

# Học máy không giám sát

Nguyễn Thanh Tùng  
Khoa Công nghệ thông tin – Đại học Thủ Đức  
[tungnt@tlu.edu.vn](mailto:tungnt@tlu.edu.vn)

Website môn học: <https://sites.google.com/a/wru.vn/cse445Fall2017>

Bài giảng có sử dụng hình vẽ trong cuốn sách “An Introduction to Statistical Learning with Applications in R” với sự  
cho phép của tác giả, có sử dụng slides các khóa học CME250 của ĐH Stanford và IOM530 của ĐH Southern California



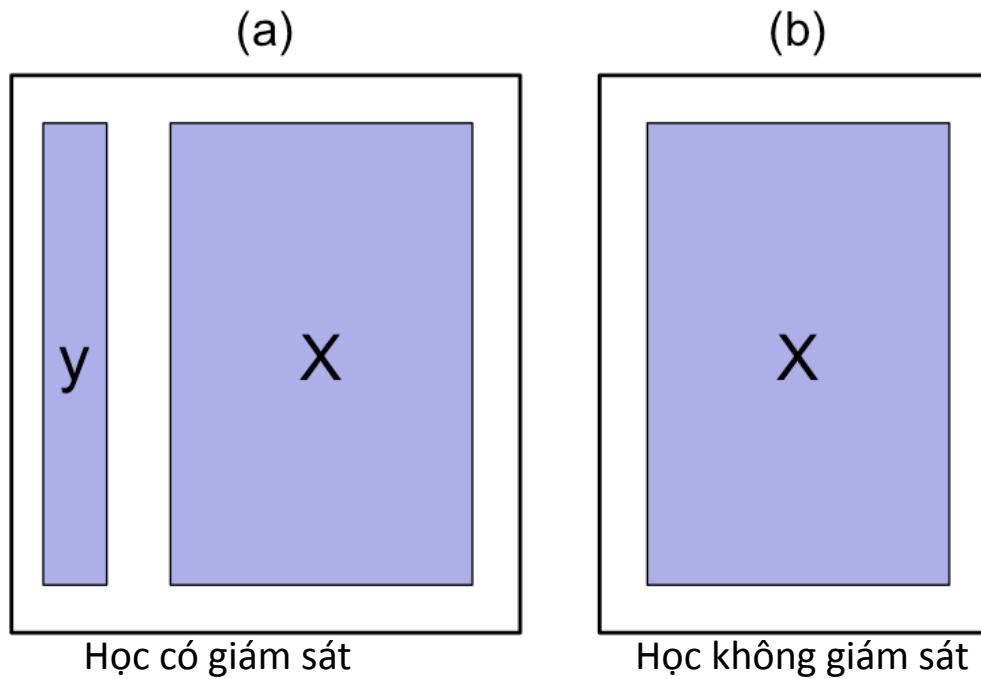
# Học máy không giám sát

- *Học không giám sát*: tập các công cụ thống kê xử lý dữ liệu chỉ có biến đầu vào, không có biến đích
  - Ta chỉ có X's mà không có các nhãn Y
  - Mục tiêu: phát hiện các mẫu/các đặc tính của dữ liệu
    - vd. trực quan hóa hoặc diễn giải dữ liệu nhiều chiều

# Học có giám sát vs. không giám sát

Học máy có giám sát: cả X và Y đều đã biết

Học máy không giám sát: chỉ biết X



# Học không giám sát

- Ví dụ ứng dụng:
    - Biết các mô ung thư của *n* bệnh nhân bị ung thư vú, cần xác định các nhóm nhỏ (subtypes) chưa biết gây nên ung thư vú
    - Các thí nghiệm về biểu diễn Gen chứa hàng ngàn biến

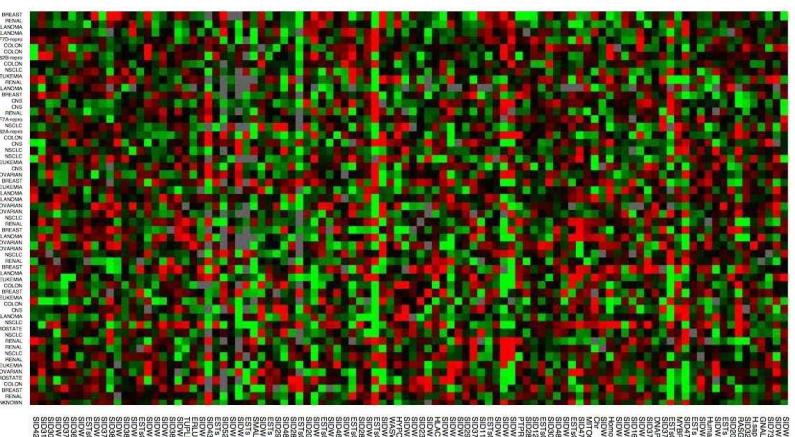


Figure 1.3, ESL



# Học không giám sát

- Ví dụ ứng dụng:

- Cho một tập các tài liệu văn bản, cần xác định tập các tài liệu có chung chủ đề như thể thao, chính trị, ca nhạc...
- Cho các ảnh khuôn mặt có số chiều cao, tìm một biểu diễn đơn giản/thu gọn của các ảnh này để đưa vào bộ phân lớp nhận dạng khuôn mặt

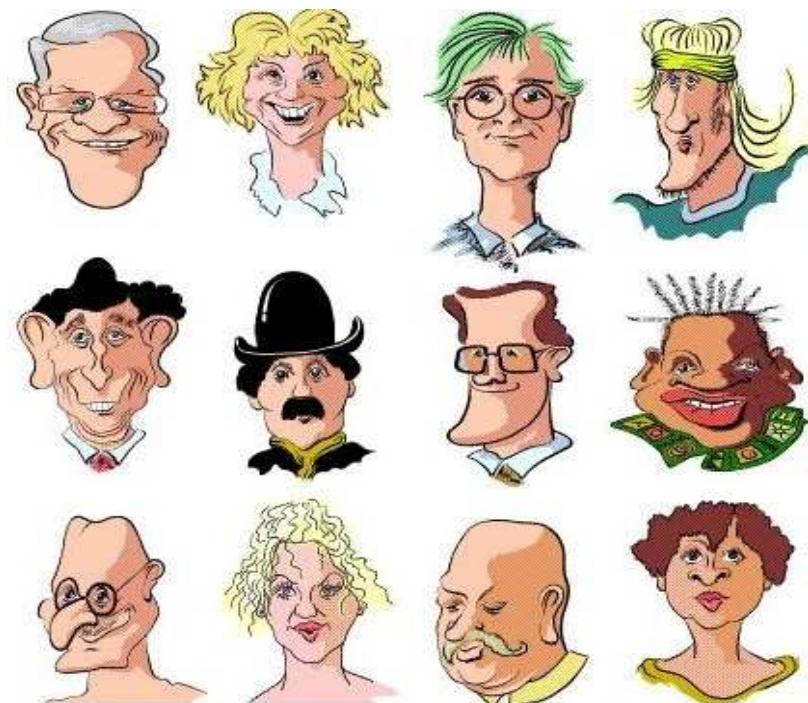


(AT&T Laboratories  
Cambridge)

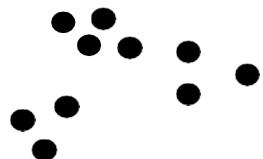
# Học không giám sát

- Tại sao học không giám sát luôn thách thức lớn?
  - Phân tích khám phá dữ liệu (Exploratory data analysis) – mục tiêu không được định nghĩa rõ ràng
  - Khó đánh giá hiệu năng – không biết được đáp án đúng (“right answer” unknown)
  - Xử lý dữ liệu với số chiều lớn

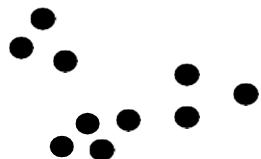
# Sắp xếp các đối tượng vào các nhóm



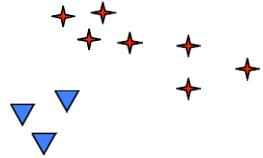
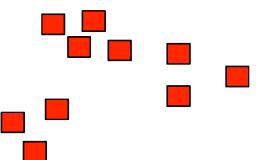
# Có bao nhiêu clusters



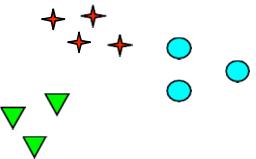
Có bao nhiêu clusters?



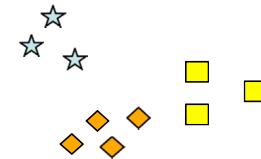
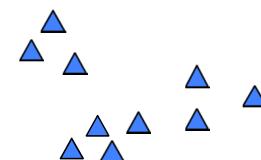
2 clusters



4 clusters



6 clusters



Tuỳ thuộc vào “resolution” !



# Học không giám sát

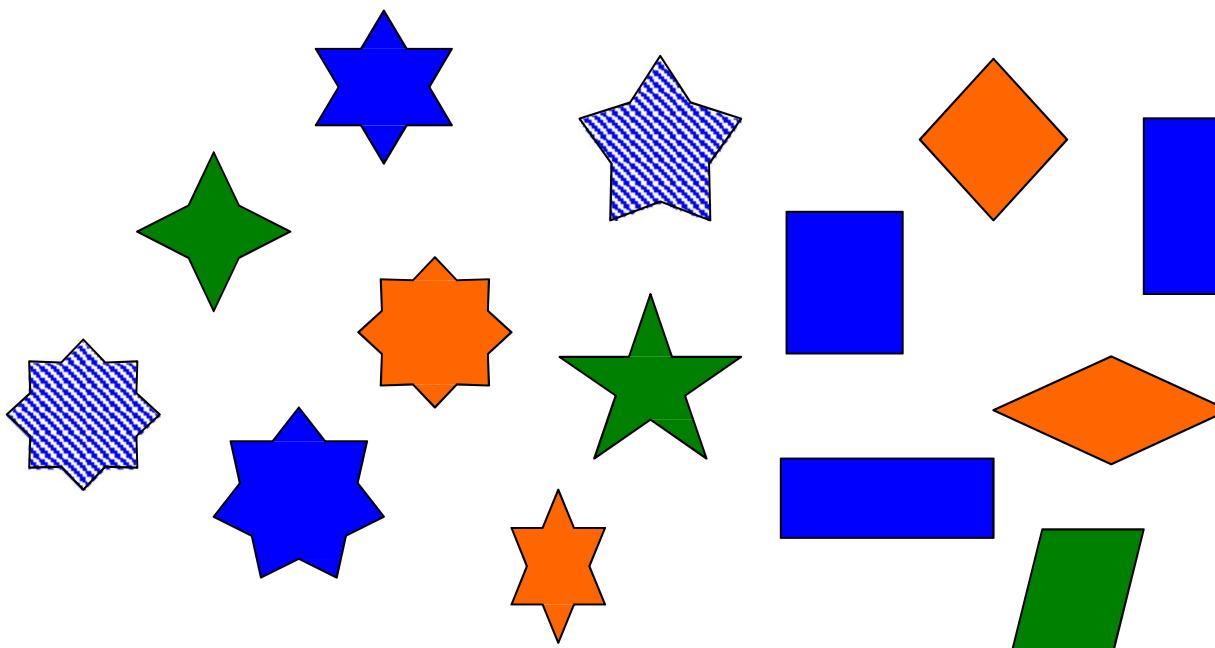
- Hai cách tiếp cận:
  - *Phân tích cụm (Cluster analysis)*
    - Xác định các nhóm mẫu đồng nhất (có các đặc tính chung)
  - *Giảm chiều dữ liệu (Dimensionality Reduction)*
    - Tìm cách biểu diễn với số chiều thấp hơn dựa trên tính chất và trực quan hóa dữ liệu

# Phân tích cụm & K-means

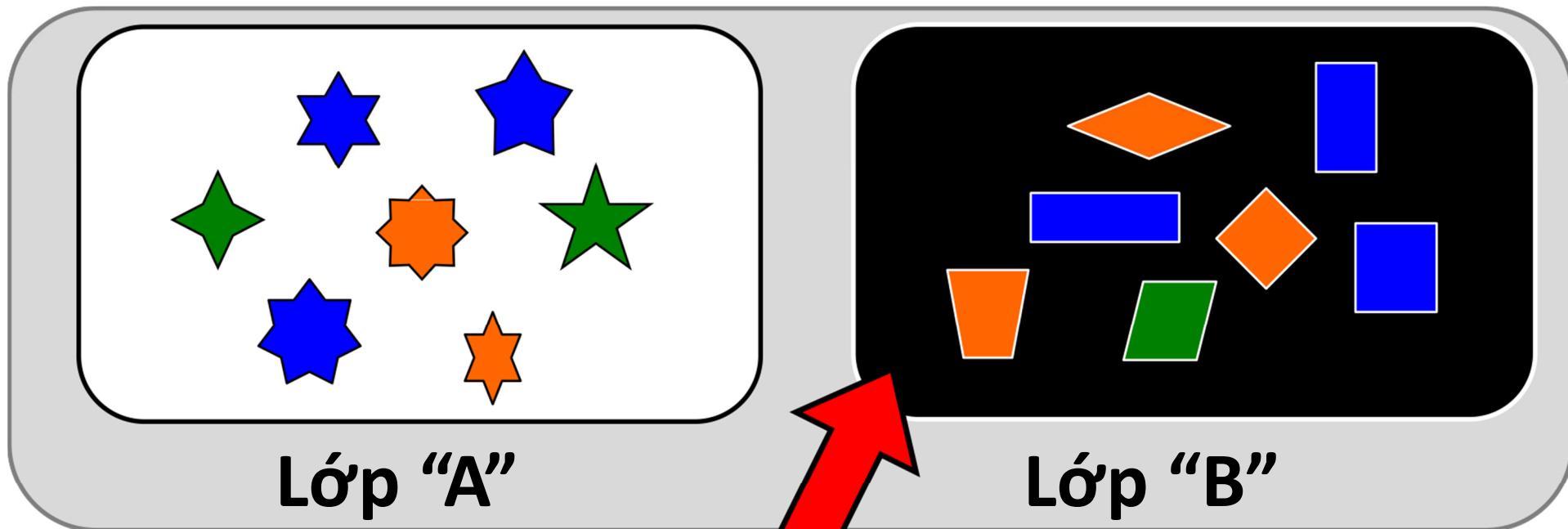
# Phân cụm

- *Phân cụm*: là tập các phương pháp nhằm tìm ra các nhóm con trong dữ liệu
  - Các mẫu có đặc điểm chung trong cùng 1 nhóm nhưng khác với các mẫu ở ngoài nhóm
  - Việc gom nhóm là phân tích cấu trúc dữ liệu nội tại, điều này khác với phân lớp

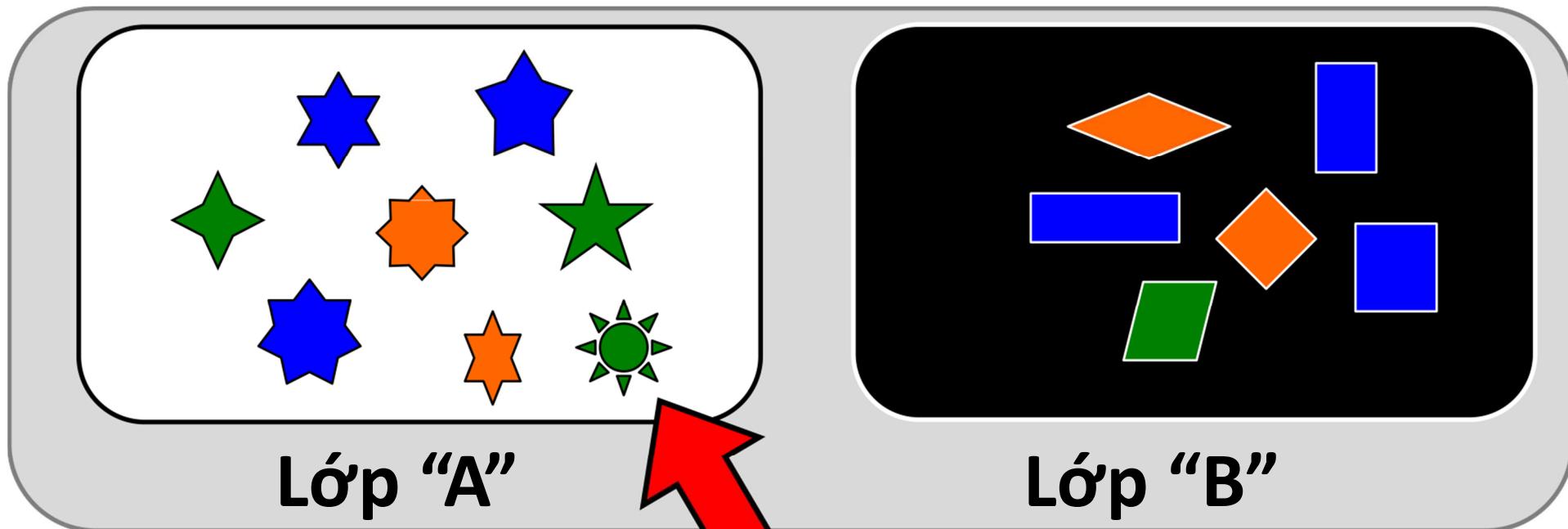
# Phân cụm vs. Phân lớp



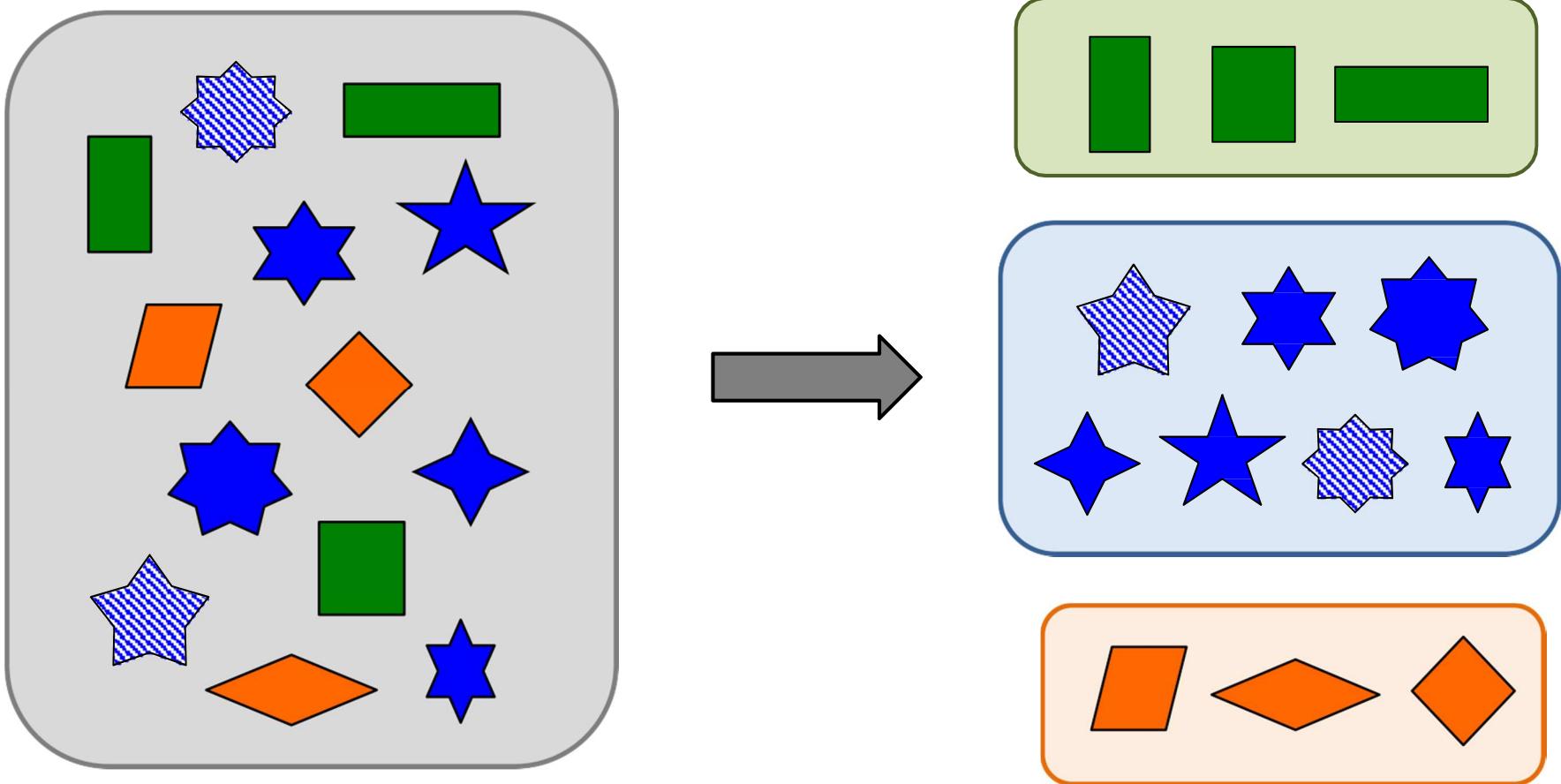
# Phân lớp



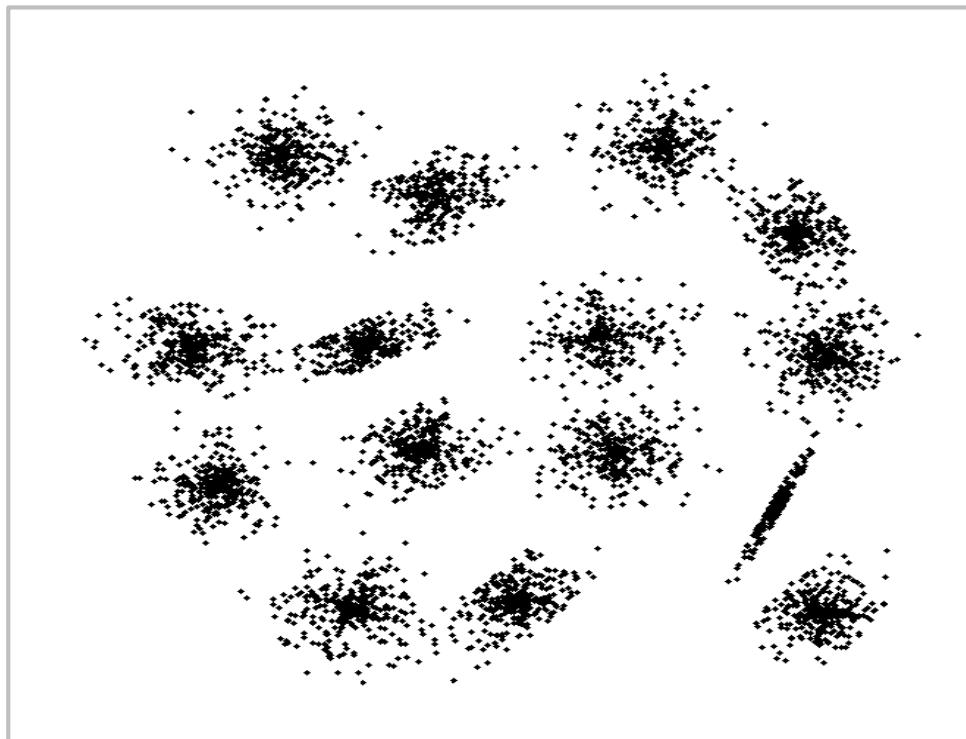
# Phân lớp



# Phân cụm



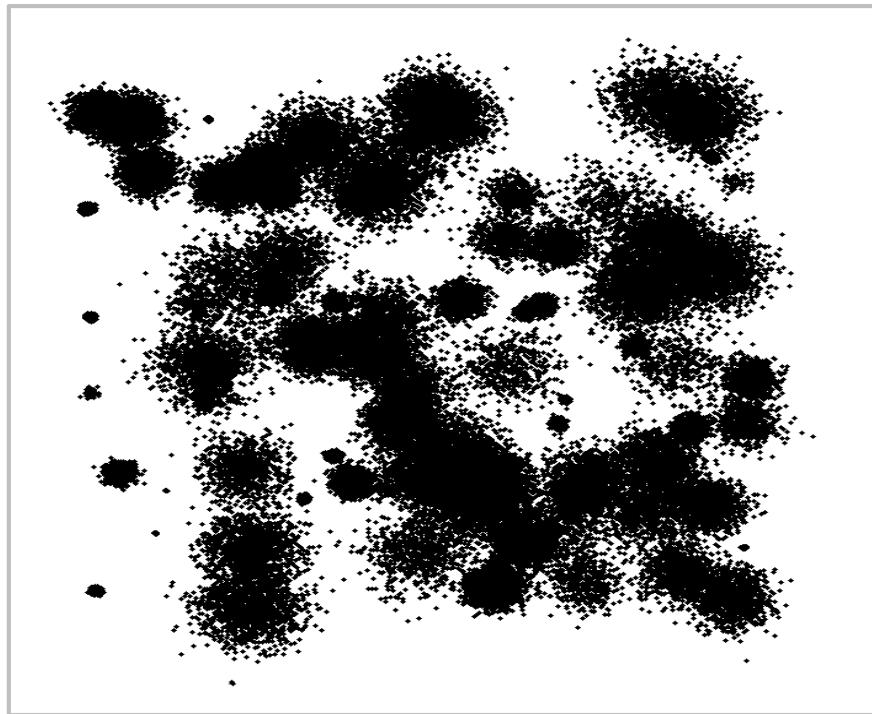
# Phân cụm



Dữ liệu lấy từ: <http://cs.joensuu.fi/sipu/datasets/>



# Phân cụm



Dữ liệu lấy từ: <http://cs.joensuu.fi/sipu/datasets/>

# Phân cụm

- Các kiểu mô hình phân cụm
  - Hai mô hình phân cụm thông dụng:
    - Phương pháp dựa trên tâm cụm (Centroid-based)
    - Phương pháp phân cấp (Hierarchical)
  - Các mô hình khác:
    - Phân cụm dựa trên mô hình (Model-based)
      - Mỗi cụm được thể hiện bằng một phân bố thống kê tham số
      - Dữ liệu là một hỗn hợp các phân bố
    - Khái niệm phân cụm fuzzy cứng vs. mềm
      - Cứng (Hard): Các mẫu được chia thành các cụm riêng biệt
      - Mềm (Soft): Các mẫu có thể thuộc nhiều hơn 1 cụm

# Phương pháp phân cấp

- Phương pháp phân cấp (phân cụm cây)

- Các cụm dựa trên khoảng cách giữa các mẫu
- Hiển thị theo phân cấp mà không theo cách phân hoạch dữ liệu

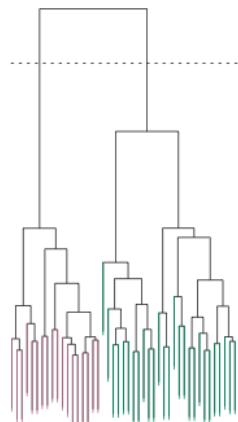
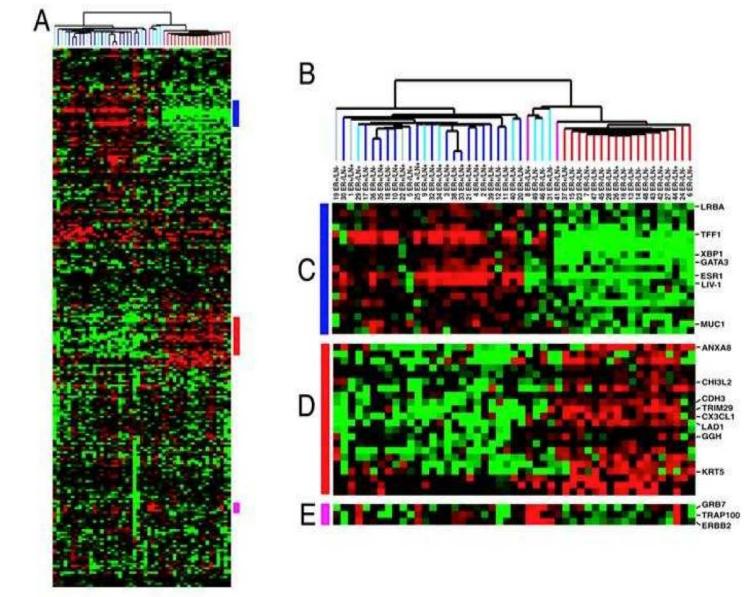


Figure 10.9 , ISL 2013



Sørlie, Therese, et al. (2003) "Repeated observation of breast tumor subtypes in independent gene expression data sets," PNAS.



# Phân cụm K-means

- Gom nhóm dữ liệu thành K cụm riêng biệt
  - Mỗi cụm K được định nghĩa bởi 1 véc tơ tâm cụm (centroid)
    - Tâm cụm: giá trị trung bình của tất cả các đối tượng trong cụm
  - Mỗi đối tượng gán cho 1 cụm đơn (tâm cụm gần nhất)
  - Yêu cầu số lượng cụm đầu vào K
  - “Phân cụm tốt” cực tiểu sự biến đổi giữa các cụm
    - “Tính tương tự (Similarity)” đo theo khoảng cách Euclidean

# Phân cụm K-means

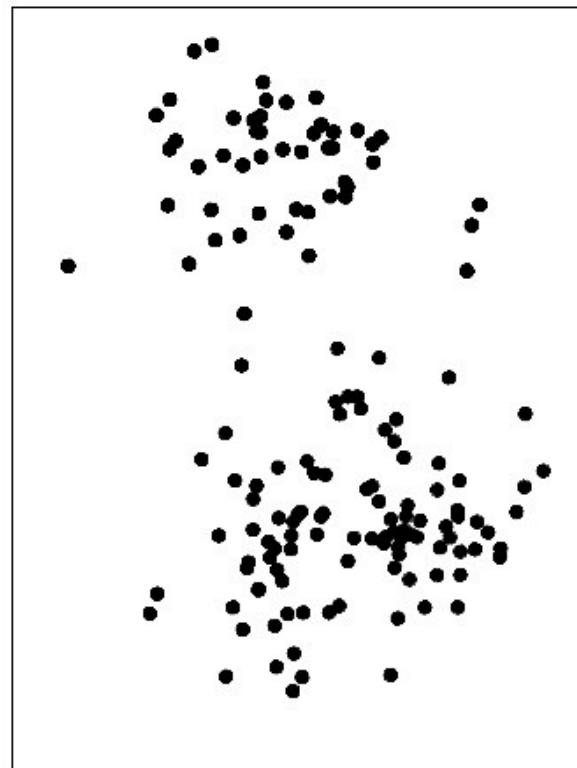


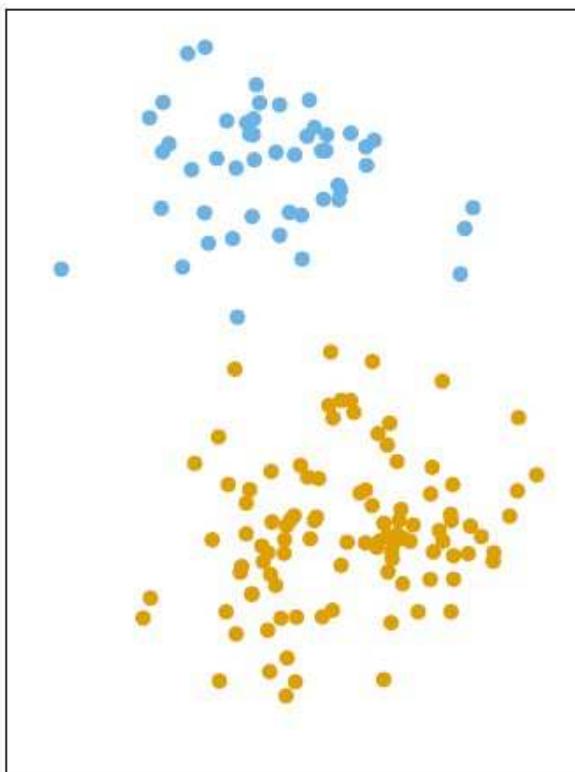
Figure 10.5 , ISL 2013\*

\*Một số hình vẽ trong bài trình bày này được lấy từ cuốn "*An Introduction to Statistical Learning, with applications in R*" (Springer, 2013) với sự đồng ý của các tác giả: G. James, D. Witten, T. Hastie and R. Tibshirani

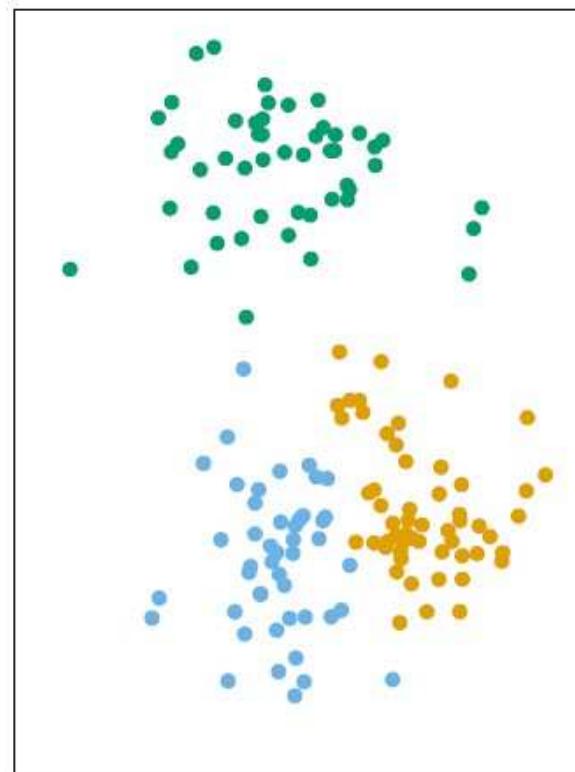


# Phân cụm K-means

K=2



K=3



K=4

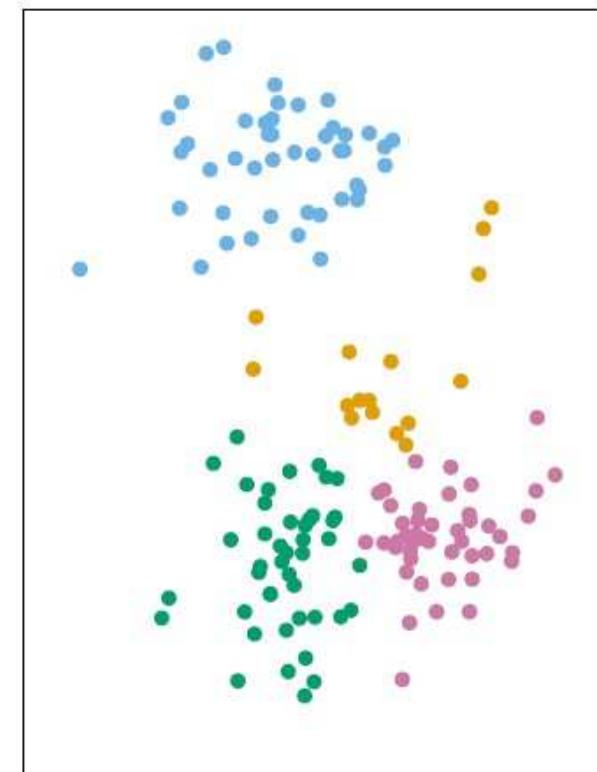


Figure 10.5 , ISL 2013



# Dữ liệu

case	sex	glasses	moustach	smile	hat
1	m	y	n	y	n
2	f	n	n	y	n
3	m	y	n	n	n
4	m	n	n	n	n
5	m	n	n	y?	n
6	m	n	y	n	y
7	m	y	n	y	n
8	m	n	n	y	n
9	m	y	y	y	n
10	f	n	n	n	n
11	m	n	y	n	n
12	f	n	n	n	n



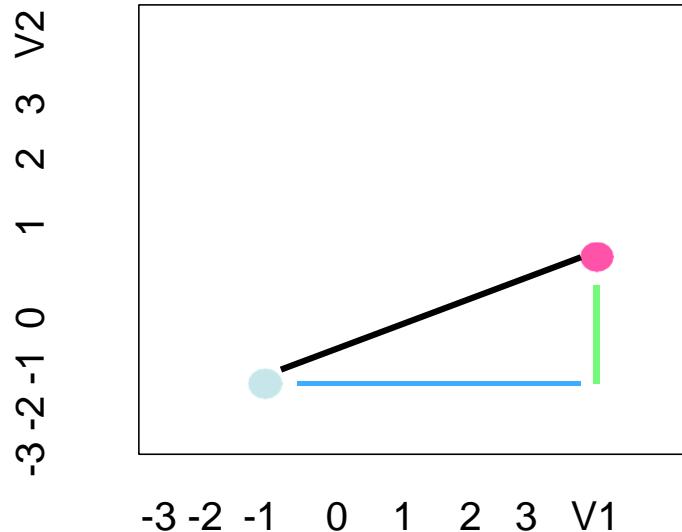
# Đo lường khoảng cách (distance)

- Euclidean  $\sqrt{[\sum (y - x)^2]}$
- Squared Euclidean  $\sum (y - x)^2$  (thông dụng)
- City-Block  $\sum |y - x|$
- Chebychev  $\max |y - x|$
- Cosine  $\cos r_{xy}$

# Euclidean

$$X = 2, 0$$

$$Y = -2, -2$$



$$\sqrt{(-2 - 2)^2 + (-2 - 0)^2}$$

$$(4^2 + 2^2) = \sqrt{20} = 4.47$$

Khoảng cách giữa X và Y = 4.47

# Squared Euclidean

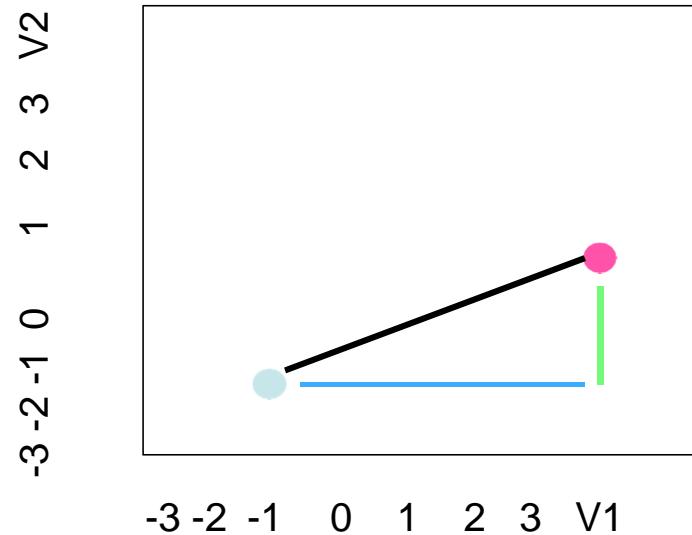
$$\sum (y - \textcolor{magenta}{x})^2$$

$$X = 2, 0$$

$$Y = -2, -2$$

$$([-2 - 2]^2 + [-2 - 0])^2$$

$$(4^2 + 2^2) = 20$$



## Squared Euclidean

# Tính toán distance

	X	Y	
(1)	4	4	$v_1$
(2)	8	4	$v_2$
(3)	15	8	$v_3$
(4)	24	4	$v_4$
(5)	24	12	$v_5$

$$\| v_3 - v_2 \|_1 = \delta_1(v_3, v_2)$$
$$= | 15-8 | + | 8-4 | = 11$$

$$\| v_3 - v_2 \|_2 = \delta_2(v_3, v_2)$$
$$= [(15-8)^2 + (8-4)^2]^{1/2}$$
$$= 65^{1/2} \sim 8.062$$

$$\| v_3 - v_2 \|_\infty =$$
$$\delta_\infty(v_3, v_2) =$$
$$\max(| 15-8 |, | 8-4 |) = 7$$



# Ví dụ với 5 điểm: Euclidean distance

	X	Y	(1)	(2)	(3)	(4)	(5)	
(1)	4	4	(1)	-	4.0	11.7	20.0	21.5
(2)	8	4	(2)		-	8.1	16.0	17.9
(3)	15	8	(3)			-	9.8	9.8
(4)	24	4	(4)				-	8.0
(5)	24	12	(5)					-

# Ma trận tương tự (Dissimilarity matrix)

(1) (2) (3) (4) (5)

\* 4.0 11.7 20.0 21.5 (1)

\* 8.1 16.0 17.9 (2)

\* 9.8 9.8 (3)

\* 8.0 (4)

\* (5)



# Phân cụm K-means

- Các tâm cụm cực tiểu sự biến đổi giữa các cụm

$$J = \frac{1}{n} \sum_{i=1}^K \sum_{x \in c_i} |x - \mu_i|^2$$

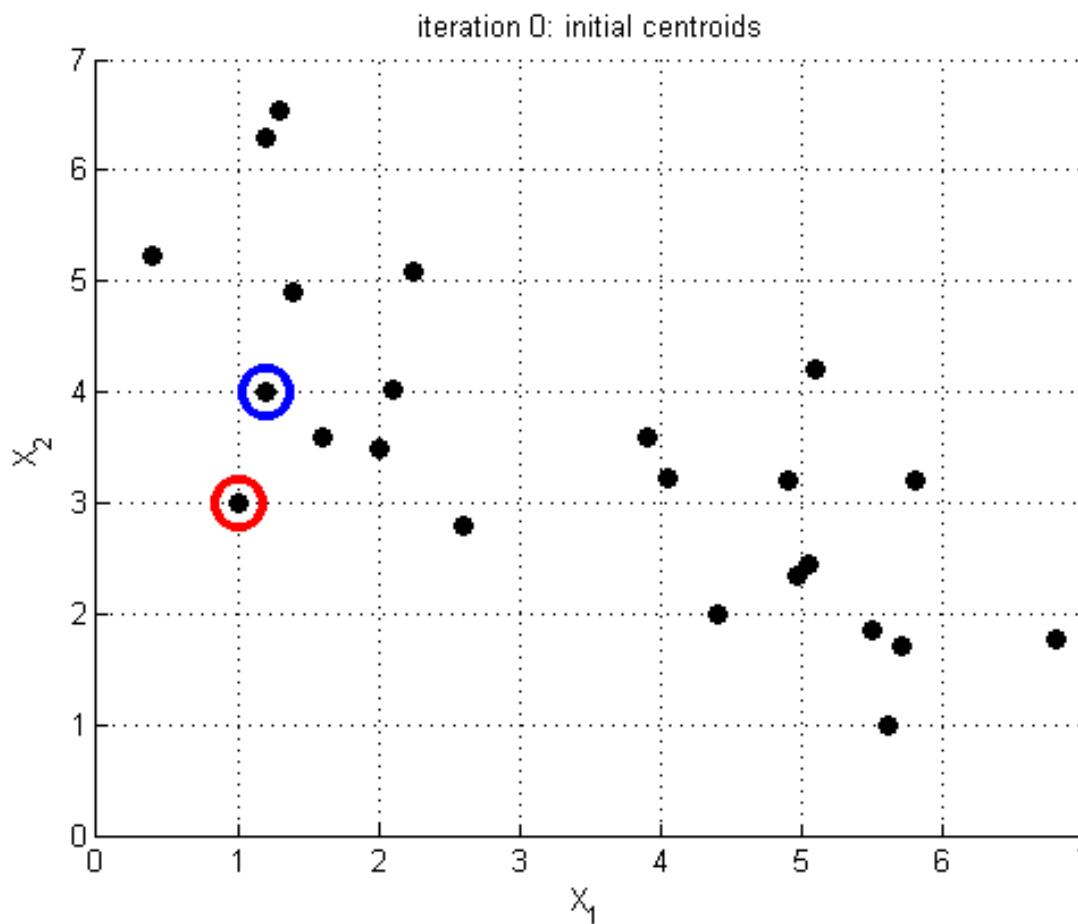
– Các tâm cụm (trung tâm của cụm):  $\mu_i = \frac{1}{|c_i|} \sum_{x \in c_i} x$

- Bài toán cực tiểu hóa này là tối ưu tổ hợp
  - Giải pháp cho cực tiểu hóa địa phương ta sử dụng phương pháp lặp

# Thuật toán K-means

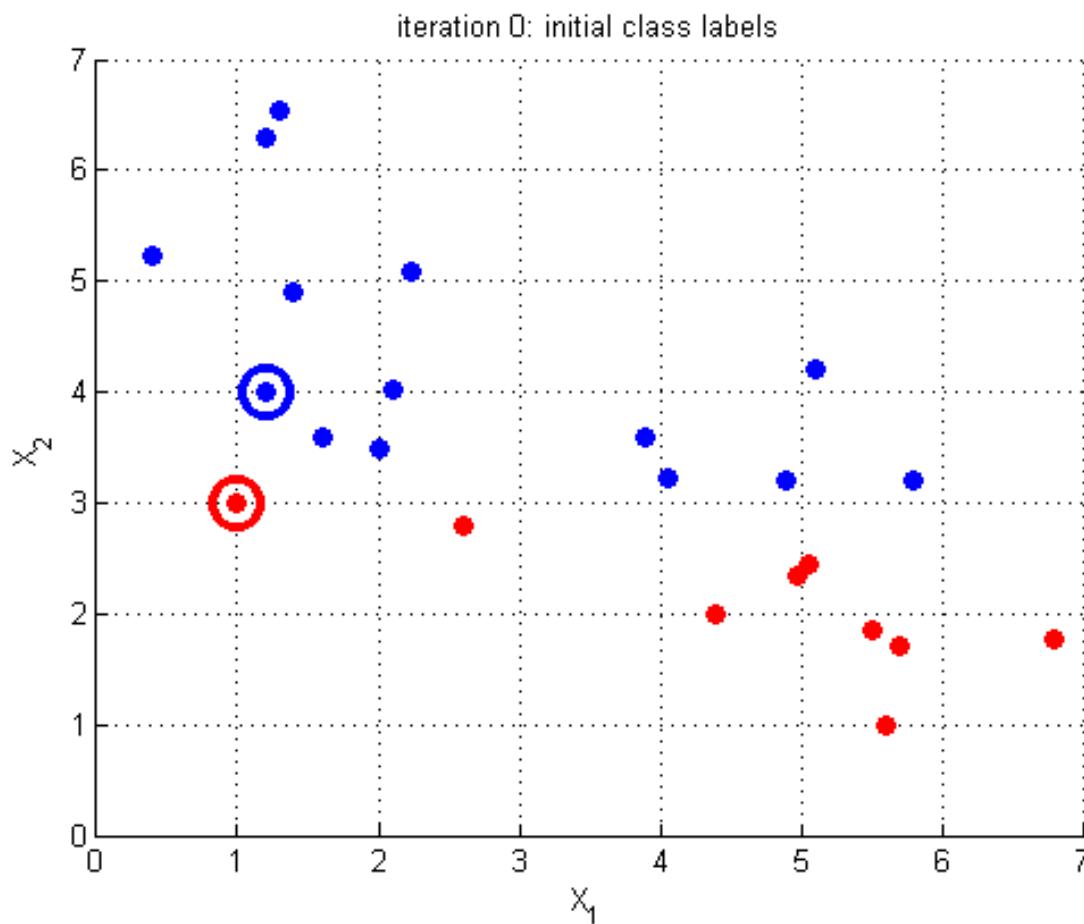
- 1) Khởi tạo chọn ngẫu nhiên K tâm cụm
- 2) Phân hoạch dữ liệu bằng cách gán mỗi đối tượng vào cụm mà nó gần tâm nhất
- 3) Tính các tâm cụm mới trong mỗi cụm
- 4) Lặp lại 2 và 3 cho đến khi thỏa mãn điều kiện
  - “thỏa mãn điều kiện” khi các tâm cụm ổn định và các đối tượng không dịch chuyển giữa các cụm

# Thuật toán K-means



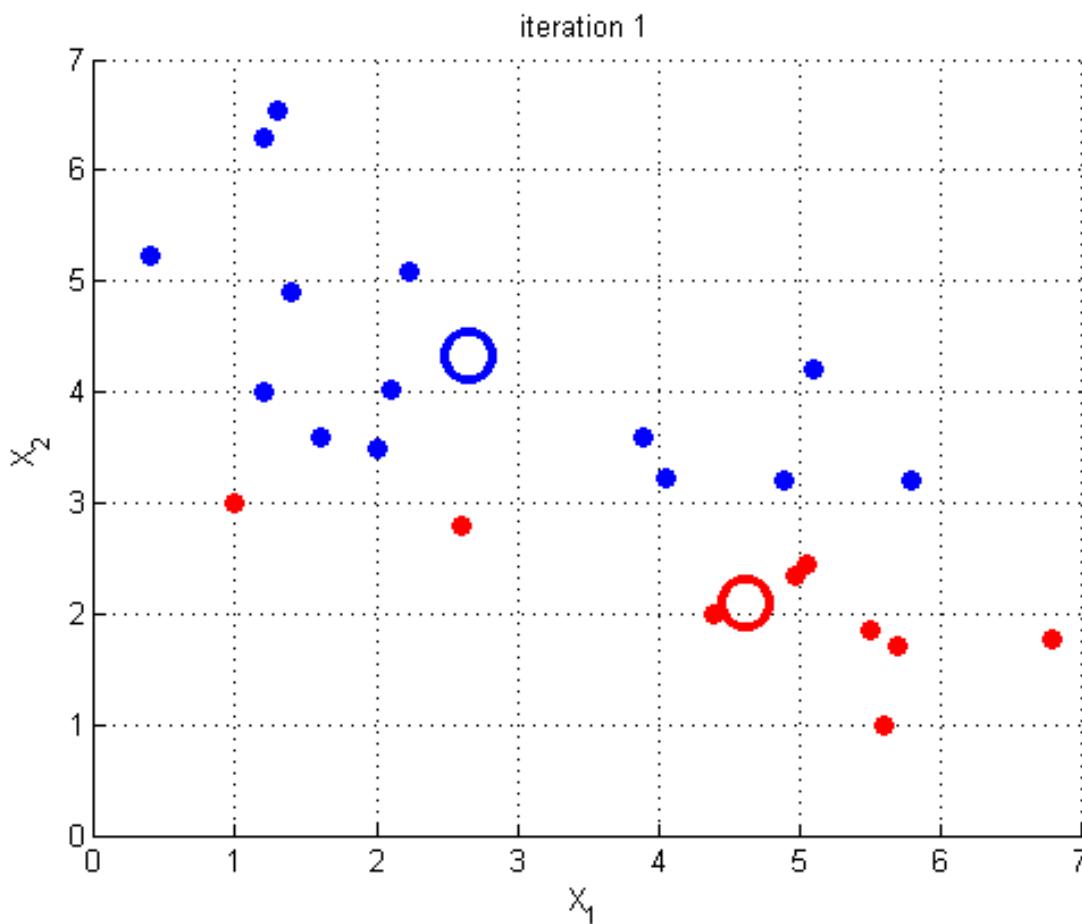
Khởi tạo tâm cụm

# Thuật toán K-means



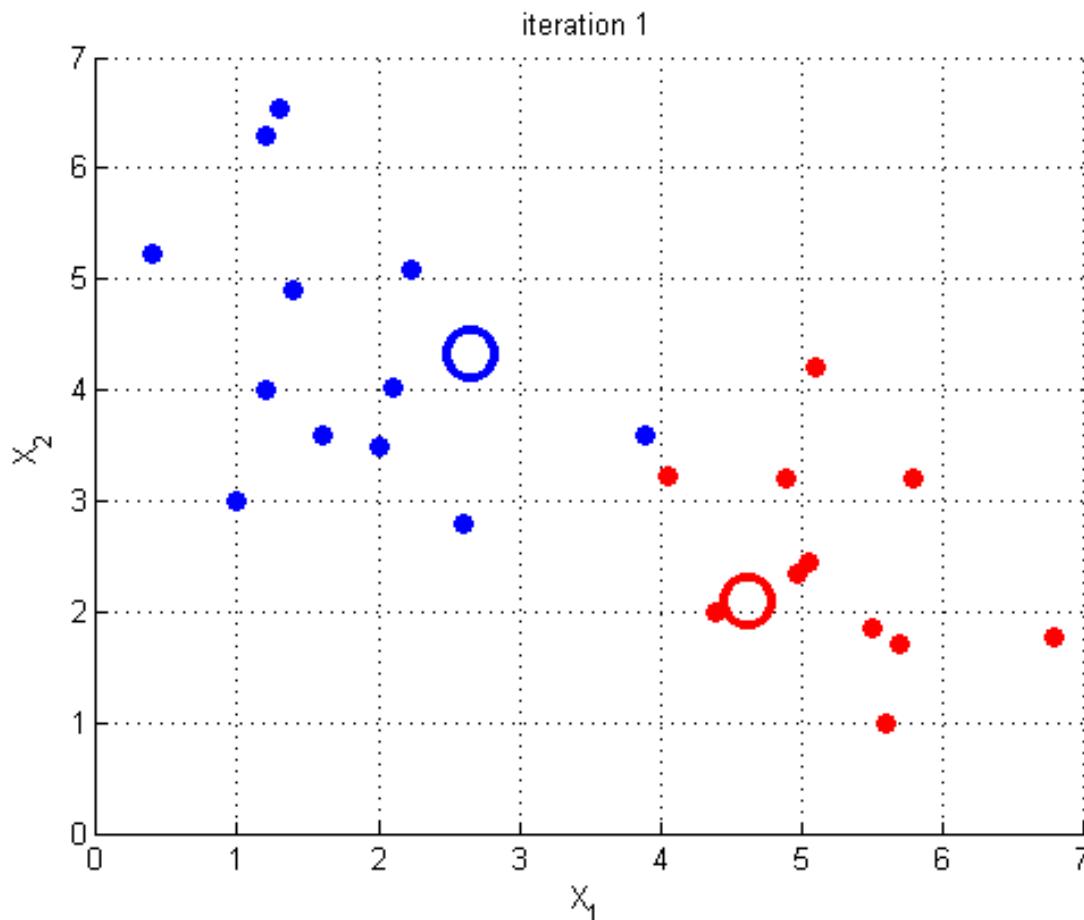
Khởi tạo tâm cụm  
Gán các cụm ban đầu

# Thuật toán K-means



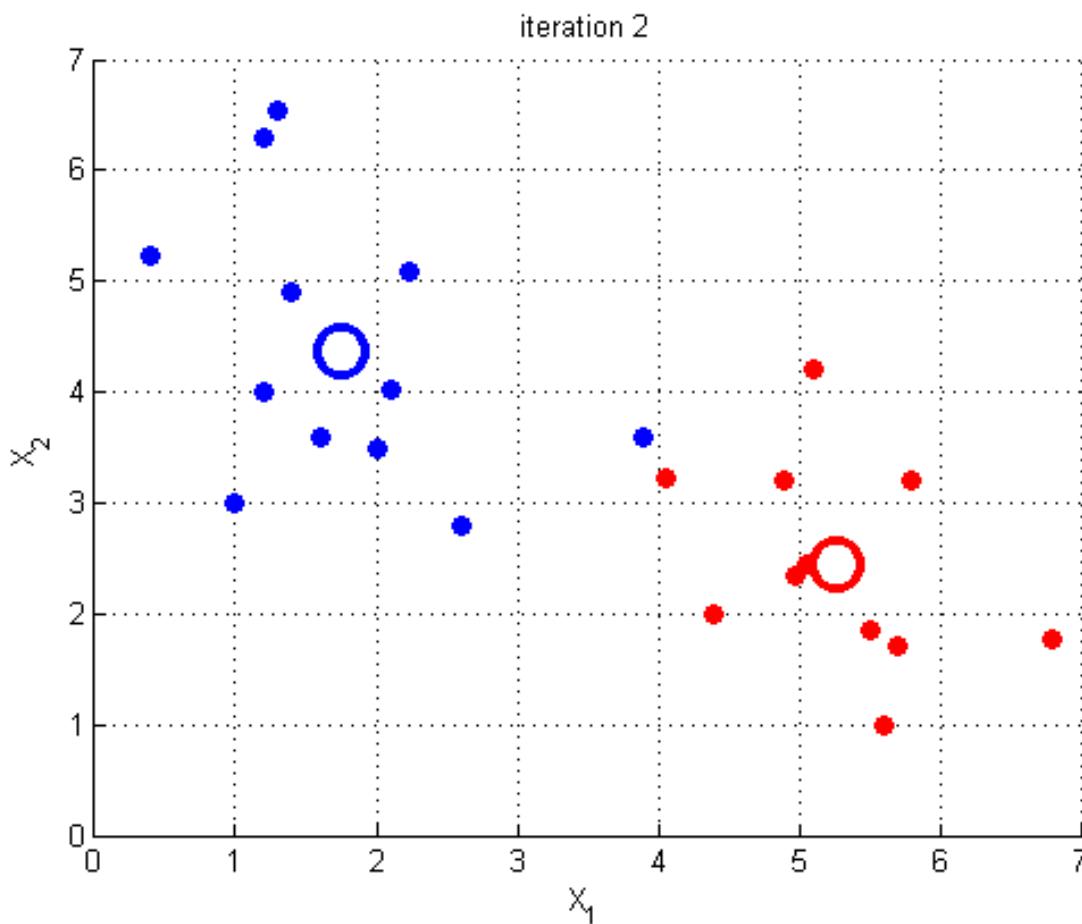
- Khởi tạo tâm cụm
- Gán các cụm ban đầu
- Cập nhật các tâm cụm

# Thuật toán K-means



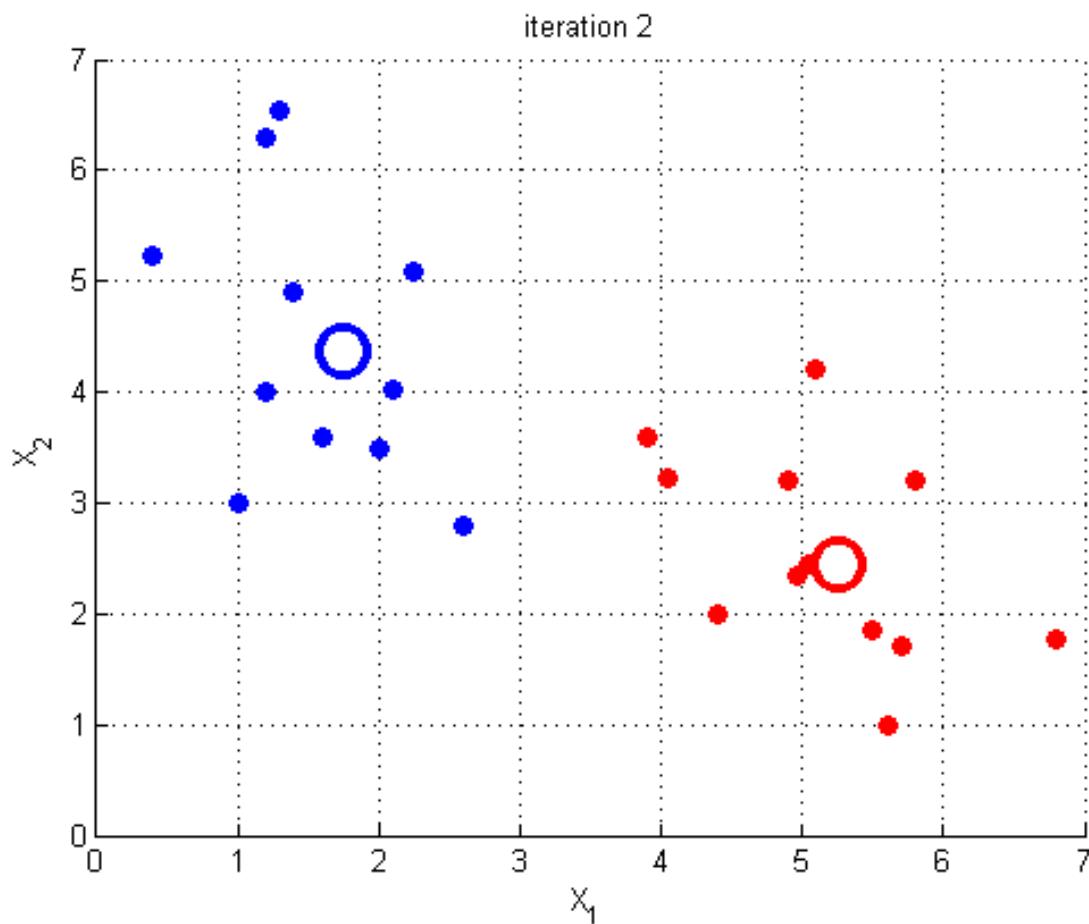
- Khởi tạo tâm cụm
- Gán các cụm ban đầu
- Cập nhật các tâm cụm
- Gán lại các cụm

# Thuật toán K-means



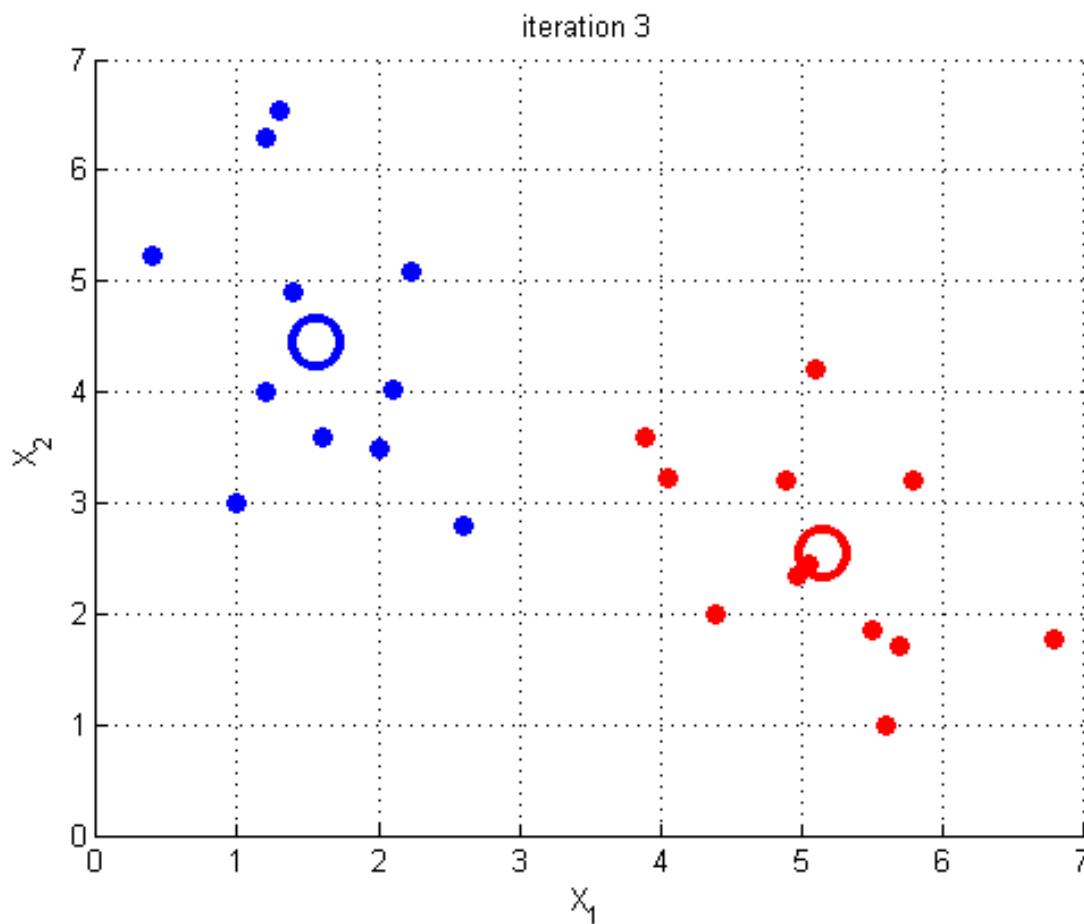
- Khởi tạo tâm cụm
- Gán các cụm ban đầu
- Cập nhật các tâm cụm
- Gán lại các cụm
- Cập nhật tâm cụm

# Thuật toán K-means



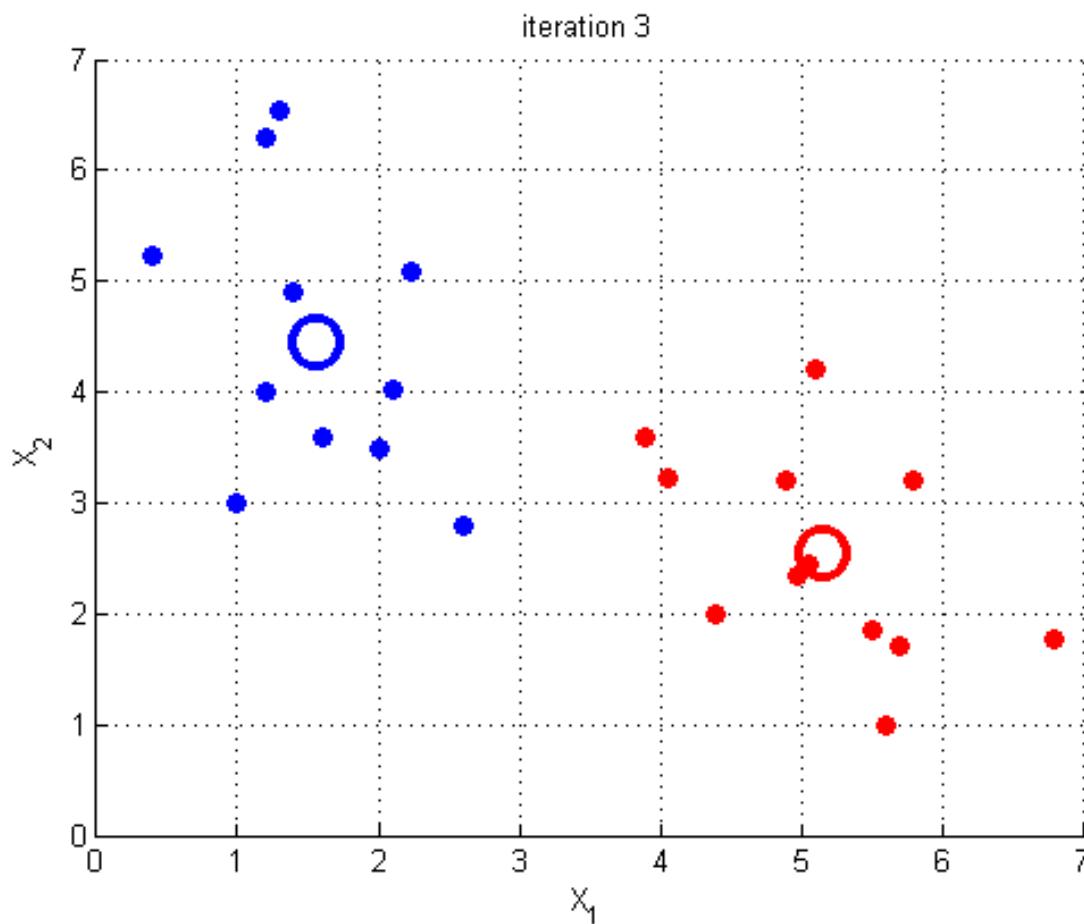
- Khởi tạo tâm cụm
- Gán các cụm ban đầu
- Cập nhật các tâm cụm
- Gán lại các cụm
- Cập nhật tâm cụm
- Gán lại các cụm

# Thuật toán K-means

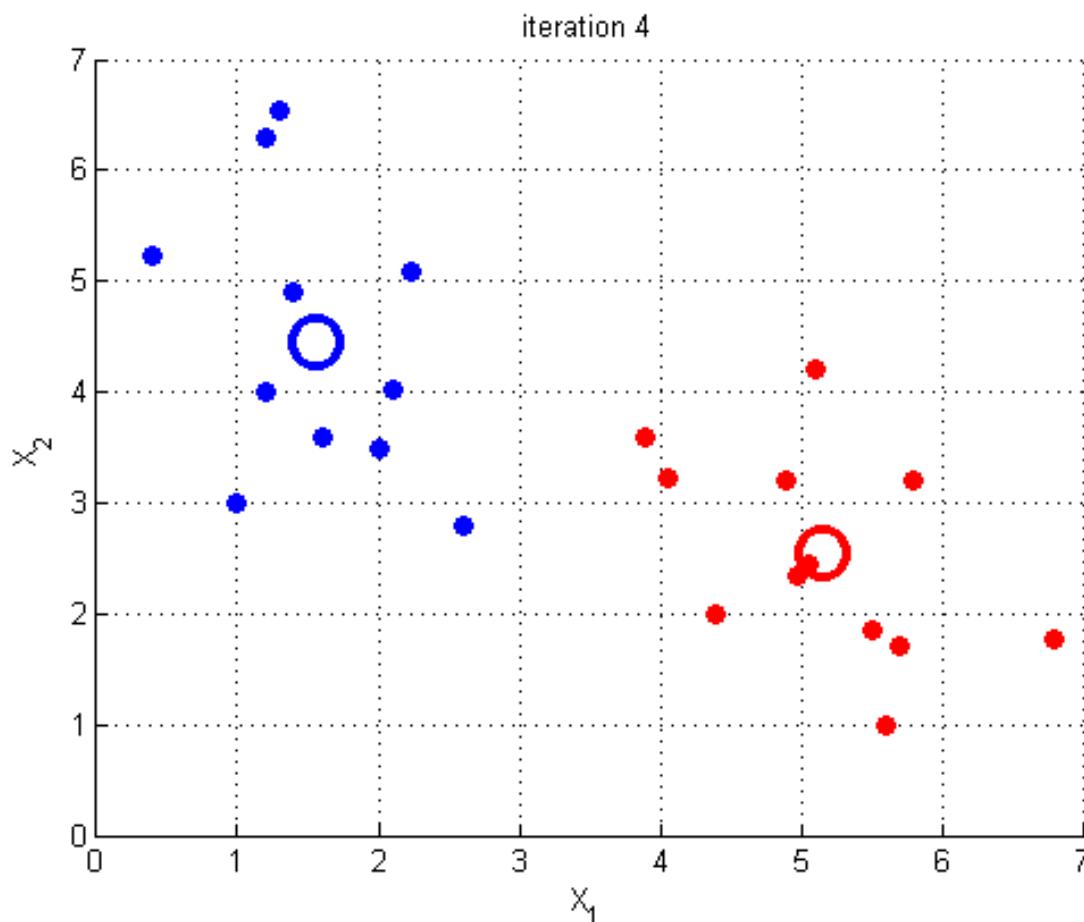


- Khởi tạo tâm cụm
- Gán các cụm ban đầu
- Cập nhật các tâm cụm
- Gán lại các cụm
- Cập nhật tâm cụm
- Gán lại các cụm
- Cập nhật tâm cụm

# Thuật toán K-means

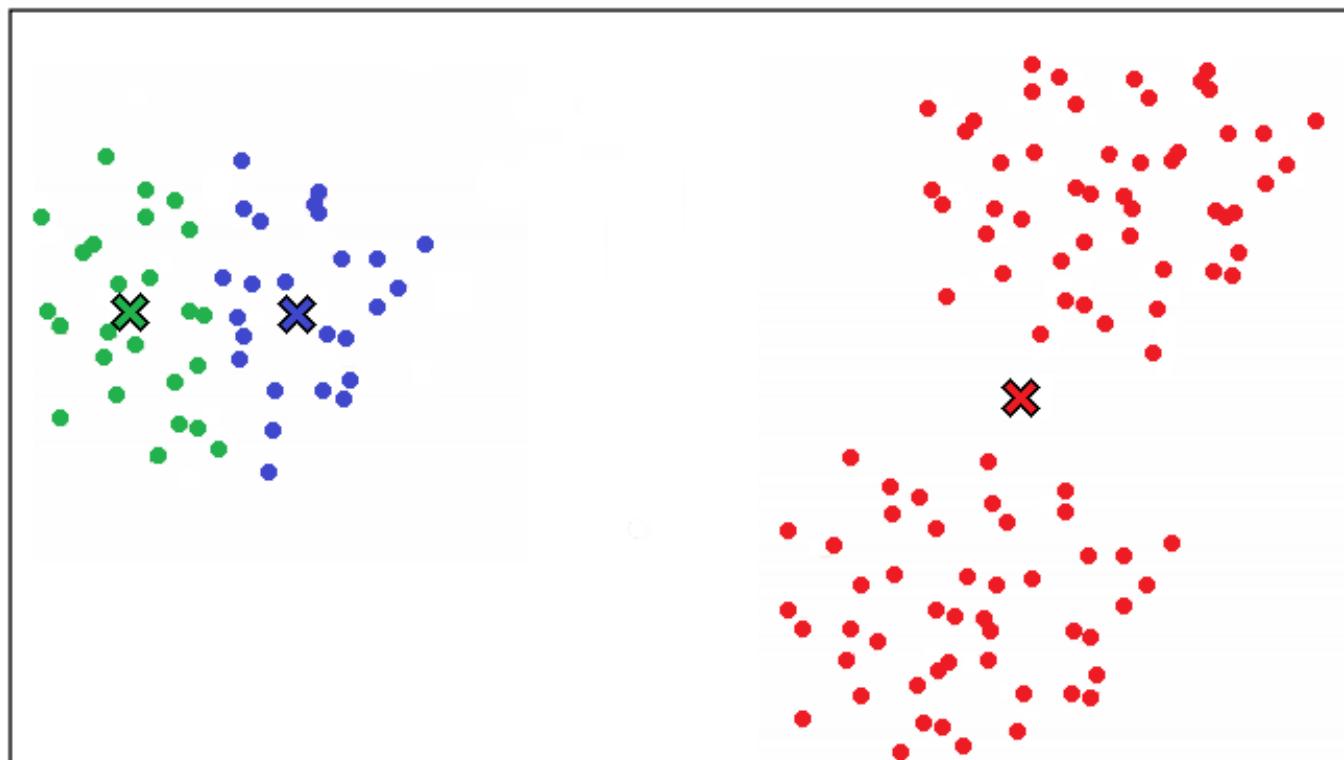


# Thuật toán K-means



# Thuật toán K-means

- Khởi tạo không tốt dẫn đến kết quả phân cụm kém



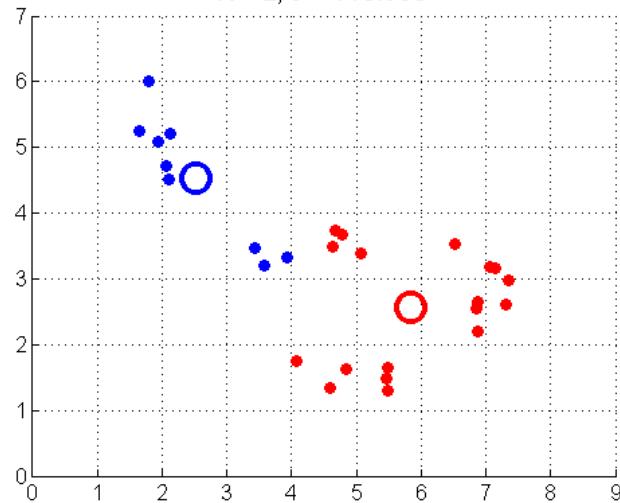
# Khởi tạo tâm cụm

- Chọn ngẫu nhiên trong K đối tượng
- Phân hoạch ngẫu nhiên dữ liệu
- Chọn K điểm xa nhau “far apart”
- Khởi tạo bằng cách sử dụng kết quả của phương pháp phân cụm khác

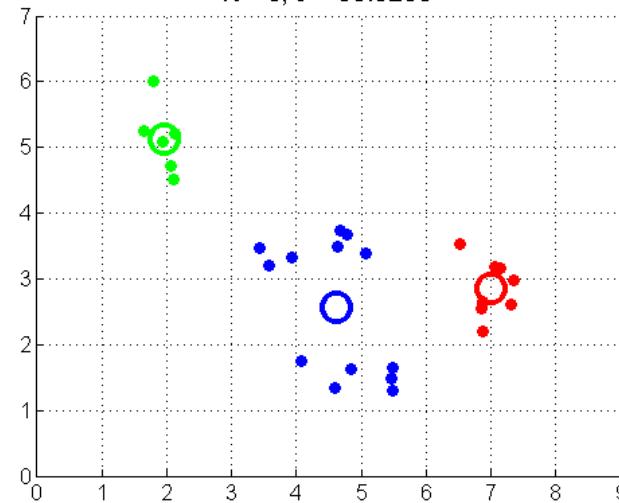
# Bao nhiêu cụm?

- K-means yêu cầu đầu vào K (# cụm)
  - Ta cần hiểu về bài toán ứng dụng để chọn K
  - Ngược lại, việc chọn K được xác định từ dữ liệu

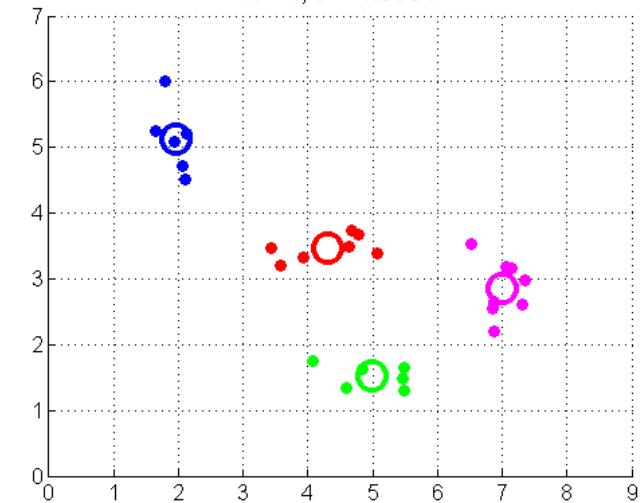
$K = 2, J = 110.935$



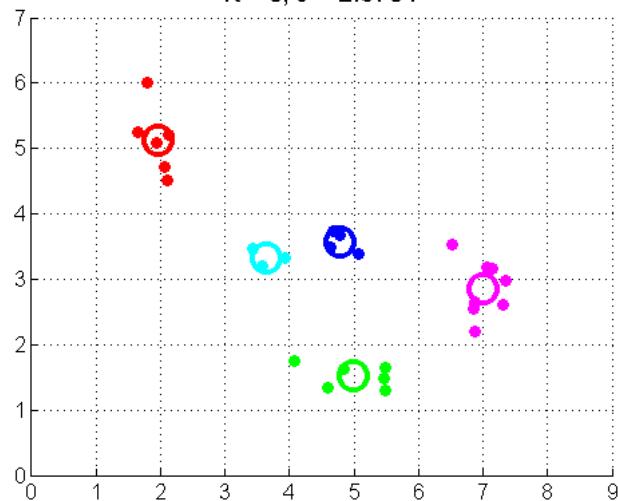
$K = 3, J = 30.5285$



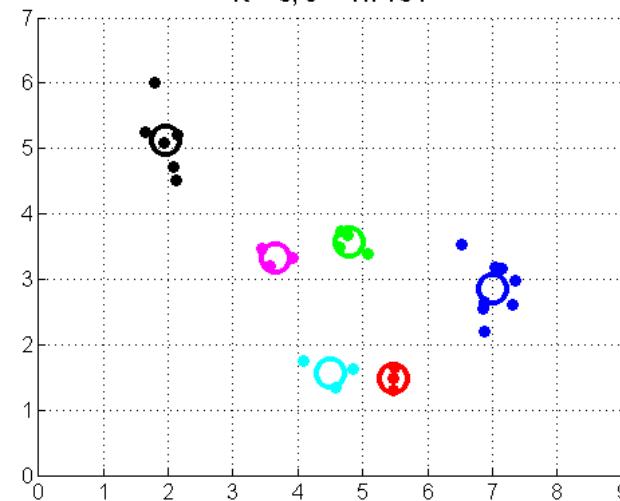
$K = 4, J = 4.0934$



$K = 5, J = 2.6754$



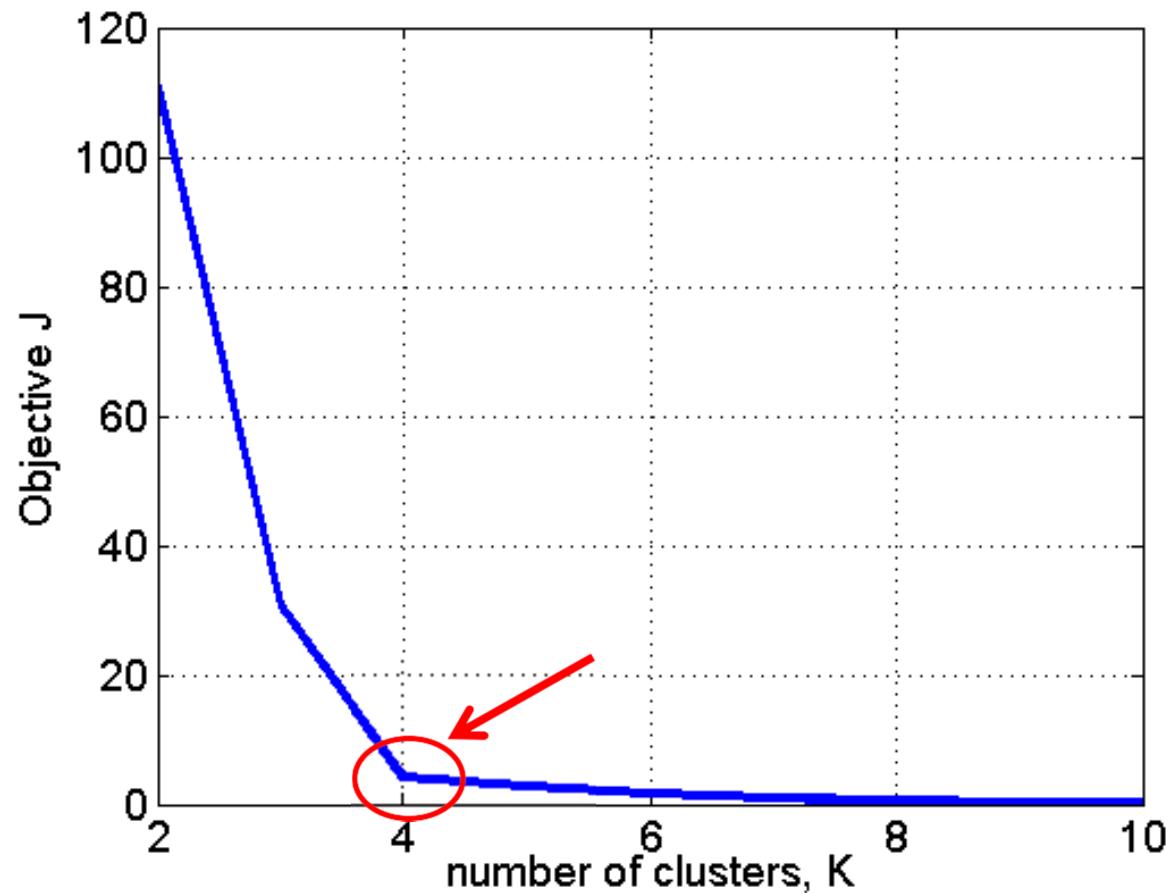
$K = 6, J = 1.7164$



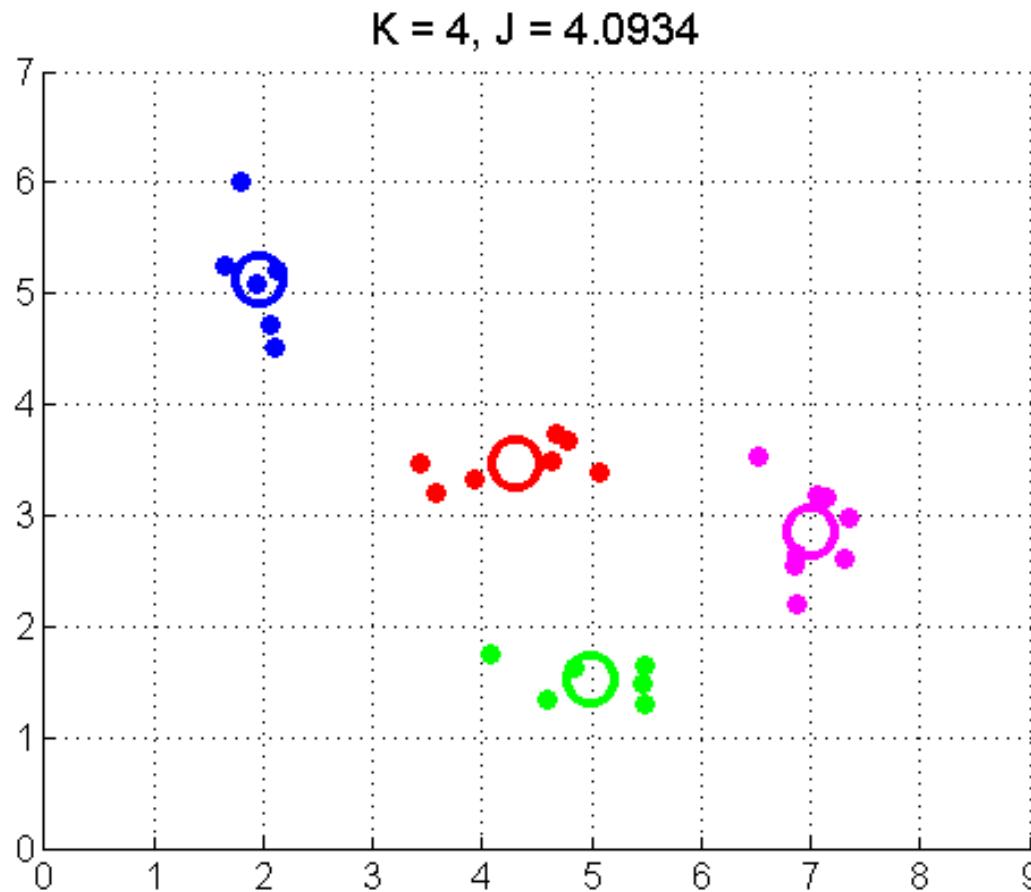
# Bao nhiêu cụm?

- Không thể tính được giá trị  $K$  để cực tiểu mục tiêu  $J$ 
  - $J$  giảm đồng thời với tăng  $K$
- Phương pháp dựa trên kinh nghiệm (Heuristic):
  - Với mỗi giá trị ứng viên của  $K$ ,
  - Tính toán phân cụm bằng K-means  $M$  lần, tìm mục tiêu nhỏ nhất  $J_K$
  - Tìm điểm “khuỷu tay (elbow)” trong đường mục tiêu ( $K$  vs  $J_K$ )

# Bao nhiêu cụm?



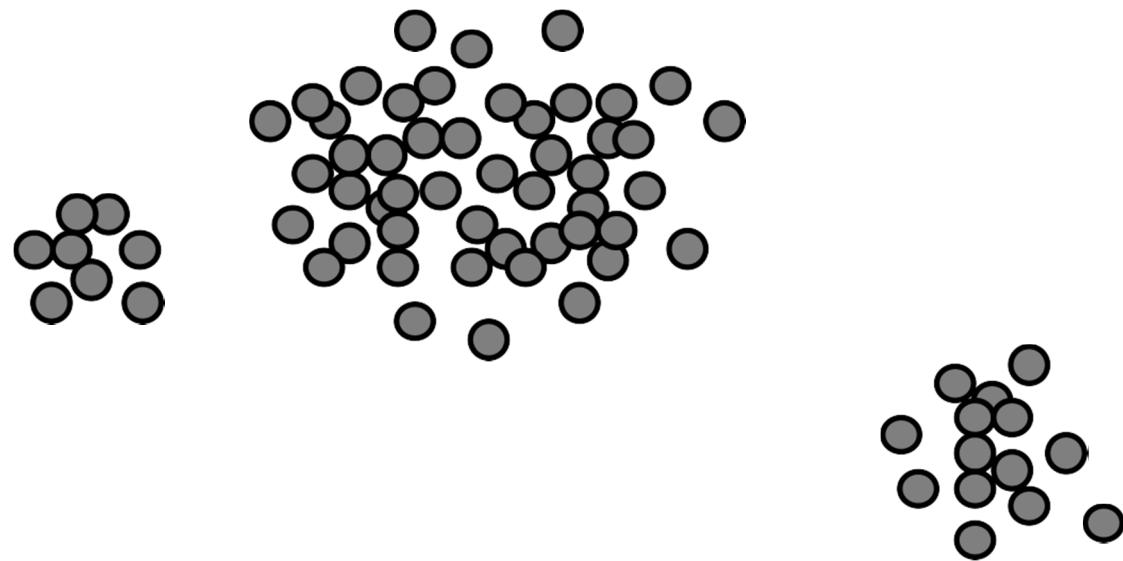
# Bao nhiêu cụm?



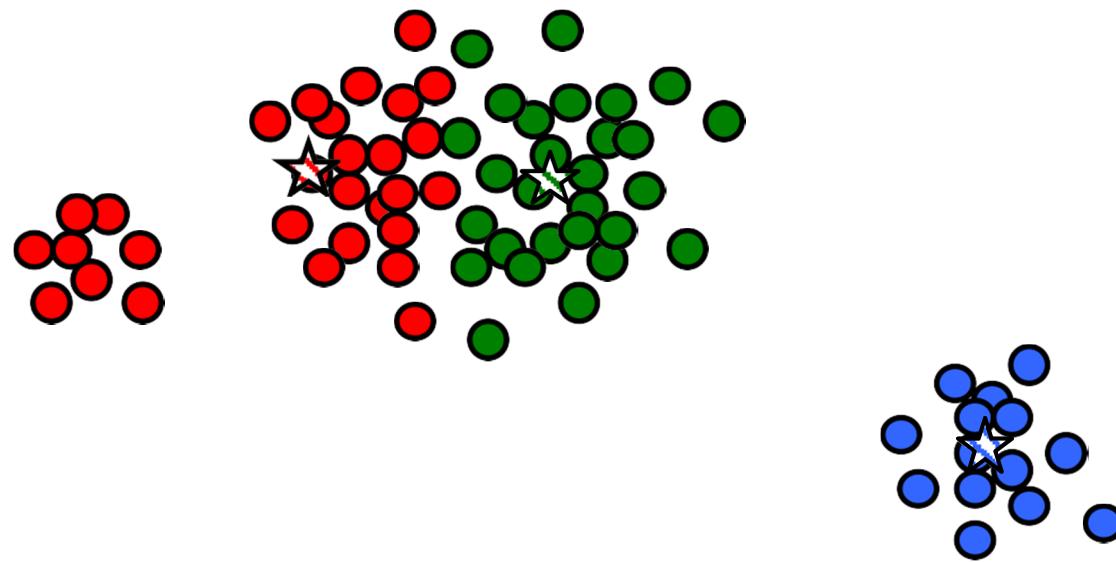
# Thuật toán K-means

- **Ưu điểm**
  - Dễ cài đặt
  - Luôn hội tụ với số lần lặp ít
  - Có thể triển khai trên những tập dữ liệu với số chiều lớn
- **Nhược điểm**
  - Giá trị K là tham số đầu vào (khó xác định tối ưu)
  - Thuật toán lặp trả về cực tiểu địa phương\*

# Thuật toán K-means



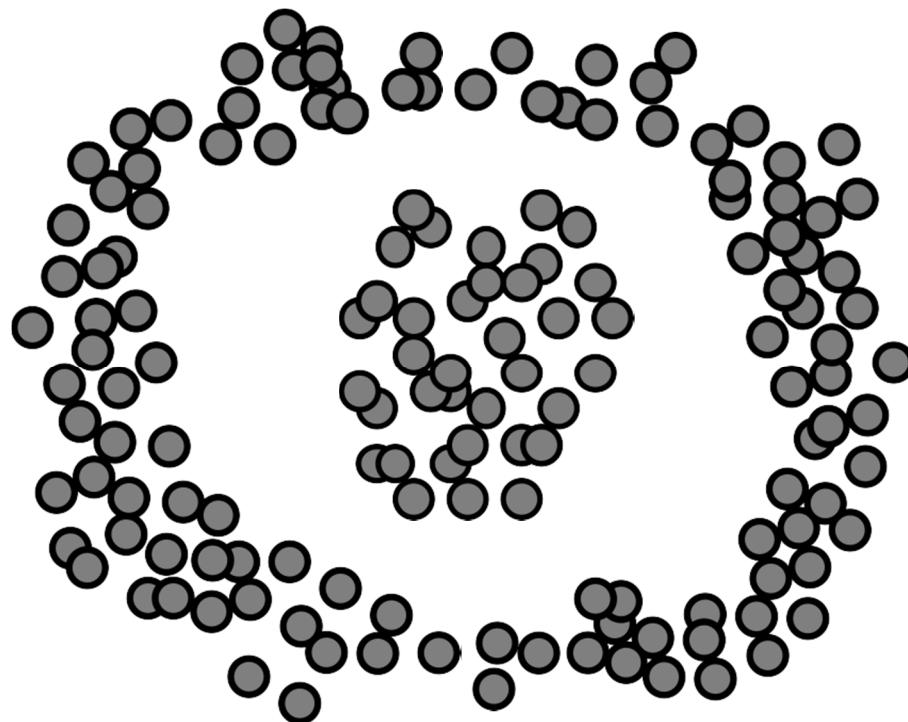
# Thuật toán K-means



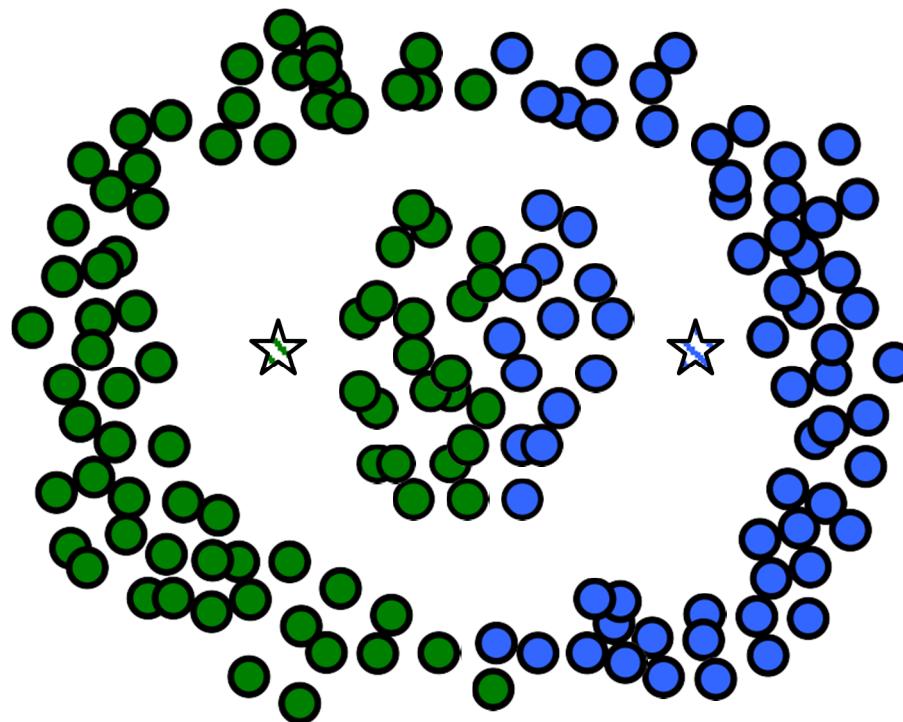
# Thuật toán K-means

- **Ưu điểm**
  - Dễ cài đặt
  - Luôn hội tụ với số lần lặp ít
  - Có thể triển khai trên những tập dữ liệu với số chiều lớn
- **Nhược điểm**
  - Giá trị K là tham số đầu vào (khó xác định tối ưu)
  - Thuật toán lặp trả về cực tiểu địa phương\*
  - Giả thiết tất cả các cụm hình cầu và có kích thước xấp xỉ nhau \*

# Thuật toán K-means



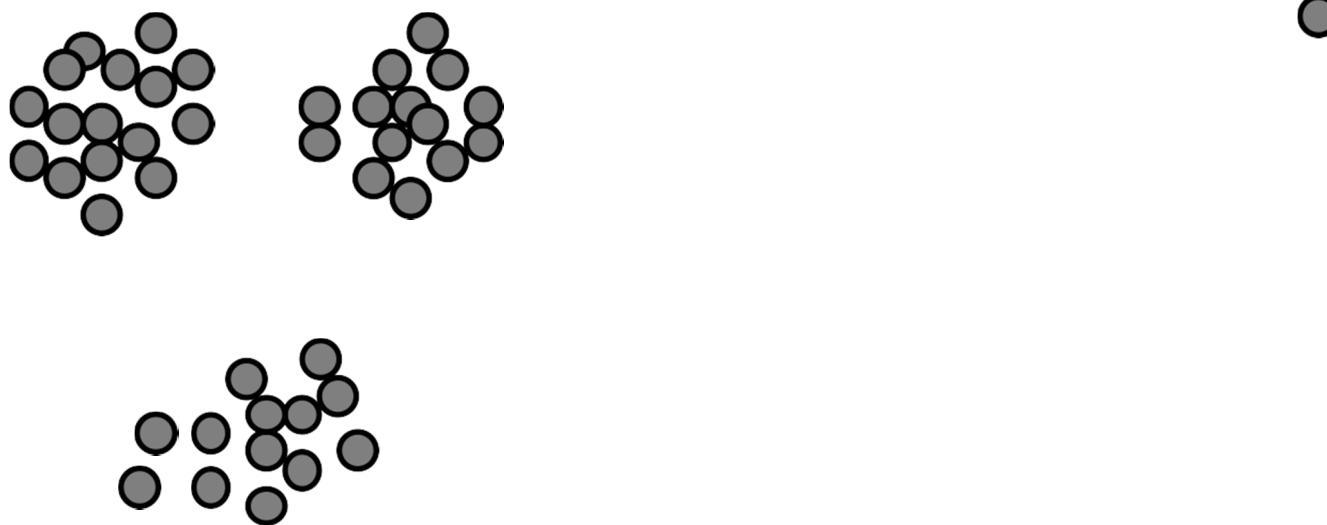
# Thuật toán K-means



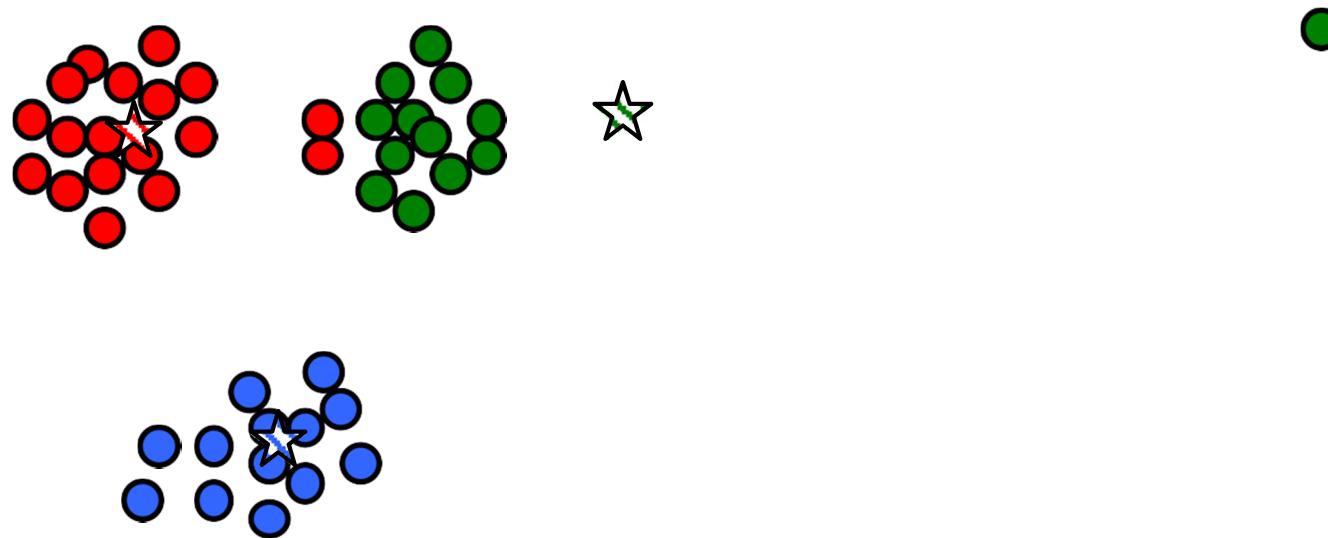
# Thuật toán K-means

- **Ưu điểm**
  - Dễ cài đặt
  - Luôn hội tụ với số lần lặp ít
  - Có thể triển khai trên những tập dữ liệu với số chiều lớn
- **Nhược điểm**
  - Giá trị K là tham số đầu vào (khó xác định tối ưu)
  - Thuật toán lặp trả về cực tiểu địa phương\*
  - Giả thiết tất cả các cụm hình cầu và có kích thước xấp xỉ nhau \*
  - Nhạy với các phần tử ngoại lai\*

# Thuật toán K-means



# Thuật toán K-means



# Thuật toán K-means

- **Ưu điểm**
  - Dễ cài đặt
  - Luôn hội tụ với số lần lặp ít
  - Có thể triển khai trên những tập dữ liệu với số chiều lớn
- **Nhược điểm**
  - Giá trị K là tham số đầu vào (khó xác định tối ưu)
  - Thuật toán lặp trả về cực tiểu địa phương\*
  - Giả thiết tất cả các cụm hình cầu và có kích thước xấp xỉ nhau \*
  - Nhạy với các phần tử ngoại lai\*
  - **\*một số nhược điểm được khắc phục bằng vài biến thể của K-means**

# Thuật toán K-means

- Khắc phục nhược điểm
  - Khởi tạo không tốt → ta chạy thuật toán nhiều lần
  - K-medians: Tâm cụm được tính bằng giá trị trung vị thay cho giá trị trung bình của K-means
  - K-medoids
    - Yêu cầu: “tâm cụm” phải là 1 trong các điểm dữ liệu
    - xử lý tốt hơn các phần tử ngoại lai
    - linh hoạt hơn – có thể dùng nhiều độ đo
    - *nhưng* thời gian tính toán lâu hơn vì phải tính các tâm cụm

# Ví dụ: Phân đoạn/nén ảnh



Ảnh: Trọng Vũ

# Phân đoạn/nén ảnh

- Ảnh → điểm ảnh (pixels) → véc tơ RGB (colors)
- Áp dụng K-means tập hợp các véc tơ RGB
  - Một véc tơ RGB ứng với 1 điểm ảnh
  - Các cụm thể hiện các màu giống nhau
- Thay thế mỗi điểm ảnh bằng một tâm cụm liên quan
  - Kết quả trên ảnh với K màu khác nhau

# Phân đoạn/nén ảnh



Chất lượng nén ảnh với số lượng  
cluster khác nhau. Ảnh: K=5

# Phân đoạn/nén ảnh



K=10

# Phân đoạn/nén ảnh



K=15

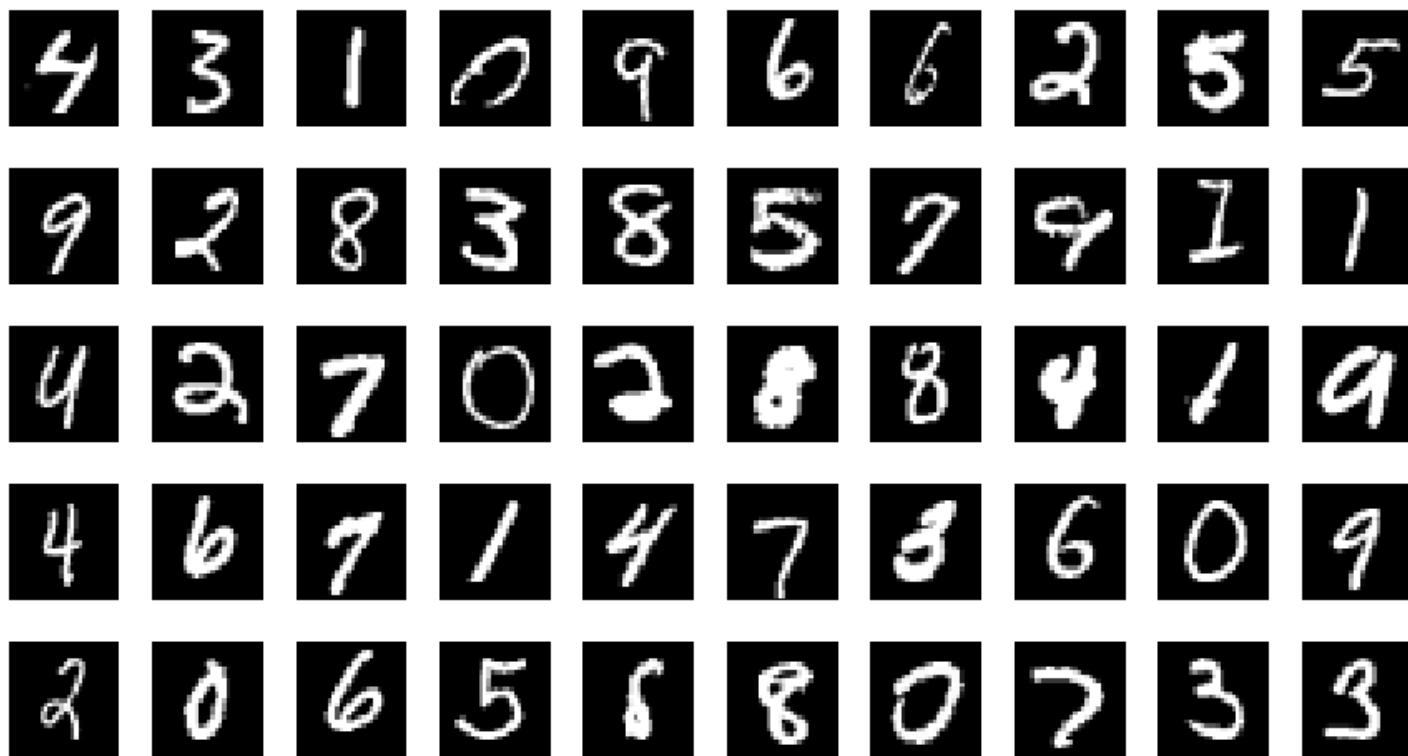


K=20

# Thuật toán K-means

- Chúng ta thực hiện thuật toán với dữ liệu có 2-3 thuộc tính (rất dễ để minh họa)
  - Trong thực tế, ta thường gặp nhiều hơn 2 thuộc tính khi phân tích dữ liệu
- Phân cụm sẽ khó khăn hơn rất nhiều khi gấp số chiều lớn

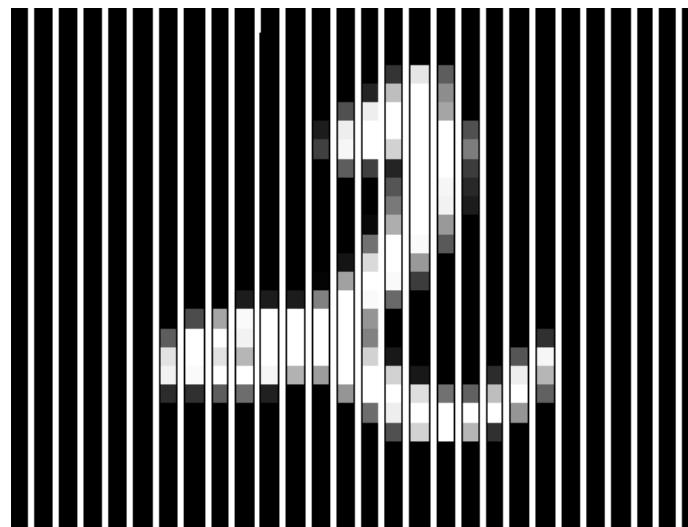
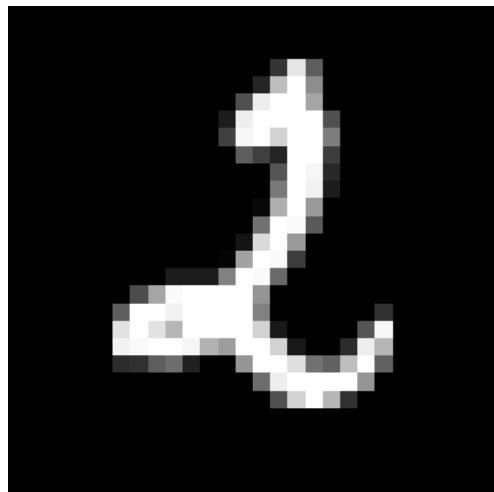
# Phân cụm chữ viết tay



MNIST dataset: <http://cis.jhu.edu/~sachin/digit/digit.html>



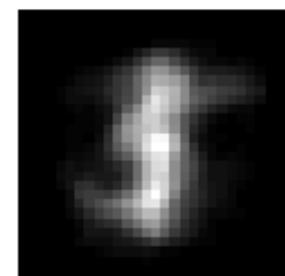
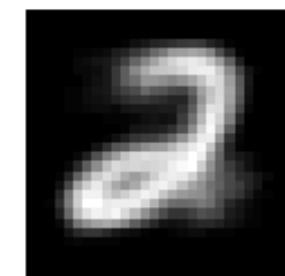
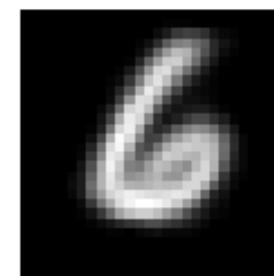
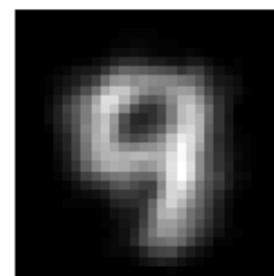
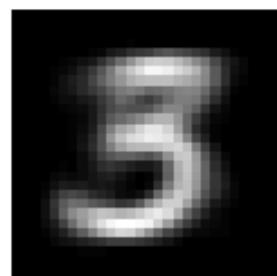
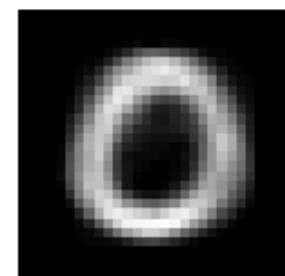
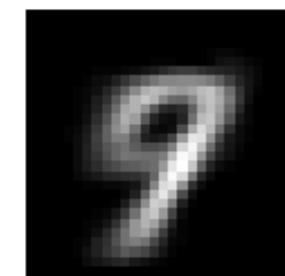
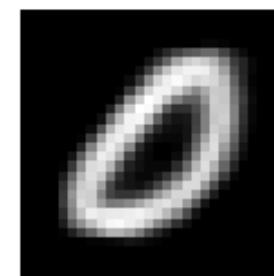
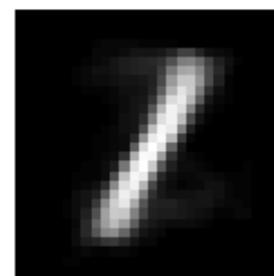
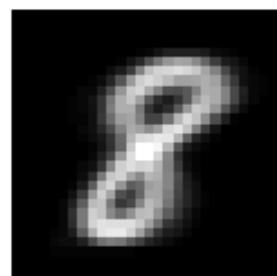
# Phân cụm chữ viết tay



# Phân cụm chữ viết tay

- Áp dụng K-means, sử dụng  $K = 10$

Cluster centroids



# Phân cụm chữ viết tay

Class A Class B Class C Class D Class E Class F Class G Class H Class I Class J

8 1 5 7 0 2 8 6 2 5

8 2 0 7 0 3 4 6 2 1

8 1 0 4 0 5 4 6 2 4

8 1 0 9 0 3 2 6 2 J

8 1 0 7 0 3 4 6 2 1

8 9 0 9 5 5 5 6 2 1

8 1 0 9 0 3 9 6 2 5

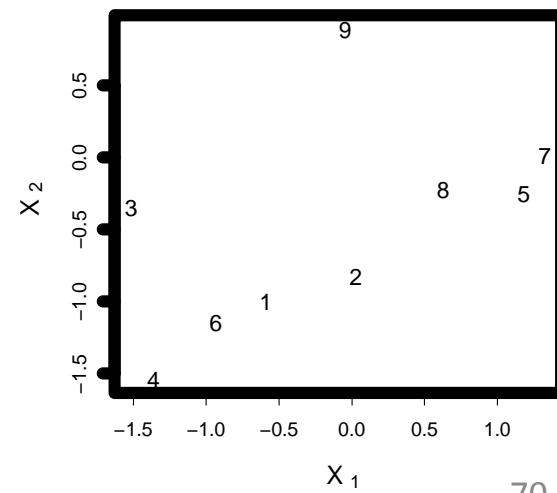
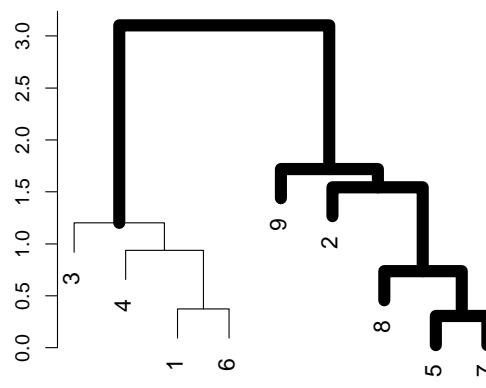


# Phân cụm phân cấp

- Phân cụm theo phương pháp K-Means yêu cầu chọn tham số đầu vào là số lượng cụm K
- Nếu ta không muốn làm theo cách trên, ta có thể dùng phương pháp phân cụm phân cấp
- Phân cụm phân cấp có ưu điểm là hiển thị các quan sát (mẫu) dạng hình cây nên dễ hình dung, được gọi là phân cụm theo cấu trúc cây (Dendogram)

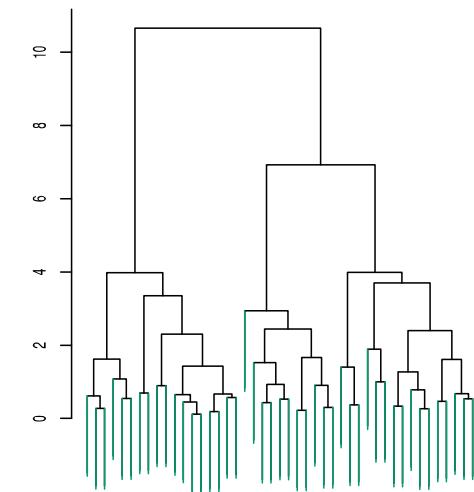
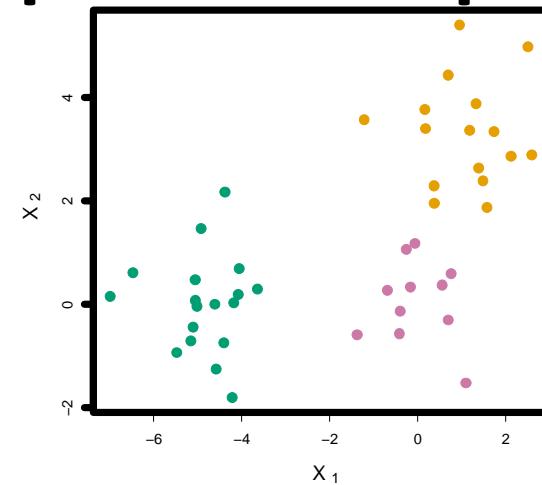
# Phân cụm phân cấp

- Đầu tiên nhập các điểm gần nhau nhất (5 và 7)
- Độ cao của việc hợp nhất (theo trực dọc) phản ánh độ tương tự của các điểm
- Sau khi các điểm được hợp nhất, chúng được xem như 1 mẫu để tiếp tục tiến hành giải thuật



# Diễn giải phương pháp phân cấp

- Mỗi “lá” của cây phân cấp biểu diễn một trong 45 mẫu
- Phần đáy của cây, mỗi mẫu là 1 lá riêng biệt. Tuy nhiên, càng lên cao các lá sẽ hợp nhất với nhau. Việc này thể hiện các mẫu có độ tương tự với các mẫu khác.
- Khi di chuyển cao lên phần ngọn của cây, số lượng mẫu đã được hợp nhất. Trước đó (phần dưới của cây) với 2 mẫu hợp nhất, chúng có chung đặc tính (gần) với nhau.



# Lựa chọn các cụm

Để chọn các cụm ta kẻ đường thẳng ngang cây phân cấp

Ta có thể chọn số lượng cụm tùy thuộc vào vị trí đường kẻ



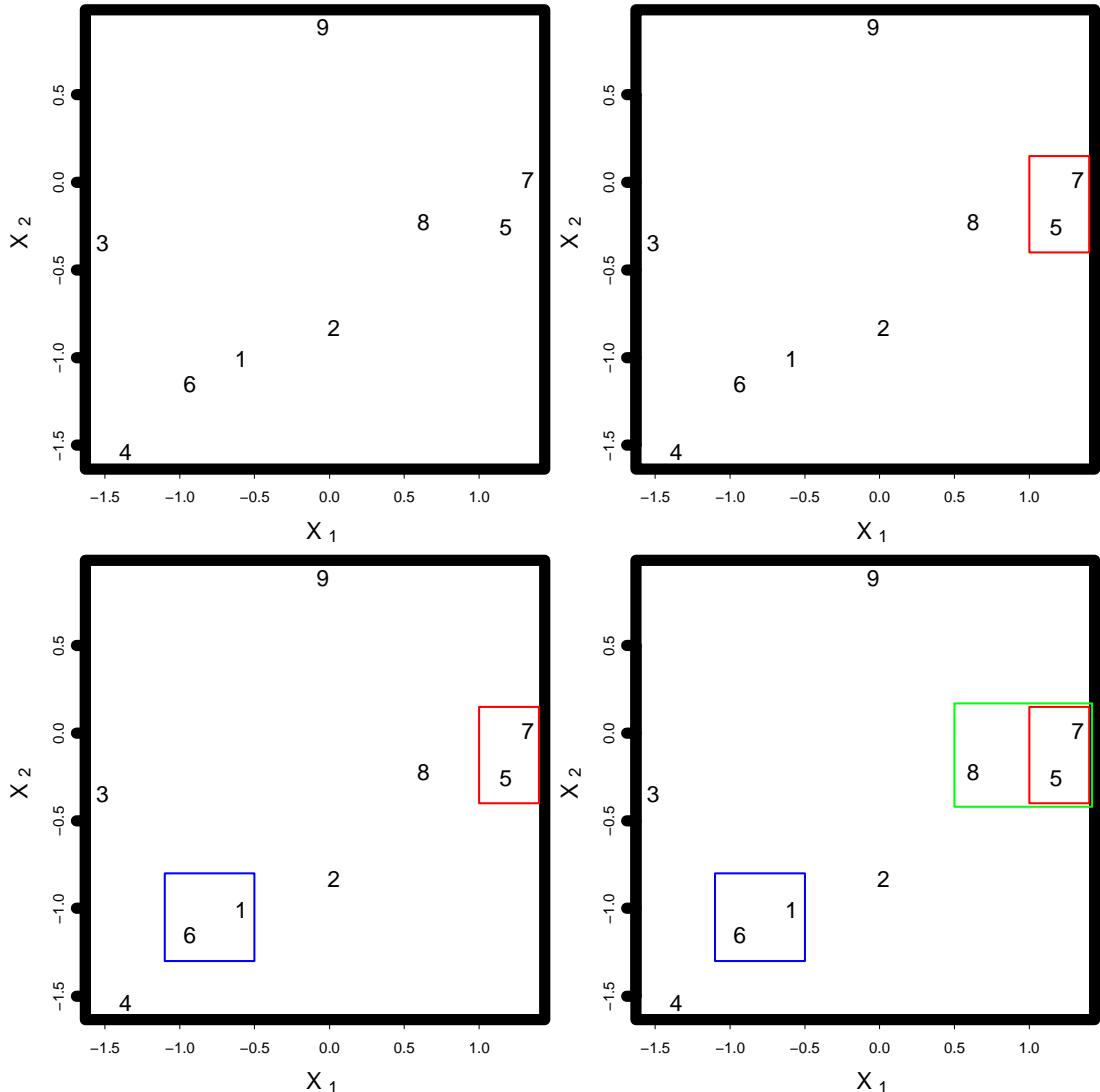
# Giải thuật (trộn các cụm)

Phân cụm bằng cấu trúc cây:

- Khởi tạo với mỗi điểm là 1 cụm riêng biệt ( $n$  cụm), chính là 1 nút trong dendrogram
- Tính toán độ tương tự (gần) giữa các điểm/cụm
- Hợp nhất 2 cụm mà chúng có độ tương tự cao nhất, ta còn lại  $n-1$  cụm
- Hợp nhất 2 cụm tiếp theo có độ tương tự cao nhất, ta còn lại  $n-2$  cụm
- Quá trình trên tiếp tục cho đến khi chỉ còn 1 cụm (là nút gốc trong dendrogram)

# Ví dụ

Bắt đầu với 9 cụm  
Hợp nhất 5 và 7  
Hợp nhất 6 và 1  
Hợp nhất cụm (5,7) với  
8.  
Quá trình tiếp tục cho  
đến khi tất cả các cụm  
được hợp nhất.



# Ta định nghĩa sự khác biệt ntn?

Việc triển khai phương pháp phân cấp cần giải quyết vấn đề khá hiển nhiên, đó là làm sao để định nghĩa sự khác biệt (dissimilarity) hoặc mối liên kết (linkage) giữa cụm hợp nhất (5, 7) và cụm 8?

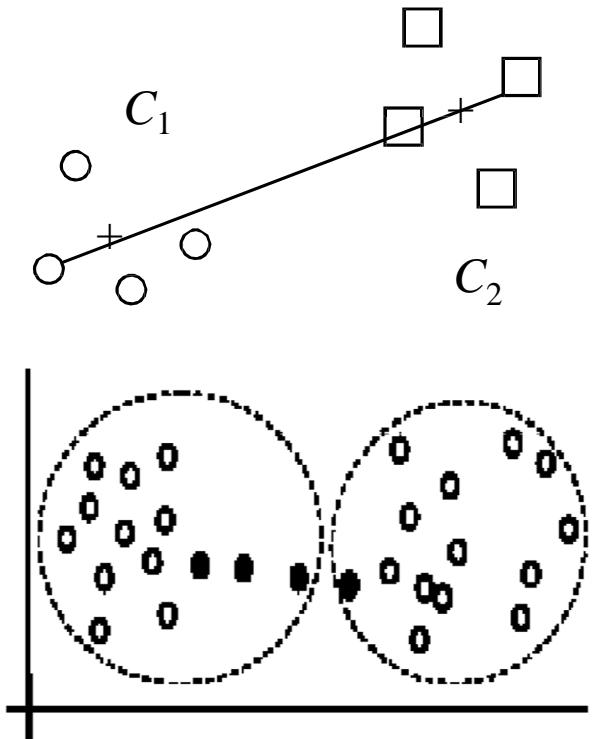
Có 4 lựa chọn:

- Liên kết đầy (Complete Linkage)
- Liên kết đơn (Single Linkage)
- Liên kết trung bình giữa các nhóm (Average Linkage)
- Liên kết tâm (Centroid Linkage)

# Các phương pháp liên kết

Liên kết đầy: Khoảng cách giữa 2 cụm là khoảng cách lớn nhất giữa 2 mẫu tương ứng của 2 cụm đó

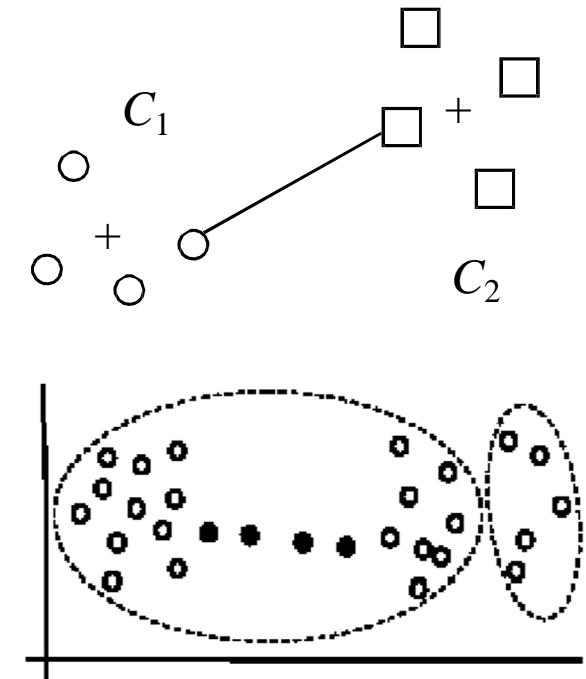
- Nhạy cảm (gặp lỗi phân cụm) đối với các ngoại lai (outliers)
- Có xu hướng sinh ra các cụm có dạng “bụi cây” (clumps)



[Liu, 2006]

# Các phương pháp liên kết

Liên kết đơn: Khoảng cách giữa 2 cụm là khoảng cách nhỏ nhất giữa các mẫu (các thành viên) của 2 cụm đó. Có xu hướng sinh ra các cụm có dạng “chuỗi dài” (long chain)



[Liu, 2006]

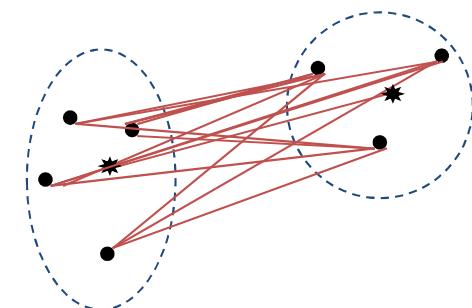
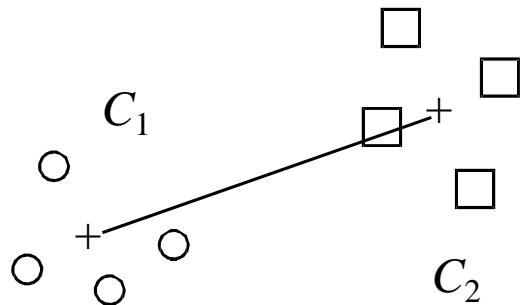
# Các phương pháp liên kết

**Liên kết trung bình**: Khoảng cách trong liên kết trung bình (Average-link) là sự thỏa hiệp giữa các khoảng cách trong liên kết hoàn toàn (Complete-link) và liên kết đơn (Single-link)

- Để giảm mức độ nhạy cảm (khả năng lỗi) của phương pháp phân cụm dựa trên liên kết đầy đối với các ngoại lai (outliers)
- Để giảm xu hướng sinh ra các cụm có dạng “chuỗi dài” của phương pháp phân cụm dựa trên liên kết đơn (dạng “chuỗi dài” không phù hợp với khái niệm tự nhiên của một cụm)
- Khoảng cách giữa 2 cụm là khoảng cách trung bình của tất cả các cặp mẫu (mỗi mẫu thuộc về một cụm)

# Các phương pháp liên kết

Liên kết tâm: Khoảng cách giữa các tâm của các mẫu tương ứng

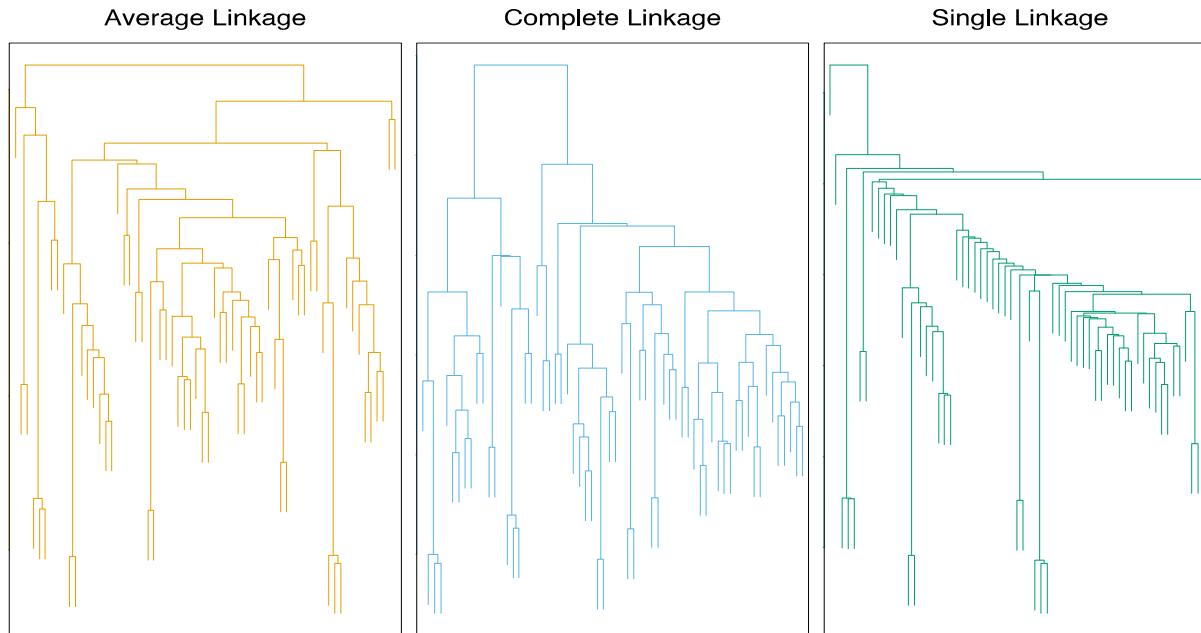


# Mối liên kết rất quan trọng

Dưới đây ta có 3 kết quả phân cụm trên cùng 1 bộ dữ liệu

Phương pháp tính mối liên kết khác nhau nhưng kết quả đem lại rất khác xa nhau

Phương pháp liên kết đầy và liên kết trung bình đường như có cỡ cụm như nhau, tuy nhiên liên kết đơn lại cho số cụm nhiều hơn vì mỗi lá của cây được hợp nhất từng lần một



# Dữ liệu của Fisher

S.Length	S.Width	P.Length	P.Width	Species
5.1	3.5	1.4	0.2	I.setosa
4.9	3	1.4	0.2	I.setosa
4.7	3.2	1.3	0.2	I.setosa
4.6	3.1	1.5	0.2	I.setosa
5	3.6	1.4	0.2	I.setosa
5.4	3.9	1.7	0.4	I.setosa
4.6	3.4	1.4	0.3	I.setosa
5	3.4	1.5	0.2	I.setosa
4.4	2.9	1.4	0.2	I.setosa
5.1	3.8	1.9	0.4	I.setosa
4.8	3	1.4	0.3	I.setosa
5.1	3.8	1.6	0.2	I.setosa
4.6	3.2	1.4	0.2	I.setosa
5.3	3.7	1.5	0.2	I.setosa
5	3.3	1.4	0.2	I.setosa
7	3.2	4.7	1.4	I.versicolor
6.4	3.2	4.5	1.5	I.versicolor
6.9	3.1	4.9	1.5	I.versicolor
5.5	2.3	4	1.3	I.versicolor
6.5	2.8	4.6	1.5	I.versicolor
5.7	2.8	4.5	1.3	I.versicolor
6.3	3.3	4.7	1.6	I.versicolor
4.9	2.4	3.3	1	I.versicolor
6.6	2.9	4.6	1.3	I.versicolor
5.2	2.7	3.9	1.4	I.versicolor
5	2	3.5	1	I.versicolor
6.2	2.8	4.8	1.8	I.virginica
6.1	3	4.9	1.8	I.virginica
6.4	2.8	5.6	2.1	I.virginica
7.2	3	5.8	1.6	I.virginica
7.4	2.8	6.1	1.9	I.virginica
7.9	3.8	6.4	2	I.virginica

```
data(iris)  
head(iris)
```

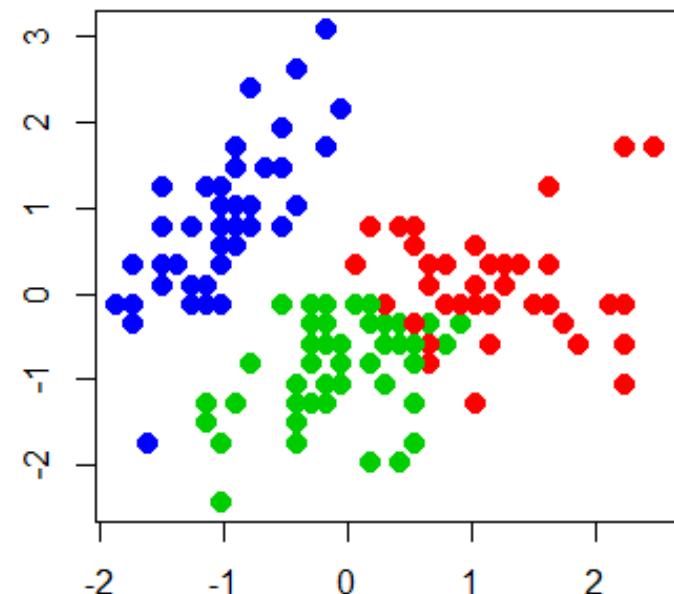
	S.Length	S.Width	P.Length	P.Width	Species
1	5.1	3.5	1.4	0.2	I.setosa
2	4.9	3.0	1.4	0.2	I.setosa
3	4.7	3.2	1.3	0.2	I.setosa
4	4.6	3.1	1.5	0.2	I.setosa
5	5.0	3.6	1.4	0.2	I.setosa
6	5.4	3.9	1.7	0.4	I.setosa



# Clustering in R

```
>data(iris)  
  
>head(iris)  
  
>s.iris <- scale(iris[1:4], center =  
TRUE, scale = TRUE)  
  
>km.res = kmeans(s.iris, 3, nstart = 25)  
  
>km.res$cluster  
  
>plot(s.iris, col=(km.res$cluster +1),  
main="K-Means Clustering+ Results with  
K=3", xlab=" ", ylab=" ", pch=20, cex=2)  
  
>km.res
```

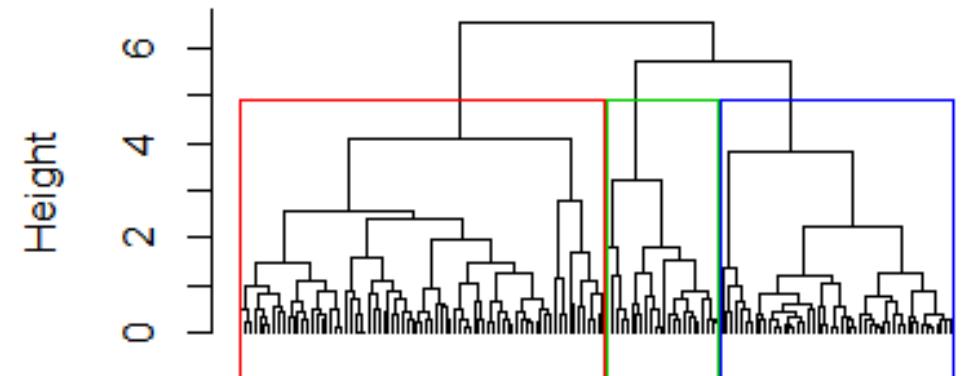
**K-Means Clustering+ Results with K=3**



# Clustering in R

```
>data(iris)  
  
>head(iris)  
  
>s.iris <- scale(iris[1:4], center = TRUE, scale = TRUE)  
  
>km.res = kmeans(s.iris, 3, nstart = 25)  
  
>dist = dist(s.iris, method = "euclidean")  
  
>hc = hclust(dist, method = "complete")  
  
>plot(hc, labels = FALSE, hang = -1)  
  
>rect.hclust(hc, k = 3, border = 2:4)
```

**Cluster Dendrogram**



# Tóm lược

- Phân tích cụm: Phương pháp rất hữu hiệu để "nhận dạng" nhóm dựa vào các biến quan sát
- Chủ yếu dựa vào khoảng cách giữa các nhóm dựa trên dữ liệu thực tế
- Một phương pháp rất có ích trong thời đại "Big Data"

# Câu hỏi?