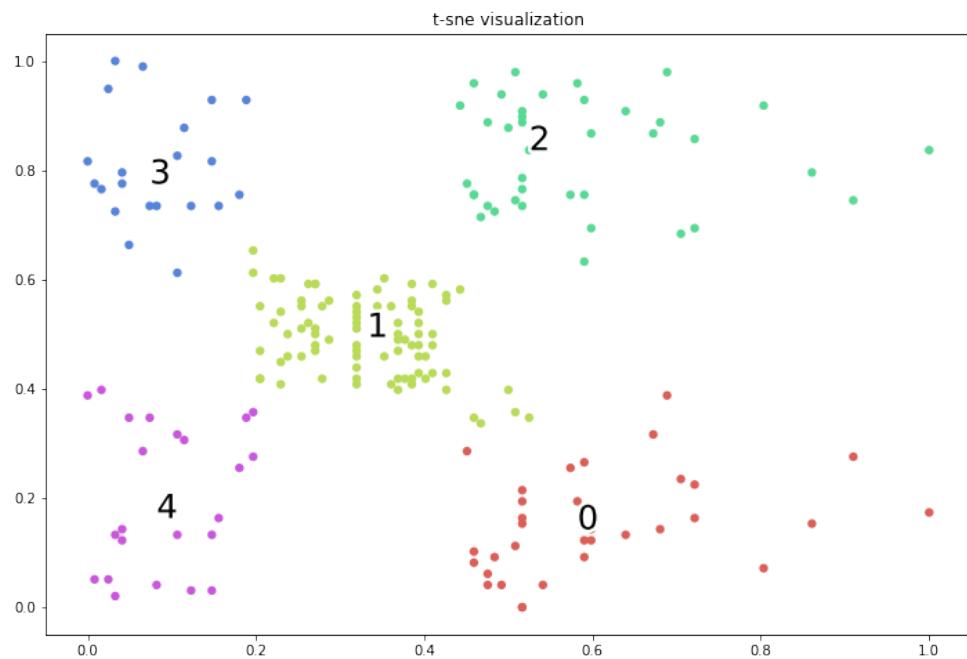
 [[ 0.01808598 -0.00031096]
 [-0.00031096  0.0091568 ]]

 [[ 0.00337483 -0.0001437 ]
 [-0.0001437   0.01026088]]

 [[ 0.00453005  0.00255303]
 [ 0.00255303  0.01918353]]]

```

Hình ảnh phân bố của các cụm ($K = 5$):



5.5 Tổng kết

GMM là một mô hình xác suất. Mô hình này thể hiện sự cải tiến so với K-Means, đó là các điểm dữ liệu được sinh ra từ một phân phối hỗn hợp của hữu hạn các phân phối Gaussian đo chiều. Tham số của những phân phối này được giả định là chưa biết. Để tìm ra tham số huấn luyện cho mô hình thì chúng ta tối đa hóa hàm auxiliary thông qua thuật toán EM, thuật toán này sẽ cập nhật nghiệm sau mỗi vòng lặp để đi đến điểm cực trị. Chúng ta có thể coi rằng GMM như là một dạng khái quát của thuật toán K-Means nhằm kết hợp với thông tin về hiệp phương sai của dữ liệu cũng như là tâm của các phân phối Gaussian tiềm ẩn.

6 Chia tỉ lệ nhiều chiều

Giả sử có n quan sát trong không gian p chiều. Giữa mỗi cặp quan sát bất kì (r, s) lại có một phép đo δ_{rs} biểu thị mức độ khác nhau giữa hai vật r và s . Giờ ta mong muốn biểu diễn các vật trên trong không gian để xem xét mối liên hệ của chúng. Đơn giản thì ta có thể coi mỗi quan sát thứ r là một điểm với các tọa độ $(n_{r1}, n_{r2}, \dots, n_{rp})$, nhưng làm thế thì chỉ có thể biểu diễn các điểm với điều kiện $p \leq 3$. Ngoài ra nếu $p \leq 3$, khi đó các điểm biểu diễn theo cách này lại không thể hiện được độ sai khác giữa mỗi cặp phần tử là δ .

Chia tỉ lệ nhiều chiều (*Multidimensional Scaling* hay viết tắt là MDS) là tập các phương pháp để tìm một cấu hình trong không gian ít chiều hơn p , mỗi điểm trong không gian tương ứng cho một vật và khoảng cách giữa các vật là phù hợp với sự khác nhau δ .

Với n vật thì ta có $n(n - 1)/2$ giá trị của δ . Nếu ta chỉ quan tâm đến thứ tự của các giá trị δ để tìm cấu hình biểu diễn các vật, gọi là lớp phương pháp *Non-metric multidimensional scaling*. Nếu ta dùng đến các giá trị δ để tìm cấu hình, đó là lớp phương pháp *metric multidimensional scaling*.

Ngoài ra, thay cho độ đo sự khác nhau δ còn có phép đo sự giống nhau s . Tuy nhiên ta có thể biến đổi linh hoạt giữa hai dạng phép đo này tùy vào mục đích sử dụng cũng như yêu cầu cần giải quyết.

6.1 Dùng khoảng cách trong chia tỉ lệ nhiều chiều và cách làm cổ điển

Giả sử có n vật với phép đo khác nhau $\{\delta_{rs}\}$. Metric MDS sẽ cố gắng tìm một tập hợp các điểm trong không gian, ở đó mỗi điểm tương ứng với một vật và khoảng cách giữa các điểm là $\{d_{rs}\}$ sao cho

$$d_{rs} \approx f(\delta_{rs}) \quad (16)$$

Với f là hàm liên tục đơn điệu.

Gọi \mathbb{O} là tập chứa các vật với độ khác nhau δ được định nghĩa trên tập $\mathbb{O} \times \mathbb{O}$. Với ϕ là một ánh xạ từ tập \mathbb{O} vào tập \mathbb{E} , nơi chứa các điểm tương ứng cho các vật. Đặt $\phi(r) = x_r$, ($r \in \mathbb{O}, x_r \in \mathbb{E}$), và $\mathbb{X} = \{x_r : r \in \mathbb{O}\}$ là tập ảnh. Khoảng cách giữa hai điểm x_r, x_s là d_{rs} . Mục đích của Metric MDS là tìm ánh xạ ϕ sao cho d_{rs} xấp xỉ bằng với $f(\delta_{rs})$ với mọi $r, s \in \mathbb{O}$.

Chia tỉ lệ cổ điển có nguồn gốc từ những năm 1930 khi Young và Householder (1938) chỉ với ma trận khoảng cách giữa các điểm trong không gian Euclidean, tọa độ của các điểm có thể được tìm thấy sao cho khoảng cách được bảo toàn. Torgerson (1952) đã đưa chủ đề này trở nên phổ biến bằng cách sử dụng vào kỹ thuật chia tỷ lệ.

Khôi phục lại tọa độ

Cho n điểm trong không gian Euclidean p chiều. Tọa độ của điểm thứ r là $\mathbf{x}_r = (x_{r1}, x_{r2}, \dots, x_{rp})^T$, ($r = 1, 2, \dots, n$). Do đó khoảng cách giữa hai điểm r và s được định nghĩa như sau:

$$d_{rs}^2 = (\mathbf{x}_r - \mathbf{x}_s)^T (\mathbf{x}_r - \mathbf{x}_s) \quad (17)$$

Và ma trận tích trong \mathbf{B} với

$$[\mathbf{B}]_{rs} = b_{rs} = \mathbf{x}_r^T \mathbf{x}_s. \quad (18)$$

Từ khoảng cách đã biết $\{d_{rs}\}$, ta tìm được ma trận \mathbf{B} , rồi tìm lại được tọa độ các điểm.

Phương pháp tìm ma trận \mathbf{B}

Hiển nhiên khi có một cấu hình thỏa mãn trong không gian nào, bằng việc xoay và tịnh tiến ta lại

thu được các cấu hình khác cũng thỏa mãn. Do vậy để tìm được một cấu hình duy nhất, ta đặt ra điều kiện:

$$\sum_{r=1}^n x_{ri} = 0, (i = 1, 2, \dots, p). \quad (19)$$

Như vậy trọng tâm của cấu hình là điểm $(0, 0, \dots, 0)$. Để tìm \mathbf{B} , trước tiên ta khai triển đẳng thức (17):

$$d_{rs}^2 = \mathbf{x}_r^T \mathbf{x}_r + \mathbf{x}_s^T \mathbf{x}_s - 2\mathbf{x}_r^T \mathbf{x}_s. \quad (20)$$

Kết hợp với (19) ta thu được các đẳng thức:

$$\frac{1}{n} \sum_{r=1}^n d_{rs}^2 = \frac{1}{n} \sum_{r=1}^n \mathbf{x}_r^T \mathbf{x}_r + \mathbf{x}_s^T \mathbf{x}_s \quad (21)$$

$$\frac{1}{n} \sum_{s=1}^n d_{rs}^2 = \mathbf{x}_r^T \mathbf{x}_r + \frac{1}{n} \sum_{s=1}^n \mathbf{x}_s^T \mathbf{x}_s \quad (22)$$

$$\frac{1}{n^2} \sum_{r=1}^n \sum_{s=1}^n d_{rs}^2 = \frac{2}{n} \sum_{r=1}^n \mathbf{x}_r^T \mathbf{x}_r \quad (23)$$

Thay (21), (22), (23) vào (20)

$$\begin{aligned} d_{rs}^2 &= \mathbf{x}_r^T \mathbf{x}_r + \mathbf{x}_s^T \mathbf{x}_s - 2\mathbf{x}_r^T \mathbf{x}_s \\ &= \frac{1}{n} \sum_{r=1}^n d_{rs}^2 + \frac{1}{n} \sum_{s=1}^n d_{rs}^2 - \frac{2}{n} \sum_{r=1}^n \mathbf{x}_r^T \mathbf{x}_r - 2\mathbf{x}_r^T \mathbf{x}_s \\ &= \frac{1}{n} \sum_{r=1}^n d_{rs}^2 + \frac{1}{n} \sum_{s=1}^n d_{rs}^2 - \frac{1}{n^2} \sum_{r=1}^n \sum_{s=1}^n d_{rs}^2 - 2\mathbf{x}_r^T \mathbf{x}_s \end{aligned}$$

Như vậy

$$\begin{aligned} b_{rs} &= \mathbf{x}_r^T \mathbf{x}_s \\ &= -\frac{1}{2} \left(d_{rs}^2 - \frac{1}{n} \sum_{r=1}^n d_{rs}^2 - \frac{1}{n} \sum_{s=1}^n d_{rs}^2 + \frac{1}{n^2} \sum_{r=1}^n \sum_{s=1}^n d_{rs}^2 \right) \\ &= a_{rs} - a_{r.} - a_{.s} + a_{..} \end{aligned}$$

Trong đó:

$$\begin{aligned} a_{rs} &= -\frac{1}{2} d_{rs}^2 \\ a_{r.} &= n^{-1} \sum_s a_{rs} \\ a_{.s} &= n^{-1} \sum_r a_{rs} \\ a_{..} &= n^{-2} \sum_r \sum_s a_{rs} \end{aligned}$$

Đặt ma trận \mathbf{A} với $[\mathbf{A}]_{rs} = a_{rs}$. Ta thu được ma trận \mathbf{B} ở dưới dạng:

$$\mathbf{B} = \mathbf{H} \mathbf{A} \mathbf{H} \quad (24)$$

6 Chia tỉ lệ nhiều chiều

Với \mathbf{H} là ma trận nửa xác định dương:

$$\mathbf{H} = \mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}^T$$

Trong đó \mathbf{I} là ma trận đơn vị và $\mathbf{1} = (1, 1, \dots, 1)^T$ là vector có n số 1.

Đẳng thức (24) có thể chứng minh như sau:

Trước tiên ta khai triển:

$$\begin{aligned}\mathbf{H}\mathbf{A}\mathbf{H} &= (\mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}^T)\mathbf{A}(\mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}^T) \\ &= \mathbf{I}\mathbf{A}\mathbf{I} - \mathbf{I}\mathbf{A}n^{-1}\mathbf{1}\mathbf{1}^T - n^{-1}\mathbf{1}\mathbf{1}^T\mathbf{A}\mathbf{I} + n^{-1}\mathbf{1}\mathbf{1}^T\mathbf{A}n^{-1}\mathbf{1}\mathbf{1}^T \\ &= \mathbf{A} - n^{-1}\mathbf{A}\mathbf{1}\mathbf{1}^T - n^{-1}\mathbf{1}\mathbf{1}^T\mathbf{A} + n^{-2}\mathbf{1}\mathbf{1}^T\mathbf{A}\mathbf{1}\mathbf{1}^T \\ &= \mathbf{C}\end{aligned}$$

Ta tính giá trị c_{rs} thông qua 4 số hạng ở vế phải. Tại vị trí hàng r cột s của:

$$\begin{aligned}\mathbf{A}_{rs} &= -\frac{1}{2}d_{rs}^2 \\ (n^{-1}\mathbf{A}\mathbf{1}\mathbf{1}^T)_{rs} &= -\frac{1}{2n} \sum_{s=1}^n d_{rs}^2 \\ (n^{-1}\mathbf{1}\mathbf{1}^T\mathbf{A})_{rs} &= -\frac{1}{2n} \sum_{r=1}^n d_{rs}^2 \\ (n^{-2}\mathbf{1}\mathbf{1}^T\mathbf{A}\mathbf{1}\mathbf{1}^T)_{rs} &= \frac{1}{2n^2} \sum_{r=1}^n \sum_{s=1}^n d_{rs}^2\end{aligned}$$

Từ đó suy ra:

$$\begin{aligned}c_{rs} &= -\frac{1}{2}d_{rs}^2 + \frac{1}{2n} \sum_{r=1}^n d_{rs}^2 + \frac{1}{2n} \sum_{s=1}^n d_{rs}^2 - \frac{1}{2n^2} \sum_{r=1}^n \sum_{s=1}^n d_{rs}^2 \\ &= -\frac{1}{2} \left(d_{rs}^2 - \frac{1}{n} \sum_{r=1}^n d_{rs}^2 - \frac{1}{n} \sum_{s=1}^n d_{rs}^2 + \frac{1}{n^2} \sum_{r=1}^n \sum_{s=1}^n d_{rs}^2 \right) \\ &= b_{rs}\end{aligned}$$

Vậy đẳng thức (24) đã được chứng minh.

Khôi phục lại tọa độ từ ma trận B

Ma trận tích trong \mathbf{B} có thể được viết lại dưới dạng:

$$\mathbf{B} = \mathbf{XX}^T$$

Trong đó $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]^T$ là một ma trận cỡ $n \times p$. Ma trận \mathbf{B} là ma trận đối xứng nên viết được dưới dạng:

$$\mathbf{B} = \mathbf{V}\Lambda\mathbf{V}^T$$

Trong đó $\Lambda = diag(\lambda_1, \lambda_2, \dots, \lambda_n)$ là ma trận đường chéo chứa các giá trị riêng λ của \mathbf{B} và $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n]$ là ma trận chứa các vector riêng trực chuẩn.

Từ đẳng thức $\mathbf{B} = \mathbf{XX}^T$, kèm theo điều kiện \mathbf{B} là ma trận nửa xác định dương thì:

$$\mathbf{X} = \mathbf{V}\Lambda^{\frac{1}{2}} \quad (25)$$

Nếu ma trận \mathbf{B} không là nửa xác định dương thì ta có thể loại bỏ các giá trị riêng âm:

$$\mathbf{X} = \mathbf{V}_1\Lambda_1^{\frac{1}{2}} \quad (26)$$

Với Λ_1 là ma trận chéo chỉ chứa các giá trị riêng không âm và \mathbf{V}_1 là ma trận chứa các vector riêng trực chuẩn tương ứng. Tuy nhiên việc loại bỏ các giá trị riêng âm sẽ khiến cấu hình mà ta thu được có thể trở nên thiếu chính xác hơn.

Số chiều của cấu hình thu được là số hàng của ma trận giá trị riêng Λ hay Λ_1 . Nếu ma trận giá trị riêng này có chứa giá trị riêng 0, thì tương ứng ma trận \mathbf{X} sẽ có cột toàn là 0. Hay nói cách khác số chiều của cấu hình là số các vector riêng khác 0 trong ma trận giá trị riêng.

Như vậy để có một cấu hình với số chiều bất q , ta tạo một ma trận Λ_q chứa q giá trị riêng không âm và ma trận \mathbf{V}_q chứa q vector riêng tương ứng. Tọa độ các điểm trong cấu hình thu được qua công thức:

$$\mathbf{X} = \mathbf{V}_q\Lambda_q^{\frac{1}{2}}$$

Do ma trận \mathbf{H} có giá trị riêng 0 với vector riêng $\mathbf{1}$ nên ma trận \mathbf{B} cũng có giá trị riêng 0. Vậy cấu hình thu được có tối đa $n - 1$ chiều.

Thêm hằng số vào d_{rs} để ma trận \mathbf{B} nửa xác định dương

Như đã thấy ở trên, nếu \mathbf{B} có giá trị riêng âm, các giá trị trong ma trận \mathbf{X} theo công thức (25) sẽ chứa các số phức. Nói cách khác là cấu hình mà ta đang tìm không thuộc trong không gian Euclide. Có nhiều phương pháp thêm hằng số để giúp ma trận \mathbf{B} trong công thức (24) là nửa xác định dương. Trong tài liệu này tác giả trình bày phương pháp khá đơn giản của Francis Cailliez, bạn đọc có thể tìm hiểu thêm tại [6].

Phương pháp của Cailliez là tìm một hằng số c^* nhỏ nhất sao cho mọi phép đo $d^{(c)}$ được định nghĩa:

$$d_{rs}^{(c)} = \begin{cases} d_{rs} + c & (r \neq s) \\ 0 & (r = s) \end{cases} \quad (27a)$$

$$(27b)$$

Đều có một cấu hình đại diện trong không gian Euclide với mọi $c \geq c^*$. Để thuận tiện cho việc biến đổi cũng như chứng minh, ta đặt:

$$\mathbf{B}_d = \mathbf{H}\mathbf{A}_d\mathbf{H}$$

Với giả thiết \mathbf{B}_d không là nửa xác định dương.

Trong đó \mathbf{A}_d chứa các phần tử dưới dạng $a_{rs} = -\frac{1}{2}d_{rs}^2$. Như vậy ta có:

$$\begin{aligned} \mathbf{B}_{d^{(c)}} &= \mathbf{H}\mathbf{A}_{d^{(c)}}\mathbf{H} \\ &= -\frac{1}{2}\mathbf{H} \begin{bmatrix} 0 & (d_{12} + c)^2 & \dots & (d_{1n} + c)^2 \\ (d_{21} + c)^2 & 0 & \dots & (d_{2n} + c)^2 \\ \dots & \dots & \dots & \dots \\ (d_{n1} + c)^2 & (d_{n2} + c)^2 & \dots & 0 \end{bmatrix} \mathbf{H} \\ &= \mathbf{B}_d + 2c\mathbf{B}_{d^{\frac{1}{2}}} - \frac{1}{2}\mathbf{H} \begin{bmatrix} 0 & c^2 & \dots & c^2 \\ c^2 & 0 & \dots & c^2 \\ \dots & \dots & \dots & \dots \\ c^2 & c^2 & \dots & 0 \end{bmatrix} \mathbf{H} \end{aligned}$$

6 Chia tỉ lệ nhiều chiều

Ta biến đổi số hạng cuối cùng trong vế phải của đẳng thức trên như sau:

$$\begin{aligned} -\frac{1}{2}\mathbf{H} \begin{bmatrix} 0 & c^2 & \dots & c^2 \\ c^2 & 0 & \dots & c^2 \\ \dots & \dots & \dots & \dots \\ c^2 & c^2 & \dots & 0 \end{bmatrix} \mathbf{H} &= -\frac{c^2}{2} (\mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}^T) (\mathbf{1}\mathbf{1}^T - \mathbf{I}) \mathbf{H} \\ &= \frac{c^2}{2} (\mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}^T) \mathbf{H} \\ &= \frac{c^2}{2} \mathbf{H}^2 = \frac{c^2}{2} \mathbf{H} \end{aligned}$$

Như vậy ta thu được:

$$\mathbf{B}_{d^{(c)}} = \mathbf{B}_d + 2c\mathbf{B}_{d^{\frac{1}{2}}} + \frac{c^2}{2}\mathbf{H} \quad (28)$$

Đẳng thức (28) giúp ta dễ dàng chứng minh định lý sau:

Định lý 1. *Tồn tại một hằng số c^* sao cho mọi phép đo $d^{(c)}$ được định nghĩa bởi (27a) (27b) đều có một đại diện (cấu hình) trong không gian Euclide với mọi $c \geq c^*$. Ngoài ra cấu hình với phép đo $d^{(c^*)}$ có nhiều nhất ($n - 2$) chiều (n là số quan sát).*

Trước tiên ta chứng minh tồn tại hằng số c^* . Ta muốn phép đo $d^{(c)}$ cho một cấu hình trong không gian Euclide hay với mọi vector cột \mathbf{x} trong không gian đều thoả mãn:

$$\begin{aligned} \mathbf{x}^T \mathbf{B}_{d^{(c)}} \mathbf{x} &\geq 0 \\ \Leftrightarrow \mathbf{x}^T \mathbf{B}_d \mathbf{x} + \mathbf{x}^T 2c \mathbf{B}_{d^{\frac{1}{2}}} \mathbf{x} + \mathbf{x}^T \frac{c^2}{2} \mathbf{H} \mathbf{x} &\geq 0 \end{aligned}$$

Gọi λ_n, μ_n lần lượt là giá trị riêng nhỏ nhất của ma trận \mathbf{B}_d và $\mathbf{B}_{d^{\frac{1}{2}}}$. Do \mathbf{B}_d không là nửa xác định dương nên $\lambda_n < 0$. Gọi Λ_d là ma trận giá trị riêng của \mathbf{B}_d và \mathbf{V}_d là ma trận vector riêng tương ứng. Với mọi \mathbf{x} trong không gian Euclide:

$$\begin{aligned} \mathbf{x}^T \mathbf{B}_d \mathbf{x} &= \mathbf{x}^T \mathbf{V}_d \Lambda_d \mathbf{V}_d^T \mathbf{x} \\ &= (\mathbf{V}_d^T \mathbf{x})^T \Lambda_d (\mathbf{V}_d^T \mathbf{x}) \\ &= \mathbf{u}^T \Lambda_d \mathbf{u} \geq \lambda_n \mathbf{u}^T \mathbf{I} \mathbf{u} \end{aligned}$$

Mà $\lambda_n \mathbf{u}^T \mathbf{I} \mathbf{u} = \lambda_n \mathbf{x}^T \mathbf{V}_d \mathbf{I} \mathbf{V}_d^T \mathbf{x} = \lambda_n \mathbf{x}^T \mathbf{x}$. Do cách định nghĩa ma trận \mathbf{H} , ta lại có:

$$\mathbf{x}^T \mathbf{x} \geq \mathbf{x}^T \mathbf{H} \mathbf{x}$$

Tóm lại ta có bất đẳng thức sau:

$$\mathbf{x}^T \mathbf{B}_d \mathbf{x} \geq \lambda_n \mathbf{x}^T \mathbf{H} \mathbf{x}$$

Tương tự, ta cũng chứng minh được:

$$\mathbf{x}^T \mathbf{B}_{d^{\frac{1}{2}}} \mathbf{x} \geq \mu_n \mathbf{x}^T \mathbf{H} \mathbf{x}$$

Từ đó ta suy ra:

$$\mathbf{x}^T \mathbf{B}_{d(c)} \mathbf{x} \geq \left(\lambda_n + 2c\mu_n + \frac{c^2}{2} \right) \mathbf{x}^T \mathbf{H} \mathbf{x}$$

Vậy $\mathbf{B}_{d(c)}$ là nửa xác định dương, do \mathbf{H} là nửa xác định dương nên ta chỉ cần:

$$\left(\lambda_n + 2c\mu_n + \frac{c^2}{2} \right) \geq 0$$

Do $c > 0$ nên:

$$c \geq -2\mu_n + (4\mu_n^2 - 2\lambda_n)^{\frac{1}{2}}$$

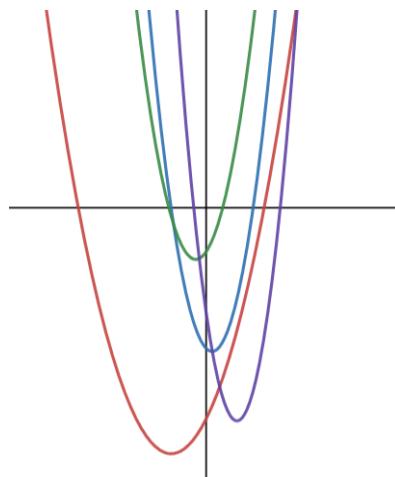
Hay nói cách khác, c^* tồn tại và bị chặn bởi một số dương:

$$c^* \leq -2\mu_n + (4\mu_n^2 - 2\lambda_n)^{\frac{1}{2}} \quad (29)$$

Như vậy, ta có thể thêm hằng số c lớn hơn c^* trong công thức (29). Tuy nhiên ta mới chỉ khẳng định c^* bị chặn và chưa tìm được giá trị nhỏ nhất của c^* .

Xét đẳng thức (28), ta thấy $\mathbf{x}^T \mathbf{B}_{d(c)} \mathbf{x}$ là một hàm của c có đồ thị dạng parabol lồi. Với mỗi \mathbf{x} ta lại được một parabol. Với mỗi parabol, c chỉ cần lớn hơn $\alpha(x)$ (giao điểm bên phải của parabol với trực hoành) thì giá trị tại c sẽ lớn hơn hoặc bằng 0. Ngoài ra parabol giao với trực tung tại điểm có tung độ là $\mathbf{x}^T \mathbf{B}_d \mathbf{x}$.

Xét tập các vector \mathbf{x} làm cho $\mathbf{x}^T \mathbf{B}_d \mathbf{x} < 0$. Khi đó lớp các parabol $\mathbf{x}^T \mathbf{B}_{d(c)} \mathbf{x}$ có dạng:



Như vậy $\alpha(x)$ trong các lớp parabol trên là dương. Kết hợp với (29) ta có thể chọn c^* :

$$c^* = \sup_{\mathbf{x}' \mathbf{B}_d \mathbf{x} < 0} \alpha(\mathbf{x}) = \alpha(\mathbf{x}^*) \quad (30)$$

Từ đó, ta thu được:

$$\begin{aligned} \mathbf{x}^T \mathbf{B}_{d(c)} \mathbf{x} &\geq 0, c \geq c^* \\ (\mathbf{x}^*)^T \mathbf{B}_{d(c^*)} \mathbf{x}^* &= 0 \end{aligned}$$

6 Chia tỉ lệ nhiều chiều

Với mọi $c \geq c^*$ thì phép đo $d^{(c)}$ có một câu hình đại diện trong không gian Euclidean. Phép đo $d^{(c^*)}$ có câu hình tối đa $(n - 2)$ chiều do $\mathbf{B}_d^{(c^*)}$ có hai vector riêng ứng với giá trị riêng 0 là \mathbf{x}^* và $\mathbf{1}$. Vậy định lý 1 được chứng minh.

Giờ ta đi tìm c^* trong công thức (30). Chú ý rằng $\mathbf{1}\mathbf{1}^T\mathbf{x} = k\mathbf{1}$ với số k nào đó:

$$\left(\mathbf{B}_d + 2c^*\mathbf{B}_{d^{\frac{1}{2}}} + \frac{(c^*)^2}{2}\mathbf{H} \right) \mathbf{Hx}^* = 0$$

Đặt $2\mathbf{B}_d\mathbf{Hx}^* = c^*\mathbf{y}$, ta viết lại đẳng thức trên dưới dạng:

$$\mathbf{y} + 4\mathbf{B}_{d^{\frac{1}{2}}}\mathbf{Hx}^* + c^*\mathbf{Hx}^* = 0$$

Từ những đẳng thức trên, rất thú vị khi ta có thể viết được hệ phương trình sau:

$$\begin{pmatrix} 0 & 2\mathbf{B}_d \\ -\mathbf{I} & -4\mathbf{B}_{d^{\frac{1}{2}}} \end{pmatrix} \begin{pmatrix} \mathbf{y} \\ \mathbf{Hx}^* \end{pmatrix} = c^* \begin{pmatrix} \mathbf{y} \\ \mathbf{Hx}^* \end{pmatrix}$$

Như vậy c^* là một giá trị riêng của ma trận:

$$\mathbf{W} = \begin{pmatrix} 0 & 2\mathbf{B}_d \\ -\mathbf{I} & -4\mathbf{B}_{d^{\frac{1}{2}}} \end{pmatrix}$$

Xét a là một giá trị riêng của \mathbf{W} với vector riêng $[\mathbf{z} \quad \mathbf{t}]^T$, ta có $\mathbf{t}^T\mathbf{B}_{d(a)}\mathbf{t} = 0$. Do $c^* = \sup_{\mathbf{x}^T\mathbf{B}_d\mathbf{x} < 0} \alpha(x)$ nên $c^* \geq a$, hay nói cách khác c^* là giá trị riêng thực lớn nhất của ma trận \mathbf{W} .

Lựa chọn câu hình thích hợp

Xét ma trận \mathbf{B} là nửa xác định dương. Từ đẳng thức (25) và \mathbf{B} luôn có giá trị riêng 0, câu hình ta thu được có nhiều nhất $n - 1$ chiều. Nếu số chiều của câu hình lớn thì việc giảm số chiều dữ liệu sẽ ít hiệu quả do ta khó quan sát được các đặc điểm của dữ liệu. Do đó ta thường tìm một hình chiếu của câu hình ban đầu lên không gian q chiều (q thường là 2 hoặc 3). Tọa độ trong không gian q chiều có dạng:

$$\mathbf{X} = \mathbf{V}_q \Lambda_q^{\frac{1}{2}}$$

Với Λ_q là ma trận đường chéo cỡ $q \times q$ chứa q giá trị riêng của ma trận \mathbf{B} , tương ứng \mathbf{V}_q là ma trận cỡ $n \times q$ chứa các vector riêng. Gọi khoảng cách giữa các điểm trong không gian q chiều là d^* . Ta chọn q giá trị riêng và q vector riêng tương ứng sao cho biểu thức dưới đây đạt giá trị nhỏ nhất:

$$\sum_{r=1}^n \sum_{s=1}^n (d_{rs}^2 - (d_{rs}^*)^2) \tag{31}$$

Đặt $\mathbf{B}_q = \mathbf{V}_q \Lambda_q \mathbf{V}'_q$. Chú ý đẳng thức (23):

$$\sum_{r=1}^n \sum_{s=1}^n (d_{rs}^2 - (d_{rs}^*)^2) = n (tr\mathbf{B} - tr\mathbf{B}_q)$$

Vậy (31) đạt giá trị nhỏ nhất nếu $tr\mathbf{B}_q$ lớn nhất, hay Λ_q chứa q giá trị riêng lớn nhất của \mathbf{B} . Để đánh giá độ tốt của câu hình, bạn đọc có thể tham khảo hàm Stress tại 6.2.

Các bước thực hiện của thuật toán chia tỉ lệ cổ điển

Có rất nhiều công đoạn mà tác giả đã giới thiệu ở trên, thuật toán có thể tóm gọn lại thành một số bước sau:

Algorithm 6 Thuật toán chia tỉ lệ cổ điển

- 1: Tính các δ_{rs} .
 - 2: Tìm ma trận $\mathbf{A} = \left[-\frac{1}{2}\delta_{rs}^2 \right]$.
 - 3: Tìm ma trận $\mathbf{B} = \mathbf{H}\mathbf{A}\mathbf{H}$.
 - 4: Tìm giá trị riêng của \mathbf{B} và các vector riêng tương ứng. Nếu \mathbf{B} không là nửa xác định dương có thể dùng kỹ thuật thêm hằng số và quay lại bước 2.
 - 5: Chọn không gian tốt nhất với số chiều p mong muốn.
 - 6: Tính toán tọa độ các điểm trong không gian p chiều và vẽ biểu đồ.
-

Với cách làm này, khoảng cách giữa điểm r và s trong cấu hình thu được từ ma trận \mathbf{B} là δ_{rs} .

6.2 Dùng thứ hạng trong chia tỉ lệ nhiều chiều và cách tiếp cận của Kruskal

Cho tập các vật thuộc tập \mathbb{O} và phép đo độ khác nhau giữa hai vật r và s thuộc \mathbb{O} là δ_{rs} . Gọi ϕ là ánh xạ từ tập \mathbb{O} vào tập \mathbf{X} với \mathbf{X} là tập con của không gian được sử dụng để biểu diễn các vật. Vật r, s tương đương với hai điểm x_r và x_s trong \mathbf{X} với khoảng cách giữa hai điểm là $d_{x_r x_s}$. Ta còn định nghĩa một hàm đo độ lệch là \hat{d} trên tập $\mathbb{O} \times \mathbb{O}$ dùng để đo sự khớp của $d_{x_r x_s}$ với độ khác nhau δ_{rs} . Mục tiêu của lớp phương pháp *Nonmetric Multidimensional Scaling* là tìm ánh xạ ϕ sao cho $d_{x_r x_s}$ xấp xỉ bằng với \hat{d}_{rs} , và thường được tìm bởi một hàm mất mát nào đó. Các điểm thuộc \mathbf{X} cùng với các khoảng cách giữa các điểm tạo nên một cấu hình.

Tập \mathbf{X} có thể là tập con của \mathbb{R}^2 với khoảng cách Euclidean, hoặc là tập con của \mathbb{R}^3 với khoảng cách Minkowski,... Với phép đo độ khác nhau δ xác định và các phương pháp để tính độ lệch \hat{d} , vấn đề Nonmetric MDS trở thành vấn đề tìm thuật toán làm cực tiểu hàm mất mát.

Ngoài ra, ta chỉ sử dụng thứ hạng của các phép đo độ khác nhau δ_{rs} cho phương pháp này. Đó cũng là lý giải của từ "Nonmetric".

Thông thường, khoảng cách giữa các điểm trong \mathbf{X} là khoảng cách Minkowski. Với hai điểm r và s trong \mathbf{X} có p chiều thì khoảng cách giữa chúng tính bởi:

$$d_{rs} = \left[\sum_{i=1}^p |x_{ri} - x_{si}|^\lambda \right]^{\frac{1}{\lambda}} \quad (\lambda > 0)$$

Với điểm r có tọa độ là $\mathbf{x}_r = (x_{r1}, \dots, x_{rp})^T$.

Tập các độ lệch $\{\hat{d}_{rs}\}$ được xem như là hàm của $\{d_{rs}\}$:

$$\hat{d}_{rs} = f(d_{rs})$$

Với f là hàm đơn điệu sao cho:

$$\delta_{rs} < \delta_{tu} \Rightarrow \hat{d}_{rs} \leq \hat{d}_{tu} \quad (32)$$

Trong phần này, tác giả trình bày phương pháp của Joseph Bernard Kruskal. Độc giả có thể đọc tìm hai bài báo [7] và [8] để tìm hiểu chi tiết hơn.

6 Chia tách lôgic nhiều chiều

Định nghĩa hàm matsu matsu

Sử dụng các ký hiệu mà tác giả đã nêu trên, hàm matsu matsu S (viết tắt của Stress) được Kruskal định nghĩa như sau:

$$S = \sqrt{\frac{S^*}{T^*}} \quad (33)$$

Trong đó $S^* = \sum_{r < s} (d_{rs} - \hat{d}_{rs})^2$ và $T^* = \sum_{r < s} d_{rs}^2$. Với cách định nghĩa này, tập $\{\delta_{rs}\}$ "nhảy vào" S thông qua điều kiện (32).

Nhờ có S , ta có thể đánh giá độ tốt của cấu hình như sau:

Bảng đánh giá độ phù hợp:

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

Cách xác định \hat{d}_{rs}

Hiển nhiên cho một tập khoảng cách trong cấu hình $\{d_{rs}\}$, cách xác định tập $\{\hat{d}_{rs}\}$ sẽ cho ra các giá trị của S^* khác nhau. Điều này sẽ làm ảnh hưởng đến quá trình tìm cực tiểu của S và dẫn đến nhiều cấu hình "tốt" theo các cách định nghĩa tập $\{\hat{d}_{rs}\}$. Để tránh khỏi vấn đề này cũng như tìm được cấu hình ưng ý nhất, ta chọn tập $\{\hat{d}_{rs}\}$ sao cho S^* đạt cực tiểu với tập $\{d_{rs}\}$ cho trước.

Để thuận tiện, ta gán nhãn lại tập $\{\delta_{rs}\}$ thành tập $\{\delta_i : i = 1, \dots, N\}$ được sắp theo thứ tự tăng dần. Tương tự ta có tập các khoảng cách $\{d_i : i = 1, \dots, N\}$ với d_i tương đương với δ_i .

Ví Dụ

Ta có bốn vật với độ khác nhau giữa các vật như sau:

$$\delta_{12} = 2.1, \delta_{13} = 3.0, \delta_{14} = 2.4, \delta_{23} = 1.7, \delta_{24} = 3.9, \delta_{34} = 3.2$$

Và bốn điểm trong cấu hình đại diện với khoảng cách:

$$d_{12} = 3.3, d_{13} = 4.5, d_{14} = 5.7, d_{23} = 3.3, d_{24} = 4.3, d_{34} = 1.3$$

Ta gán nhãn lại $\{d_{rs}\}$ và $\{\delta_{rs}\}$:

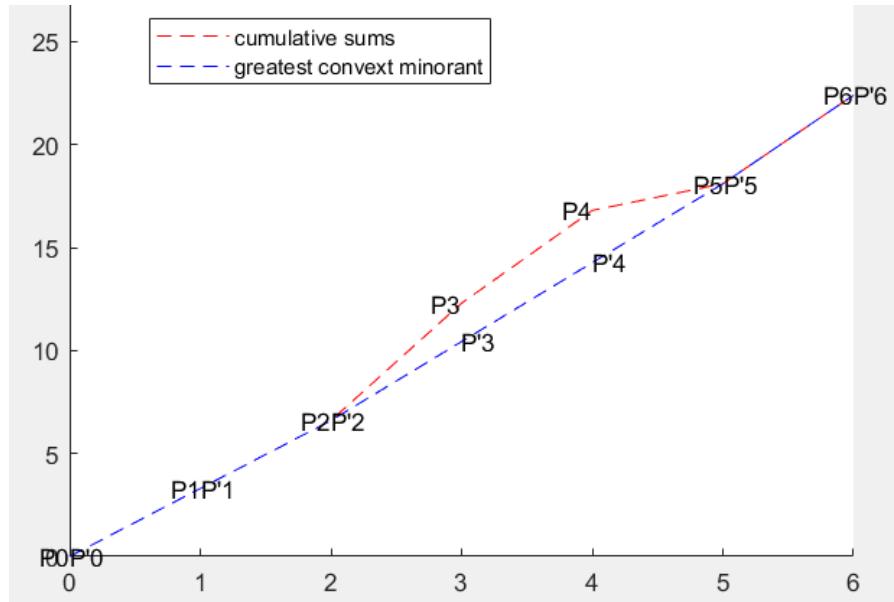
$$\delta_1 = 1.7, \delta_2 = 2.1, \delta_3 = 2.4, \delta_4 = 3.0, \delta_5 = 3.2, \delta_6 = 3.9$$

$$d_1 = 3.3, d_2 = 3.3, d_3 = 5.7, d_4 = 4.5, d_5 = 1.3, d_6 = 4.3$$

Cực tiểu hóa S tương đương với việc cực tiểu hóa $S^* = \sum_i (d_i - \hat{d}_i)^2$. Gọi D_i là tổng tích lũy của d_i :

$$D_i = \sum_{j=1}^i d_j \quad (i = 1, \dots, N)$$

Ta vẽ các điểm có tọa độ $(0, 0)$, (i, D_i) và nối các điểm liên tiếp trên đồ thị và gán nhãn các điểm là P_0, \dots, P_N . Độ dốc của đoạn thẳng nối P_{i-1} với P_i là d_i . Hàm lồi tốt nhất (The greatest convex minorant) của các tổng tích lũy là cận trên bé nhất của tất cả các hàm lồi có đồ thị nằm dưới đồ thị của các tổng tích lũy. Hình sau minh họa cho ví dụ trên:



Hình 26: Đồ thị của các tổng tích lũy và hàm lồi tốt nhất

Ta có các điểm P'_0, \dots, P'_N là các điểm nằm trên hàm lồi tốt nhất với tọa độ là $(0, 0)$ và (i, D'_i) . Ta xác định $\{\hat{d}_{rs}\}$ như sau:

$$\begin{aligned}\hat{d}_1 &= D'_1 \\ \hat{d}_i &= D'_i - D'_{i-1}\end{aligned}$$

Ngoài ra, do hàm các tổng tích lũy là đồng biến và tính chất của hàm lồi tốt nhất, nếu $D'_i < D_i$ thì $\hat{d}_i = \hat{d}_{i+1}$.

Giờ ta đi chứng minh cách xác định tập $\{\hat{d}_{rs}\}$ như bên trên sẽ làm cực tiểu hóa S^* với tập $\{d_{rs}\}$ cố định. Giả sử ta có tập đo độ lệch khác là $\{d_{rs}^*\}$ cũng thỏa mãn điều kiện (32). Ta phải chỉ ra rằng:

$$\sum_{i=1}^N (d_i - d_{rs}^*)^2 \geq \sum_{i=1}^N (d_i - \hat{d}_i)^2 \quad (34)$$

Đặt

$$\begin{aligned}D_i^* &= \sum_{j=1}^i d_j^* \\ D'_i &= \sum_{j=1}^i \hat{d}_j\end{aligned}$$

6 Chia tỉ lệ nhiều chiều

Công thức Abel:

$$\sum_{i=1}^N a_i b_i = \sum_{i=1}^{N-1} A_i (b_i - b_{i+1}) + A_N b_N$$

Trong đó $A_i = \sum_{j=1}^i a_j$.

Xét:

$$\begin{aligned} \sum_{i=1}^N (d_i - d_i^*)^2 &= \sum_{i=1}^N \left[(d_i - \hat{d}_i) + (\hat{d}_i - d_i^*) \right]^2 \\ &= \sum_{i=1}^N (d_i - \hat{d}_i)^2 + \sum_{i=1}^N (\hat{d}_i - d_i^*)^2 + 2 \sum_{i=1}^N (d_i - \hat{d}_i)(\hat{d}_i - d_i^*) \end{aligned}$$

Áp dụng công thức Abel cho số hạng cuối ở vế phải:

$$\begin{aligned} &\sum_{i=1}^N (d_i - \hat{d}_i)(\hat{d}_i - d_i^*) \\ &= \sum_{i=1}^{N-1} (D_i - D'_i)(\hat{d}_i - \hat{d}_{i+1}) - \sum_{i=1}^{N-1} (D_i - D'_i)(d_i^* - d_{i+1}^*) + (D_N - D'_N)(\hat{d}_N - d_N^*) \end{aligned}$$

Có $D_N - D'_N = 0$ do điểm cuối của hàm lồi tốt nhất và P_N là trùng nhau. Giờ ta xét $(D_i - D'_i)(\hat{d}_i - \hat{d}_{i+1})$. Nếu $D_i = D'_i$ thì ta có số thang thứ i bằng 0. Trường hợp còn lại là $D_i > D'_i$, lúc này $\hat{d}_i = \hat{d}_{i+1}$ nên số thang thứ i cũng bằng 0. Hay nói cách khác là $\sum_{i=1}^{N-1} (D_i - D'_i)(\hat{d}_i - \hat{d}_{i+1}) = 0$. Ta có $d_i^* \leq d_{i+1}^*$ nên $-\sum_{i=1}^{N-1} (D_i - D'_i)(d_i^* - d_{i+1}^*)$ là một số dương. Tóm lại:

$$\begin{aligned} \sum_{i=1}^N (d_i - d_i^*)^2 &\geq \sum_{i=1}^N (d_i - \hat{d}_i)^2 + \sum_{i=1}^N (\hat{d}_i - d_i^*)^2 \\ \iff \sum_{i=1}^N (d_i - d_i^*)^2 &\geq \sum_{i=1}^N (d_i - \hat{d}_i)^2 \end{aligned}$$

Vậy bất đẳng thức (34) được chứng minh. Quay lại ví dụ trên, ta xác định:

$$\hat{d}_1 = \hat{d}_2 = 3.3, \hat{d}_3 = \hat{d}_4 = \hat{d}_5 = \frac{23}{6}, \hat{d}_6 = 4.3$$

Lúc này $S = 0.33$.

Cấu hình làm cực tiểu Stress

Giờ ta muốn tìm một cấu hình trong không gian p chiều với khoảng cách Minkowski $\lambda > 0$. Ta có định giá trị của p và λ .

Tất cả các điểm trong cấu hình đều có thể miêu tả như một vector (một điểm) trong không gian np chiều (còn gọi là không gian cấu hình) với tọa độ x_{il} trong đó $i = 1$ tới n và $l = 1$ đến p là tất cả các tọa độ của các điểm trong cấu hình.

$$\mathbf{x} = (x_{11}, \dots, x_{1p}, x_{21}, \dots, x_{n1}, \dots, x_{np})$$

Với mỗi điểm trong không gian cầu hình, hay với mỗi cầu hình, lại cho một giá trị của S . Nói cách khác S là một hàm:

$$S = S(x_{11}, \dots, x_{1p}, x_{21}, \dots, x_{n1}, \dots, x_{np})$$

Vấn đề bây giờ là tìm một điểm trong không gian cầu hình làm cực tiểu hóa hàm S , hay nói cách khác là tìm cực tiểu của hàm số nhiều biến. Ta sẽ dùng phương pháp "method of steepest descent" hay "method of gradients". Ta bắt đầu từ một điểm, di chuyển nó một chút theo hướng ngược với gradient tại đó (hướng giảm nhanh nhất). Gradient tại một điểm là một vector:

$$\left(\frac{\partial S}{\partial x_{11}}, \dots, \frac{\partial S}{\partial x_{1p}}, \dots, \frac{\partial S}{\partial x_{np}} \right)$$

Vector trên luôn dương, đi theo vector đối sẽ tới điểm cực tiểu. Lặp lại quá trình trên, sẽ đến một điểm mà vector gradient tại đó bằng vector 0 (hoặc gần bằng), đó là điểm làm cực tiểu S hay là cầu hình cần tìm.

Cách làm trên sẽ cho ta một điểm cực tiểu địa phương trong không gian cầu hình, chưa chắc là cực tiểu toàn cục. Để khắc phục điều này, ta có thể bắt đầu từ nhiều điểm xuất phát và chọn điểm cực tiểu nhỏ nhất rồi hy vọng nó là cực tiểu toàn cục. Tất nhiên Stress đủ nhỏ thì ta vẫn có một cầu hình đủ tốt mà không cần bận tâm liệu S đã nhỏ nhất chưa.

Điểm bắt đầu có rất nhiều cách lựa chọn, như tọa độ của điểm tuân theo phân phối đều liên tục trong đoạn $[-1; 1]$ hay tuân theo phân phối Poisson,...

Gọi g là gradient tại điểm \mathbf{x} trong không gian cầu hình, α là hệ số nhảy. Cầu hình tiếp theo được tìm thông qua công thức:

$$\mathbf{x}_{i+1} = \mathbf{x}_i - \frac{\alpha}{\text{mag}(g)} g \quad (35)$$

Trong đó:

$$\text{mag}(g) = \sqrt{\frac{\sum_{r,s} g_{rs}^2}{\sum_{r,s} x_{rs}^2}}$$

Giá trị bắt đầu của $\alpha = 0.2$ và thay đổi sau mỗi lần lặp. Cụ thể:

$$\alpha_{\text{present}} = \alpha_{\text{previous}} \cdot (\text{angle factor}) \cdot (\text{relaxation factor}) \cdot (\text{good luck factor})$$

$$\theta = \frac{\sum_{r,s} g_{rs} g_{rs}''}{\sqrt{\sum_{r,s} g_{rs}^2} \sqrt{\sum_{r,s} g_{rs}''^2}} \quad (g'' \text{ là gradient trước đó, } g \text{ là gradient hiện tại})$$

$$\text{angle factor} = 4.0 \cos^3 \theta$$

$$\text{relaxation factor} = \frac{1.3}{1 + (5\text{step ratio})^5}$$

$$5 \text{ step ratio} = \min \left[1, \left(\frac{\text{present stress}}{\text{stress 5 iterations ago}} \right) \right]$$

$$\text{good luck factor} = \min \left[1, \frac{\text{present stress}}{\text{previous stress}} \right]$$

Cách chọn α như trên dựa trên kinh nghiệm của Kruskal và không có bằng chứng rằng cách chọn như trên là tối ưu.

6 Chia tỉ lệ nhiều chiều

Giờ ta tính đạo hàm riêng của S theo x_{ui} bất kỳ:

$$\begin{aligned}\frac{\partial S}{\partial x_{ui}} &= \frac{1}{2} \sqrt{\frac{T^*}{S^*} \left(T^* \frac{\partial S^*}{\partial x_{ui}} - S^* \frac{\partial T^*}{\partial x_{ui}} \right)} \\ &= \frac{1}{2} S \left(\frac{1}{S^*} \frac{\partial S^*}{\partial x_{ui}} - \frac{1}{T^*} \frac{\partial T^*}{\partial x_{ui}} \right) \\ \frac{\partial S^*}{\partial x_{ui}} &= 2 \sum_{r < s} (d_{rs} - \hat{d}_{rs}) \frac{\partial d_{rs}}{\partial x_{ui}} \\ \frac{\partial T^*}{\partial x_{ui}} &= 2 \sum_{r < s} d_{rs} \frac{\partial d_{rs}}{\partial x_{ui}}\end{aligned}$$

Với khoảng cách Minkowski:

$$\frac{\partial d_{rs}}{\partial x_{ui}} = d_{rs}^{1-\lambda} (x_{ri} - x_{si})^{\lambda-1} (\beta^{ru} - \beta^{su}) \operatorname{sig}(x_{ri} - x_{si})$$

Với:

$$\beta^{rs} = \begin{cases} 0 & (r \neq s) \\ 1 & (r = s) \end{cases} \quad (36a)$$

$$(36b)$$

Tóm lại:

$$\frac{\partial S}{\partial x_{ui}} = S \sum_{r < s} (\delta^{ru} - \delta^{su}) \left[\frac{d_{rs} - \hat{d}_{rs}}{S^*} - \frac{d_{rs}}{T^*} \right] \frac{|x_{ri} - x_{si}|^{\lambda-1}}{d_{rs}^{\lambda-1}} \operatorname{sig}(x_{ri} - x_{si}) \quad (37)$$

Kỹ thuật lắp của Kruskal

Ta gói gọn kỹ thuật qua các bước của thuật toán sau:

Algorithm 7 Kỹ thuật lắp của Kruskal

- 1: Chọn cấu hình ban đầu.
 - 2: Chuẩn hóa cấu hình sao cho trọng tâm tại điểm gốc và bình phương khoảng cách trung bình tới điểm gốc là 1. Ta làm vậy do giá trị S không đổi bởi phép tịnh tiến và co dãn. Việc lắp có thể dẫn đến cấu hình bị nở ra quá lớn.
 - 3: Tìm tập khoảng cách (d_{rs}) .
 - 4: Tìm tập độ lệch $\{\hat{d}_{rs}\}$.
 - 5: Tìm gradient g tại điểm hiện tại. Dừng lắp nếu g đủ nhỏ hoặc bằng 0
 - 6: Tính hệ số nhảy α .
 - 7: Tính cấu hình mới theo công thức (35).
 - 8: Quay lại bước 2.
-

6.3 Ví dụ

Ta có dữ liệu về khoảng cách theo đường chim bay giữa các địa điểm sau:

6 Chia tỉ lệ nhiều chiều

	Hà Nội	Hà Đông	Hòa Bình	Mai Châu	Mộc Châu	Sơn La	Tuần Giáo	Điện Biên Phủ	Mường Lay	Lai Châu	Sa Pa	Lào Cai	Yên Bái	Vĩnh Yên	Việt Trì
Hà Nội	0														
Hà Đông	12	0													
Hòa Bình	82	64	0												
Mai Châu	147	129	65	0											
Mộc Châu	206	189	124	60	0										
Sơn La	328	311	246	191	122	0									
Tuần Giáo	406	389	324	269	200	78	0								
Điện Biên Phủ	478	461	396	341	272	150	72	0							
Mường Lay	492	475	410	355	286	164	86	93	0						
Lai Châu	406	413	356	398	329	207	182	189	96	0					
Sa Pa	361	360	361	403	334	212	188	256	163	67	0				
Lào Cai	329	328	393	435	366	244	220	288	195	99	32	0			
Yên Bái	171	170	155	220	181	215	283	365	371	275	208	176	0		
Vĩnh Yên	55	54	111	176	224	258	336	408	422	373	306	274	116	0	
Việt Trì	80	79	86	151	199	233	311	383	397	348	281	249	91	25	0

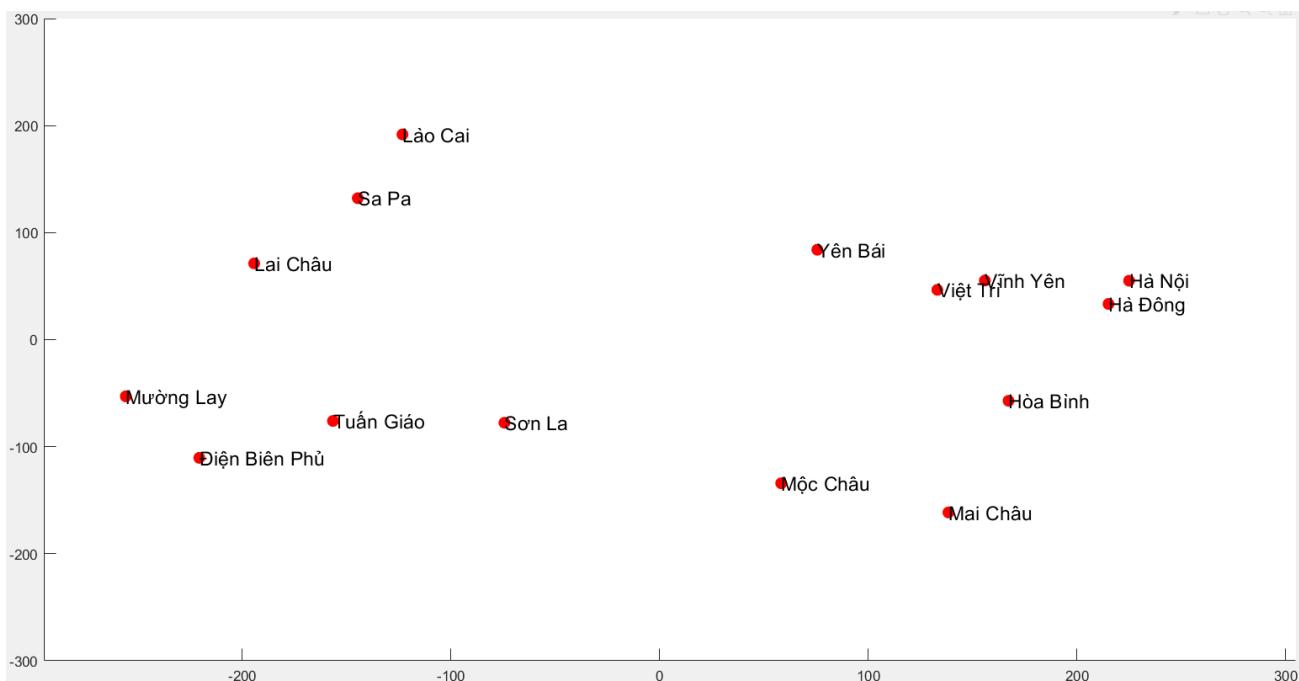
Hình 27: Khoảng cách giữa các tỉnh thành phố

Các giá trị riêng của ma trận B:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
4.1313e+05	1.4984e+05	-5.3889e+04	2.1210e+04	-1.8679e+04	9.5233e+03	7.8541e+03	-6.0389e+03	-5.5666e+03	4.4243e+03	-2.2704e+03	-264.6756	-5.1401e-12	172.1581	315.9152

Hình 28: Giá trị riêng của ma trận B

Ta chọn ra hai giá trị riêng lớn nhất để xây dựng cấu hình:



Hình 29: Vị trí tương đối các tỉnh thành phố với cách làm cổ điển

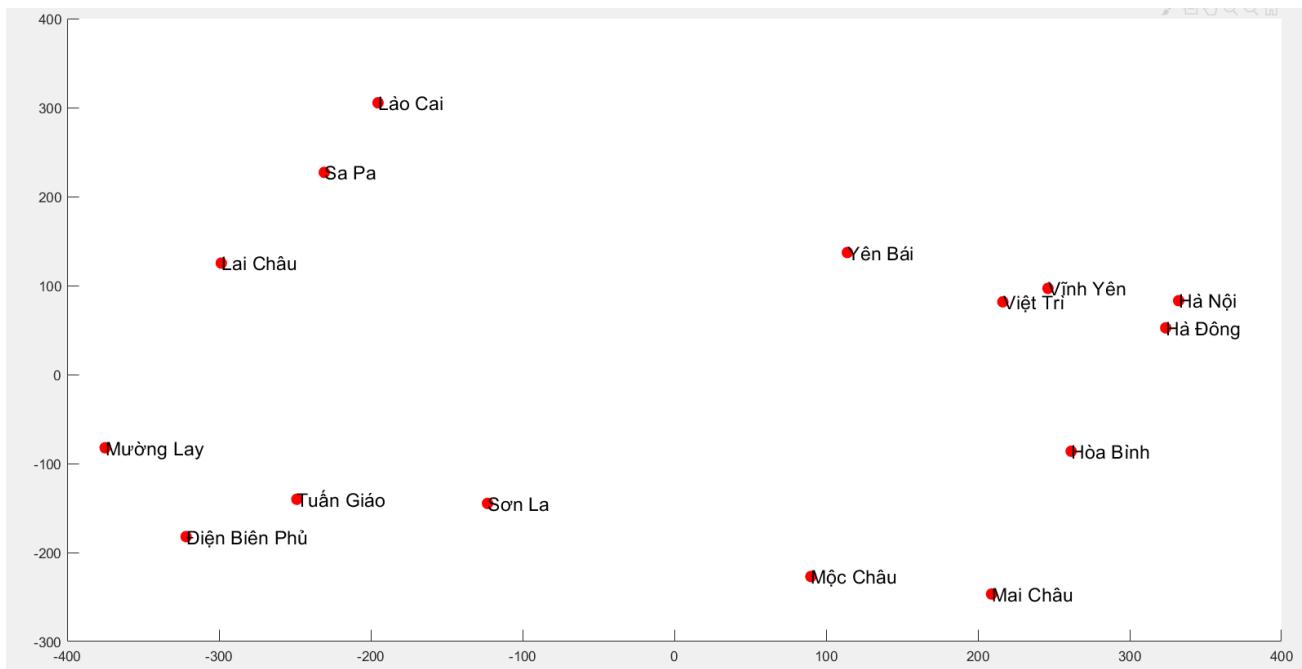
Để ý thấy ma trận B không phải nửa xác định dương, ta thêm hằng số $c^* = 287.2911$. Các giá trị riêng mới như sau:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
9.5724e+05	4.0540e+05	1.2244e+05	1.0899e+05	8.0592e+04	2.6456e+04	7.1001e+04	6.2016e+04	5.7752e+04	4.2351e+04	4.5564e+04	4.7445e+04	4.7027e+04	4.1003e-10	1.6457e-11

Hình 30: Giá trị riêng mới của B sau khi thêm hằng số

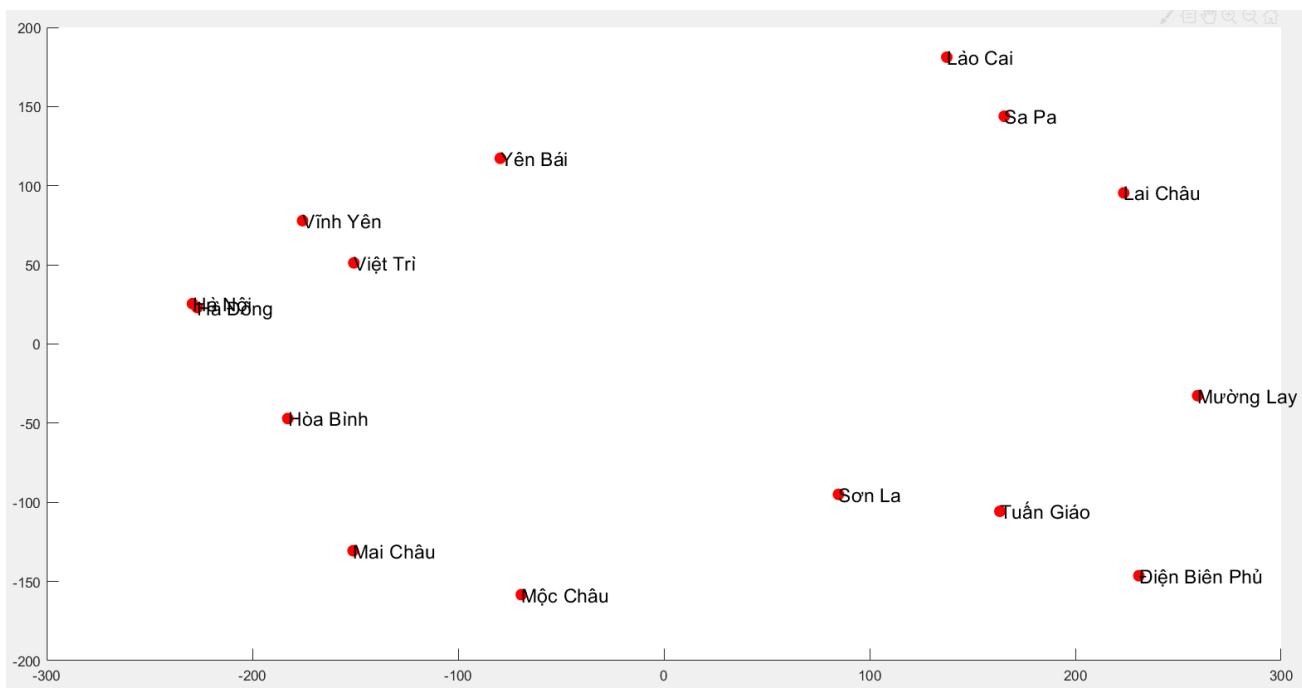
Lại chọn ra hai giá trị riêng lớn nhất và các vector riêng tương ứng, ta được cấu hình:

6 Chia tỉ lệ nhiều chiều



Hình 31: Cấu hình mới với B nửa xác định dương

Nếu dùng thuật toán của Kruskal để tìm cấu hình trong không gian hai chiều, ta thu được cấu hình như sau:



Hình 32: Cấu hình với thuật toán Kruskal

Cả ba cấu hình trên nhìn chung đều tốt với *Stress* của mỗi cấu hình đều nhỏ hơn 5%. Các cấu hình có thể sai lệch với thực tế, tuy nhiên vẫn cho ta thấy được các địa điểm gần nhau như thế nào, cũng như vị trí tương đối giữa các địa điểm.

6.4 Tổng kết

Như vậy qua hai phần kiến thức vừa nêu, tác giả đã trình bày hai phương pháp đơn giản nhất thuộc hai lớp phương pháp lớn. Nhìn chung mỗi phương pháp đều có ưu và nhược điểm riêng. Với cách làm cổ điển, cấu hình tìm được khá dễ dàng thông qua các phép nhân ma trận và tìm giá trị cũng như vector riêng. Tuy nhiên chỉ thu được một cấu hình và có thể cấu hình đó không tốt. Với cách làm của Kruskal, việc lập trình thuật toán cũng như tìm kiếm cấu hình sẽ phức tạp hơn do các công thức phải tính toán nhiều, khi có nhiều quan sát thì quá trình tìm cực tiểu mất nhiều thời gian. Tuy nhiên ta lại thu được nhiều cấu hình hơn nên khả năng tìm được cấu hình ưng ý cũng cao hơn. Cuối cùng, các kiến thức mà tác giả giới thiệu đều dừng lại ở mức cơ bản và lấy từ các bài báo đã lâu, độc giả có thể đọc các bài báo mà tác giả đã nêu trên để hiểu rõ hơn cũng như các bài báo mới hơn để có thêm những cải tiến của các phương pháp. Ngoài ra, chia tỉ lệ nhiều chiều còn rất nhiều lớp phương pháp khác, bạn đọc có thể tìm hiểu tại [9].

Tài liệu

- [1] R. Johnson and D. Wichern, *Applied multivariate statistical analysis*. Pearson Education, 2007, vol. 6.
- [2] D.Wishart, “An algorithm for hierarchical classifications,” *Biometrics*, vol. 25, no. 1, 1969.
- [3] G.N.Lance and W.T.Williams, “A general theory of classificatory sorting strategies: 2. clustering systems,” *The Computer Journal*, 1967.
- [4] T. Zhang, R. Ramakrishnan, and M. Livny, “Birch: an efficient data clustering method for very large databases,” in *SIGMOD '96*, 1996.
- [5] D. Arthur and S. Vassilvitskii, “k-means++: The advantages of careful seeding,” *Society for Industrial and Applied Mathematics, United States*, 2007.
- [6] F. Cailliez, “The analytical solution of the additive constant problem,” *Psychometrika*, vol. 48, no. 2, 1983.
- [7] J. B. Kruskal, “Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis,” *Psychometrika*, vol. 29, no. 1, 1964.
- [8] ——, “Non-metric multidimensional scaling: A numerical method.” *Psychometrika*, vol. 29, no. 1, 1964.
- [9] T. F.Cox and M. A.A.Cox, *Multidimensional Scaling*. Chapman & Hall/CRC, 2001.