กำหนดวิธีคลัสเตอร์ริ่ง

วิธีการคลัสเตอร์ริ่งที่อยู่บนรากฐาน
ของวิธีวัดความคล้ายคลึงกัน
ระหว่าง Object ที่จะถูกแบ่งกลุ่ม

ตัวอย่าง

$$sim(Q, D_i) = \frac{\sum_{j=1}^{t} w_{qj} * w_{d_{ij}}}{\sqrt{\sum_{j=1}^{t} (w_{qj})^2 * \sum_{j=1}^{t} (w_{d_{ij}})^2}}$$

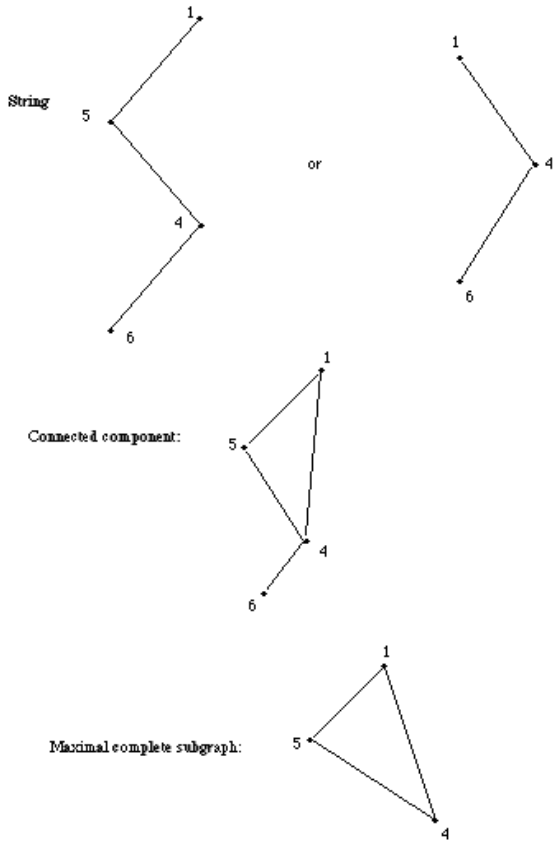
$Q = (0.4, 0.8)$ D2 =(0.2,0.7)

$$sim(Q, D_2) = \frac{(0.4*0.2)+(0.8*0.7)}{\sqrt{[(0.4)^2+(0.8)^2]*[(0.2)^2+(0.7)^2]}}$$

$$= \frac{0.64}{\sqrt{0.42}} = 0.98$$

threshold

- กราฟที่ได้นั้นจะสอดคล้องกับค่าความคล้ายคลึงที่กำหนด เราเรียกว่า threshold จะทำให้ออปเจ็กต์ 2 ออปเจ็กต์ที่มีค่าความคล้ายคลึงกันมีค่าเท่ากันหรือมากกว่าถูกเชื่อมโยงเป็นโหนด 2 โหนดในกราฟที่มีการเชื่อมโยงกัน



Graph Theoretic Method

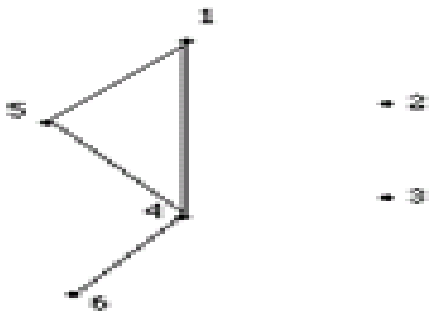
Objects: (1,2,3,4,5,6)

Similarity matrix:

	1	2	3	4	5	6
1						
2	.6					
3	.6	.8				
4	.9	.7	.7			
5	.9	.8	.8	.9		
6	.5	.5	.5	.9	.5	

Threshold: .89

Graph:



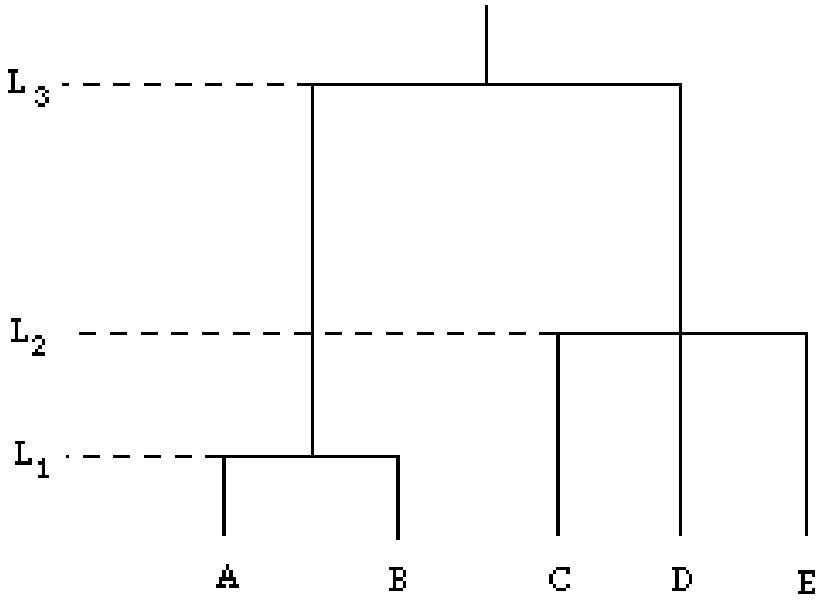
แสดงสัมประสิทธิ์ของความคล้ายคลึงกันของ 6 ออปเจ็กต์ที่สามารถใช้ดึงออปเจ็กต์ที่มีความคล้ายคลึงกันได้

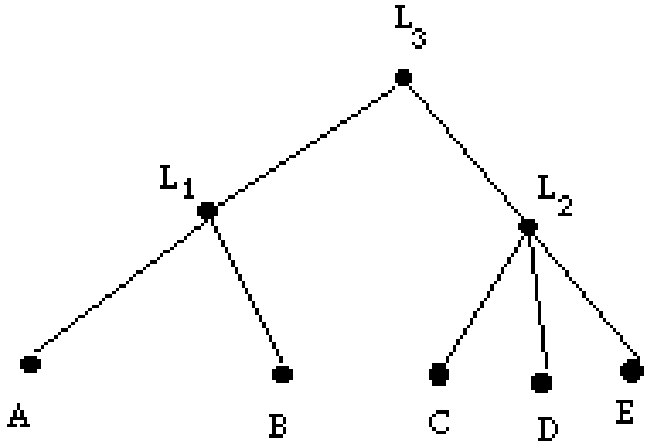
Single Link

เป็นวิธีการคลัสเตอร์ริ่งที่ทำให้เกิด Hierarchic Cluster ซึ่งใช้การคำนวณค่าความสัมพันธ์ของ Objects อัลกอริทึมนี้อยู่บนพื้นฐานของสัมประสิทธิ์ความไม่คล้ายคลึงกันของข้อมูลนำเข้า (dissimilarity coefficient)

ผลลัพธ์ที่ได้เป็นลำดับขั้น ของระดับตัวเลขที่มีส่วนร่วมกัน เรียกว่า dendrogram ในลักษณะโครงสร้างต้นไม้ (tree structure) ซึ่งแต่ละโหนดแทนหนึ่งคลัสเตอร์

Single Link





รูปแสดงถึง **single-link clusters** ที่ดึงจากสัมประสิทธิ์ความไม่คล้ายคลึงกันจากการกำหนด **thresholding**

Dissimilarity matrix:

2	.4			
3	.4	.2		
4	.3	.3	.3	
5	.1	.4	.4	.1
	1	2	3	4

Binary matrices:

2	0			
3	0	0		
4	0	0	0	
5	1	0	0	1
	1	2	3	4

Threshold = .1

2	0			
3	0	1		
4	0	0	0	
5	1	0	0	1
	1	2	3	4

Threshold = .2

2	0			
3	0	1		
4	1	1	1	
5	1	0	0	1
	1	2	3	4

Threshold = .3

Graphs and clusters:

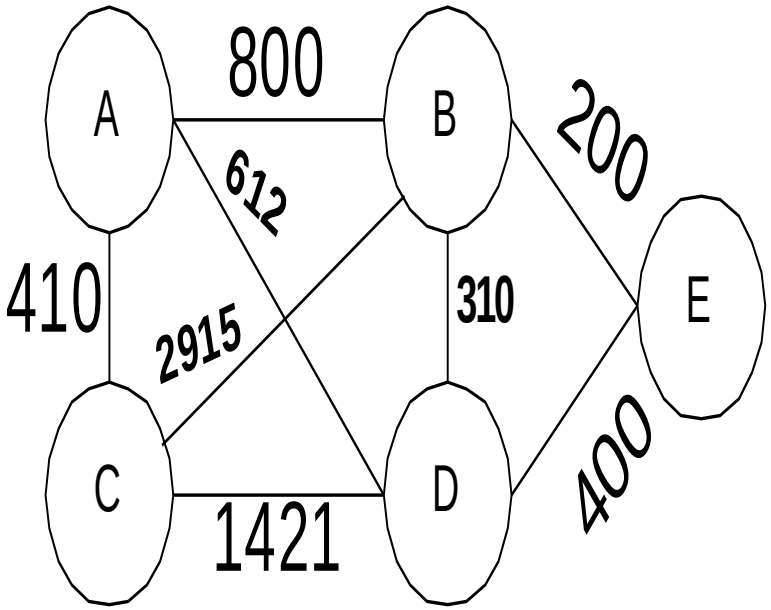
Minimum Spanning Tree หรือ MST

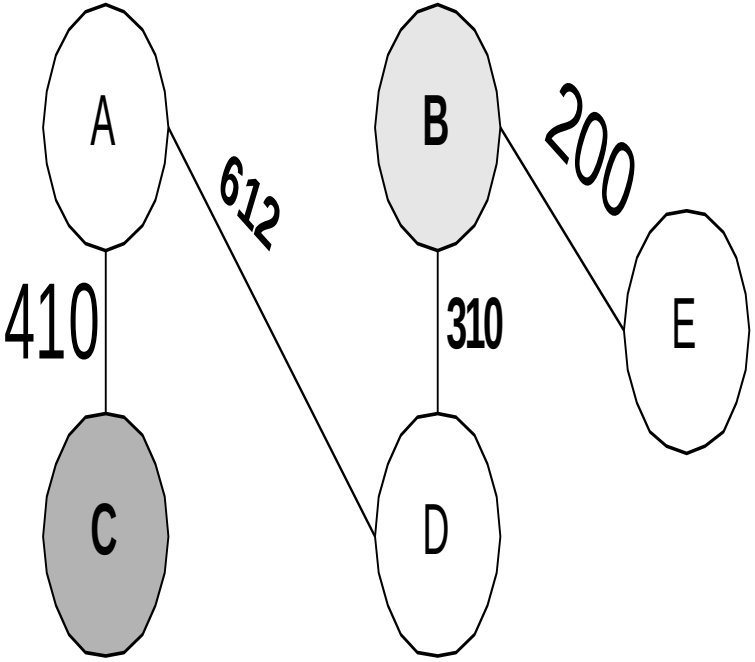
- ต้นไม้ที่ได้จากสัมประสิทธิ์ความไม่คล้ายคลึงกันเหมือนกับ Single-link tree
- ความแตกต่างที่เห็นได้ชัดเจนคือ โหนดของ Single-linked tree นั้น จะแทนคลัสเตอร์ แต่ Minimum Spanning Tree นั้นจะแทนออปเจ็กต์แต่ละตัวที่ถูกรวมกลุ่ม

Minimum Spanning Tree หรือ MST

- คือต้นไม้ที่มีความยาวในการเชื่อมโยงแต่ละออปเจ็กต์ที่มีค่าน้อยที่สุด ซึ่งความยาวนี้อาจเป็นน้ำหนักของการเชื่อมโยงในต้นไม้นั้น
- เราจะสามารถดึง single-link hierarchy จากการประมวลผลของ thresholding ก็ตามแต่เราไม่สามารถย้อนกลับได้ ซึ่ง MST จะมีประโยชน์มากกว่าในการเชื่อมโยงเพราะเป็นอิสระในการทำงาน

Minimum Spanning Tree หรือ MST





ตัวอย่าง

- ชุดเอกสารประกอบด้วยเอกสาร D1, D2, D3 เอกสารแต่ละฉบับผ่านการตัดคำ (word segmentation) และดึงคำหยุดออกไป ดังนั้นเหลือคำสำคัญดังนี้

D_1: คอมพิวเตอร์ สารสนเทศ คอมพิวเตอร์ คอมพิวเตอร์

D_2: อินเตอร์เน็ต คอมพิวเตอร์ อินเตอร์เน็ต ข้อมูล

D_3: ระบบ อินเตอร์เน็ต

ตัวอย่าง

D_1: คอมพิวเตอร์ สารสนเทศ คอมพิวเตอร์ คอมพิวเตอร์

D_2: อินเตอร์เน็ต คอมพิวเตอร์ อินเตอร์เน็ต ข้อมูล

D_3: ระบบ อินเตอร์เน็ต

จากนั้นหาความถี่ของคำที่ไม่ซ้ำกันในเอกสารแต่ละฉบับ

ตารางที่ 1. ความถี่ของคำในชุดเอกสาร

เอกสาร	คำ	tf
D_1	คอมพิวเตอร์	3
D_1	สารสนเทศ	1
D_2	อินเตอร์เน็ต	2
D_2	คอมพิวเตอร์	1
D_2	ข้อมูล	1
D_3	ระบบ	1
D_3	อินเตอร์เน็ต	1

ตัวอย่าง

ตารางที่ 1. ความถี่ของคำในชุดเอกสาร

เอกสาร	คำ	*tf*
D_1	คอมพิวเตอร์	3
D_1	สารสนเทศ	1
D_2	อินเตอร์เน็ต	2
D_2	คอมพิวเตอร์	1
D_2	ข้อมูล	1
D_3	ระบบ	1
D_3	อินเตอร์เน็ต	1

ตารางที่ 2. ค่า *idf* ของคำในชุดเอกสาร

คำ	*df*	*idf*
T_1: คอมพิวเตอร์	2	0.18
T_2: สารสนเทศ	1	0.48
T_3: อินเตอร์เน็ต	2	0.18
T_4: ระบบ	1	0.48
T_5: ข้อมูล	1	0.48

คำนวณน้ำหนักของแต่ละเอกสาร

$$w_{ik} = tf_{ik} * \log(N / n_k)$$

$T_k = \text{term } k \text{ in document } D_i$

$tf_{ik} = \text{frequency of term } T_k \text{ in document } D_i$

$idf_k = \text{inverse document frequency of term } T_k \text{ in } C$

$N = \text{total number of documents in the collection } C$

$n_k = \text{the number of documents in } C \text{ that contain } T_k$

$$idf_k = \log\left(\frac{N}{n_k}\right)$$

ตัวอย่าง

ตารางที่ 2. ค่า *idf* ของคำในชุดเอกสาร

คำ	*df*	*idf*
T_1: คอมพิวเตอร์	2	0.18
T_2: สารสนเทศ	1	0.48
T_3: อินเตอร์เน็ต	2	0.18
T_4: ระบบ	1	0.48
T_5: ข้อมูล	1	0.48

เวกเตอร์เอกสารทั้งหมดแทนด้วยเมตริกซ์
โดยแถวของเมตริกซ์คือเอกสารทั้งหมด
และสดมภ์คือคำที่ไม่ซ้ำกันทั้งหมดในชุดเอกสาร

	T_1	T_2	T_3	T_4	T_5
D_1	0.74	0.67	0	0	0
D_2	0.57	0	0.29	0	0.77
D_3	0	0	0.35	0.94	0

เมตริกซ์เอกสารคำ ใช้เป็น input
ในขั้นตอนการจำกลุ่มเอกสารและ
นำไปหาค่าความคล้ายคลึงของเอกสารคู่หนึ่งๆได้

คำนวณหาความคล้ายคลึงกัน

	T_1	T_2	T_3	T_4	T_5
D_1	0.74	0.67	0	0	0
D_2	0.57	0	0.29	0	0.77
D_3	0	0	0.35	0.94	0

$$sim(Q, D_i) = \frac{\sum_{j=1}^{t} w_{qj} * w_{d_{ij}}}{\sqrt{\sum_{j=1}^{t} (w_{qj})^2 * \sum_{j=1}^{t} (w_{d_{ij}})^2}}$$

วิธีการคลัสเตอร์ริ่ง

จากคำอธิบาย(Descriptions)
ของ Object โดยตรง

cluster representative

- ซึ่งเราเรียกว่า cluster profile หรือ classification vector หรือ centroid
- ไม่มีการวัดความคล้ายคลึงกัน แต่เก็บโครงสร้างที่เหมาะสมโดยจำกัดจำนวนของคลัสเตอร์และขนาดของแต่ละคลัสเตอร์
- สรุปจากหลักเกณฑ์บางอย่างเพื่อเป็นตัวแทนของออปเจ็กต์ในกลุ่มซึ่งตัวแทนนี้จะมีความใกล้เคียงกับทุกๆออปเจ็กต์ในคลัสเตอร์โดยเฉลี่ยตามความรู้สึก

วิธีการคลัสเตอร์ริ่งที่กระทำได้จากคำอธิบาย(Descriptions) ของ Object

- ใช้ matching function บางครั้งเรียกว่า similarity หรือ correlation function
- อัลกอริทึมนี้จะใช้พารามิเตอร์จากการสังเกต (empirically) อันได้แก่
  - จำนวนของคลัสเตอร์ที่ต้องการ
  - ขนาดต่ำสุดและขนาดสูงสุดของแต่ละคลัสเตอร์
  - ค่าของ threshold บน matching function ซึ่งออปเจ็กต์ที่มีค่าต่ำกว่า threshold ที่กำหนดจะไม่ถูกรวมในคลัสเตอร์นี้
  - การควบคุมการซ้อนทับ(Overlap)ระหว่างคลัสเตอร์ .
  - พารามิเตอร์ที่เลือกอย่างไม่มีกฎเกณฑ์จะถูกปรับให้อยู่ในระดับที่ให้ผลที่ดีที่สุด
- การทำงานของอัลกอริทึมนี้จะกระทำซ้ำๆ เพื่อให้ได้ผลลัพธ์ที่เหมาะสมที่สุด

Rocchio

- วิธีของร็อคชิโอจะสร้างตัวแยกเอกสาร โดยคิดน้ำหนักของเทอม

<w1,..wn> กับกลุ่ม *ci* ด้วยสูตรการคำนวณ

$$w_{ki} = \left[\frac{B}{\left| \left\{ \vec{d}_j \mid ca_{ij} = 1 \right\} \right|} \times \sum_{\{\vec{d}_j \mid ca_{ij}=1\}} w_{kj} \right] - \left[\frac{Y}{\left| \left\{ \vec{d}_j \mid ca_{ij} = 0 \right\} \right|} \times \sum_{\{\vec{d}_j \mid ca_{ij}=0\}} w_{kj} \right]$$

โดยที่ w_{kj} เป็นน้ำหนักของเทอม t_k ซึ่งอยู่ในเอกสารฝึกฝน $\vec{d}_j$ ส่วน B และ Y เป็น

ค่าพารามิเตอร์ควบคุมความสัมพันธ์ของตัวอย่างที่อยู่ในกลุ่ม และไม่อยู่ในกลุ่ม

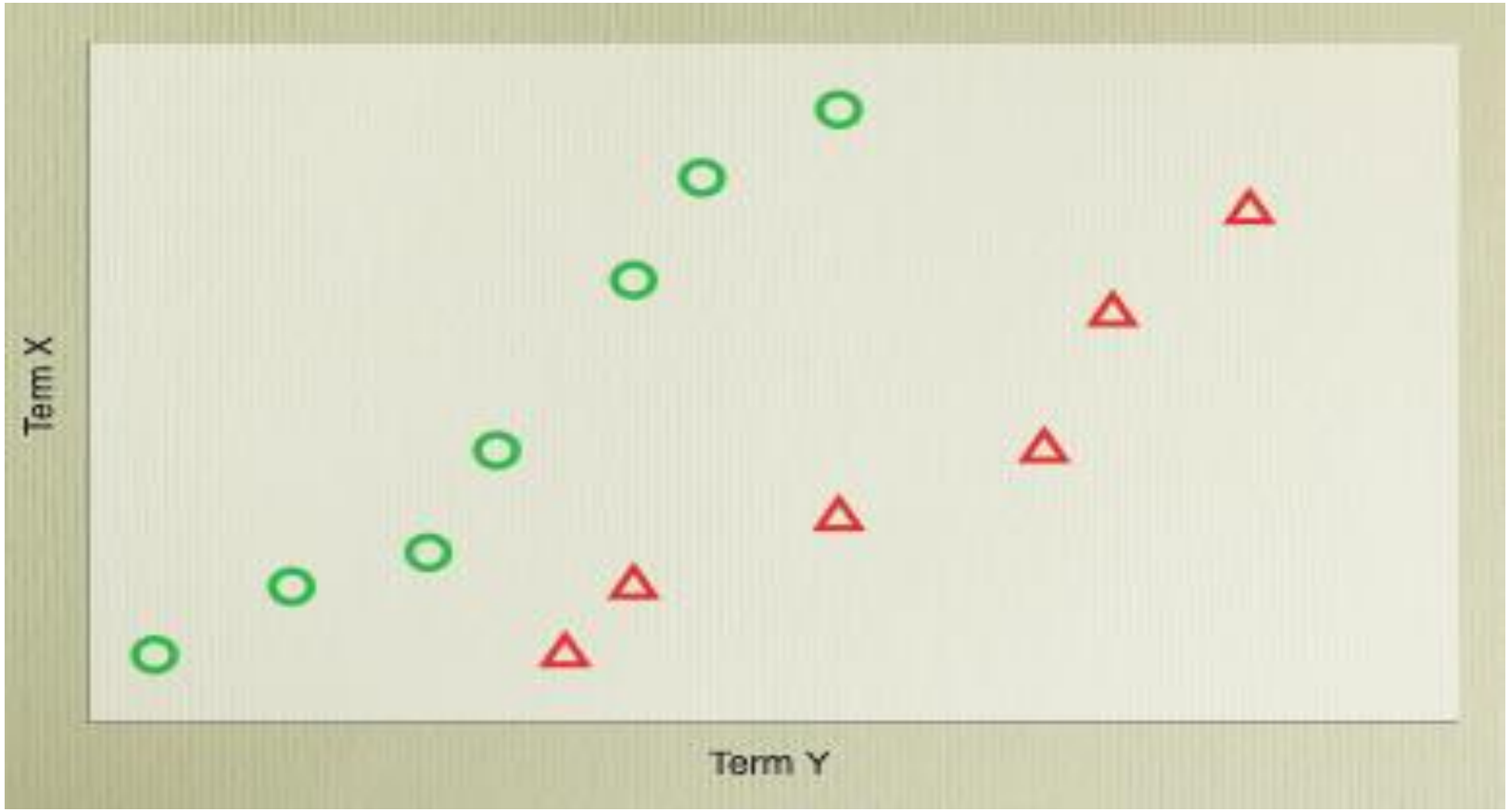
อัลกอริทึมของ Rocchio's clustering

- **ระยะแรก** เป็นระยะที่ทำการเลือกออปเจ็กต์จำนวนหนึ่งเป็นศูนย์กลางของคลัสเตอร์ สำหรับออปเจ็กต์ที่เหลือกำหนดให้เป็นกลุ่ม rag-bag
- กฎที่กำหนดขึ้นในเทอมของ thresholds บน matching function คลัสเตอร์สุดท้ายอาจจะมีการซ้อนทับกัน(overlap)

- **ระยะที่สอง** เป็นการกระทำขั้นตอนซ้ำๆ เพื่อที่จะหาพารามิเตอร์นำเข้า อาจเป็นขนาดของคลัสเตอร์

- **ระยะที่สาม** เป็นเป็นการพิจารณาออปเจ็กที่เหลือที่ไม่ได้ถูกกำหนด และมีการขจัดออปเจ็กต์ที่มีการซ้อนทับกันระหว่างคลัสเตอร์ ให้น้อยลง

Rocchio's clustering



Rocchio

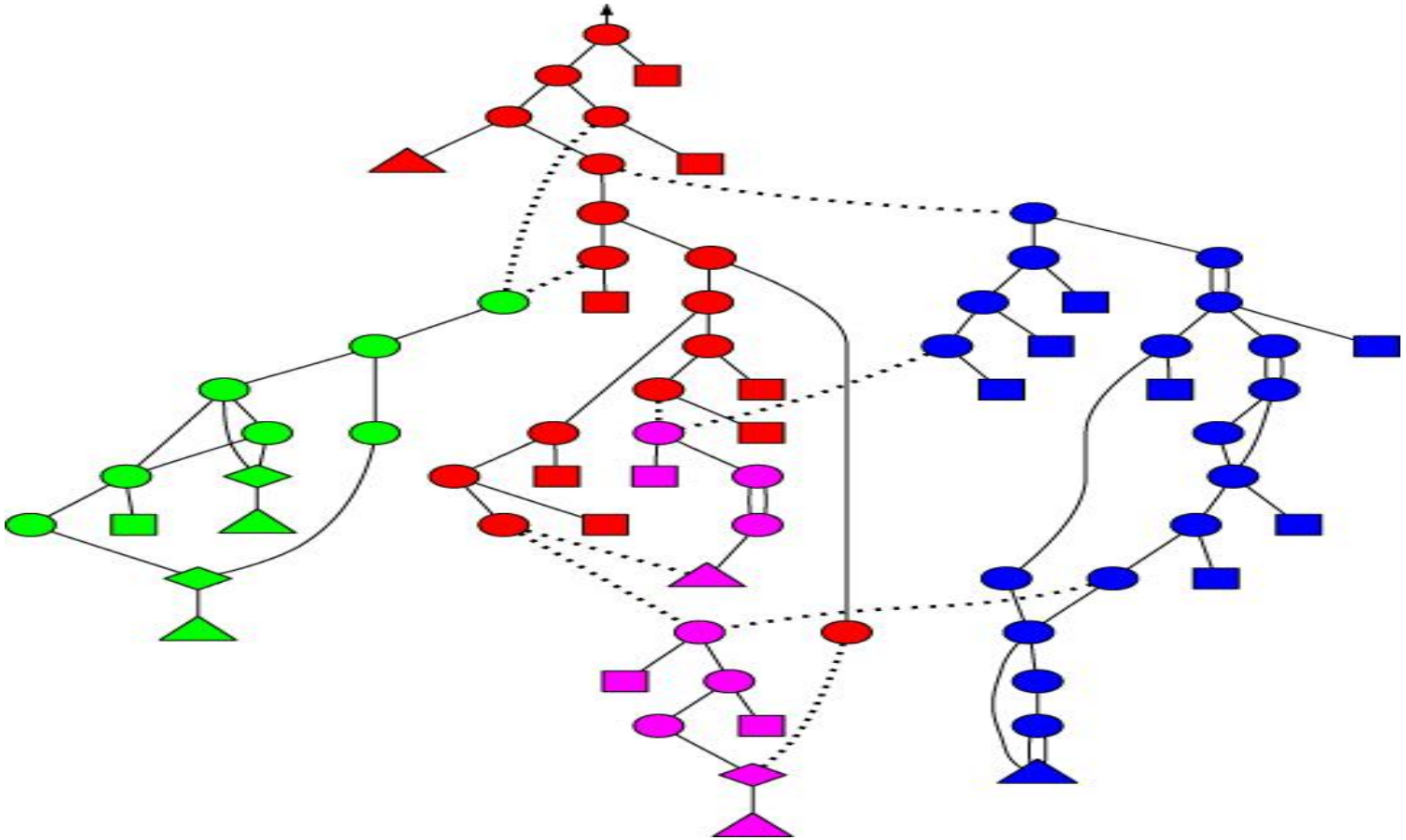
ในขั้นตอนการคัดแยกเอกสารจะนำเอาตัวแทนเอกสารไปเปรียบเทียบกับตัวแทนของกลุ่มฝึกฝน ถ้ามากกว่าค่าผ่านอ้างอิงก็สามารถจัดหมวดหมู่เอกสารได้

ข้อดีของวิธีนี้คือ ง่ายต่อการสร้างตัวจำแนกหมวดหมู่ สำหรับข้อเสียคือ ถ้าเอกสารในกลุ่มมีแนวโน้มไม่อยู่ในกลุ่ม จะทำให้การจำแนกหมวดหมู่ผิดพลาด

Single-Pass algorithm

- คำอธิบายออปเจ็กต์ถูกดำเนินการเป็นลำดับ
- ออปเจ๊กต์ตัวแรกจะถูกกำหนดให้เป็นตัวแทนคลัสเตอร์ของคลัสเตอร์แรก
- ออปเจ็กต์หลังจากนั้นจะถูกจับคู่กับตัวแทนคลัสเตอร์ทั้งหมด
- ถ้าออปเจ็กต์มีการซ้อนทับกัน จะมีการกำหนดให้ออปเจ็กต์มีค่าตามเงื่อนไขบน matching function
- เมื่อออปเจ็กต์ใดถูกกำหนดเป็นตัวแทนของคลัสเตอร์ จะมีการคำนวณใหม่
- ถ้าออปเจ็กต์ใดพลาดจากการทดสอบ(test) จะนำกลับไปเป็นตัวแทนคลัสเตอร์สำหรับคลัสเตอร์ใหม่
- ซึ่งการกระทำซ้ำๆนี้ การจัดกลุ่มสุดท้ายจะขึ้นกับพารามิเตอร์ข้อมูลเข้าที่เกิดจากการกำหนดจากการสังเกตของความแตกต่างของเซตออปเจ็กต์

Single-Pass algorithm



K-means

- ขั้นตอนวิธีการจัดกลุ่มโดยวิธีแบ่งกั้น(Partitioning) ระเบียนข้อมูลถูกแบ่งกั้นเป็นกลุ่มที่ไม่มีสมาชิกร่วมกันเลย โดยใช้การกั้นระหว่างกลุ่มด้วยระยะทาง

K-means

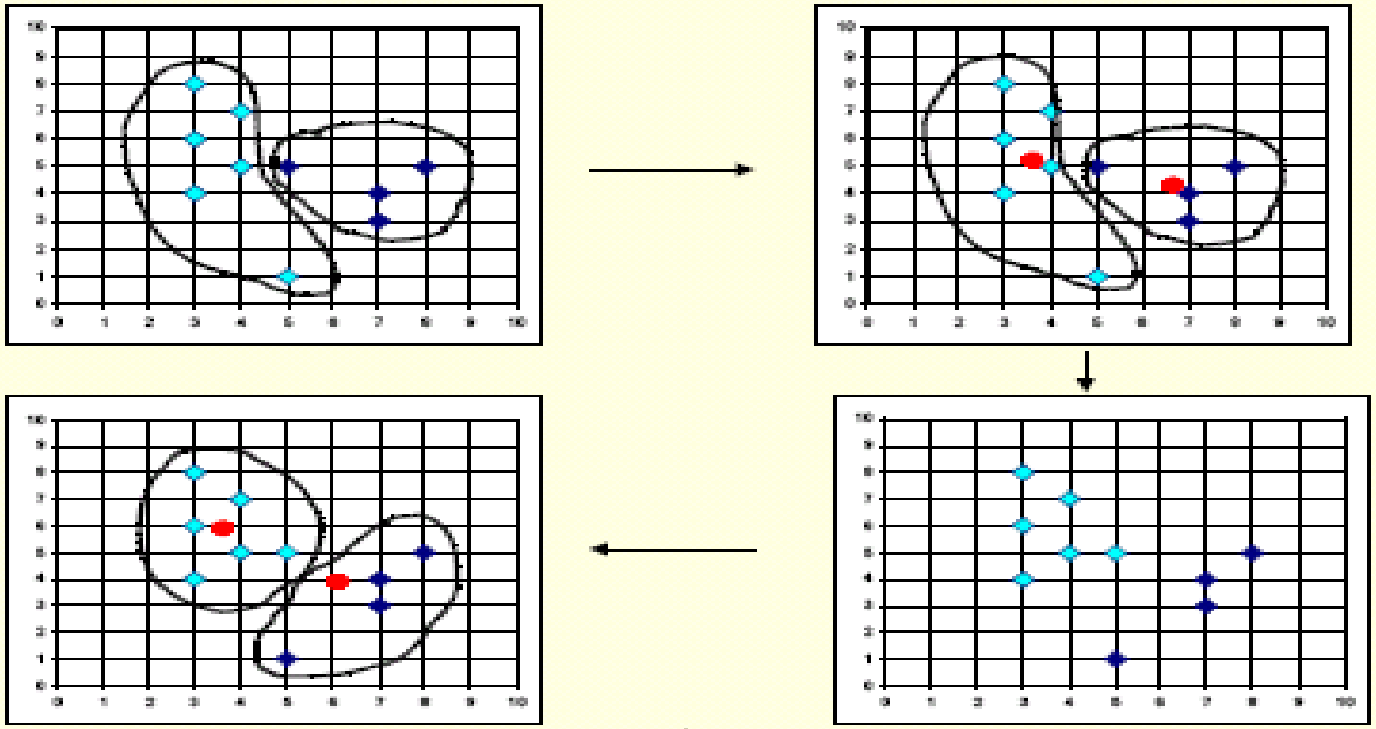
กำหนดให้ข้อมูล n ระเบียน แบ่งเป็น K กลุ่มที่ไม่มีสมาชิกร่วมกัน

วิธีการเกาะกลุ่มโดยใช้วิธีแบ่งกั้นมีขั้นตอนดังนี้

- แบ่งกลุ่มข้อมูลเป็น k กลุ่มที่ไม่ใช่เซตว่าง
- คำนวณจุดกึ่งกลาง(centroid) ของกลุ่ม โดยใช้ค่าเฉลี่ยเลขคณิต (mean)
- สำหรับแต่ละระเบียน นำระเบียนเทียบกับจุดกึ่งกลาง เพื่อกำหนดกลุ่มให้กับระเบียน โดยเลือกระยะจากระเบียนไปจุดกึ่งกลางที่ใกล้ที่สุด
- วนซ้ำจนกระทั่งไม่มีการเปลี่ยนกลุ่มของระเบียน หรือครบจำนวนรอบสูงสุดที่กำหนดไว้

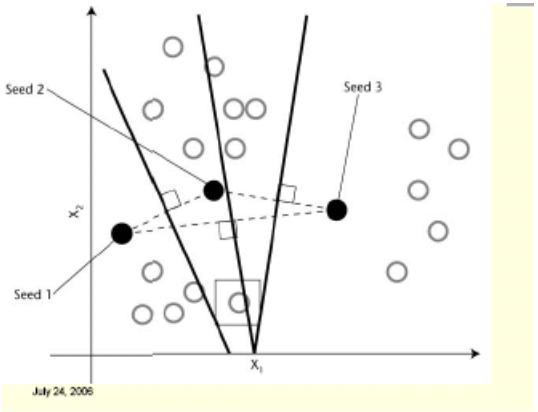
ตัวอย่าง การทำงานของขั้นตอนวิธี **k-mean**

ให้ $k = 2$, สีฟ้าแทนกลุ่ม 1, สีน้ำเงินแทนกลุ่ม 2 และสีแดงแทนค่ากึ่งกลาง

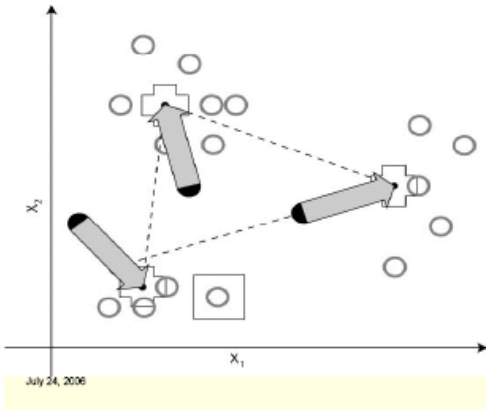


Cluster analysis

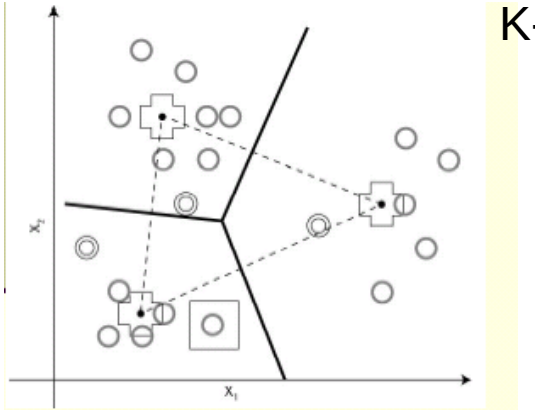
ตัวอย่าง การทำงานของขั้นตอนวิธี k-mean



K-means clustering1



K-means clustering2



K-means clustering3

k-mean

ข้อดี

ประสิทธิภาพมีค่าเท่ากับ O(tkn)

เมื่อ n คือจำนวนข้อมูล

k คือจำนวนกลุ่ม

t คือจำนวนรอบที่ต้องวนซ้ำ

โดยปกติแล้วค่า k และ t จะน้อยกว่า n

ผลลัพธ์ที่ได้เป็นผลเฉลยเฉพาะที่(local optimal) ถ้าต้องการผลเฉลยที่ให้ค่าที่ดีที่สุด(global optimal) จำเป็นต้องใช้วิธีอื่นช่วย

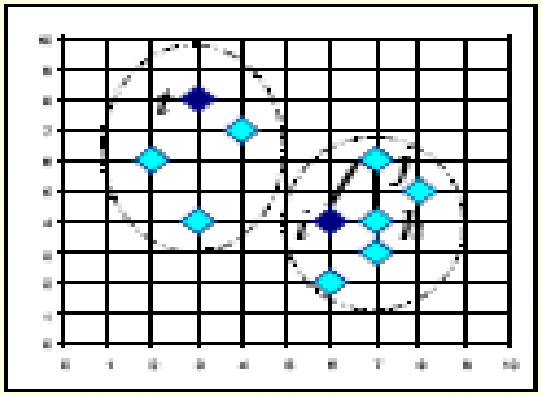
ข้อเสีย

1. ใช้ได้เฉพาะลักษณะประจำที่เป็นจำนวน เพราะต้องใช้ค่าเฉลี่ย
2. ต้องมีการกำหนดค่า k ก่อนเริ่มขั้นตอนวิธี
3. ถ้าข้อมูลผิดปกติจะทำให้ค่าเฉลี่ยมีค่าที่มากหรือน้อยเกินไป
4. ไม่เหมาะกับข้อมูลที่มีลักษณะกลุ่มที่ไม่ใช่เซตนูน(non-convex shapes)

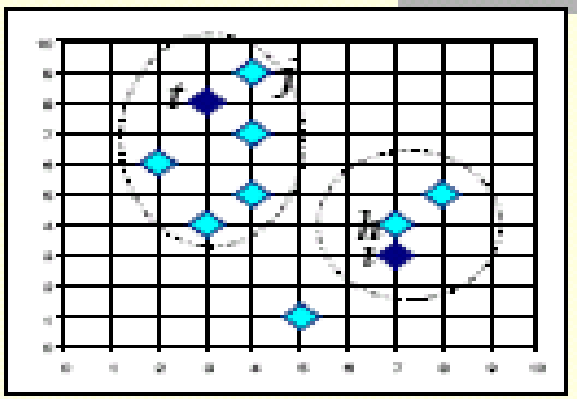
PAM (Partitioning Around Medoids,1987)

- เลือกข้อมูล k ตัวใช้เป็นตัวแทนข้อมูลอย่างสุ่ม
- สำหรับแต่ละคู่ของข้อมูล h ที่ไม่ใช่ medoid กับ iที่เป็น medoid
- คำนวณค่าการสลับ TCih
  - ถ้า TCih<0 ให้แทน iด้วย h กล่าวคือมีการเปลี่ยนตัวแทน
- ทำซ้ำจนกระทั่งไม่มีการเปลี่ยนกลุ่มของ medoid อีก

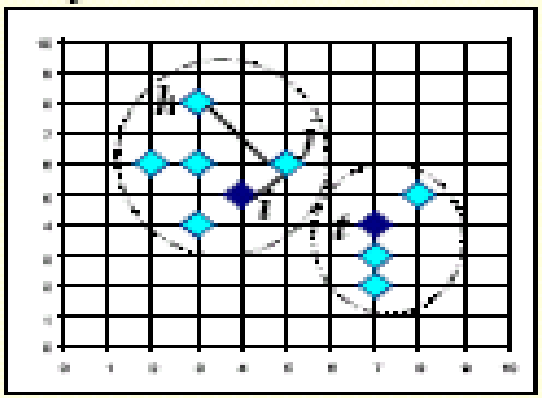
PAM Clustering: $TC_{ih}=\Sigma_j C_{jih}$



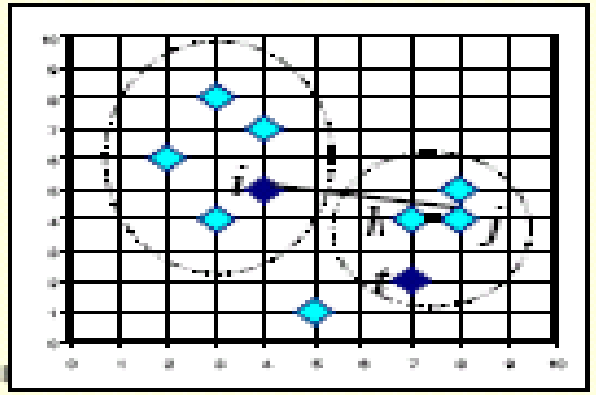
$C_{jih} = d(j, h) - d(j, i)$



$C_{jih} = 0$

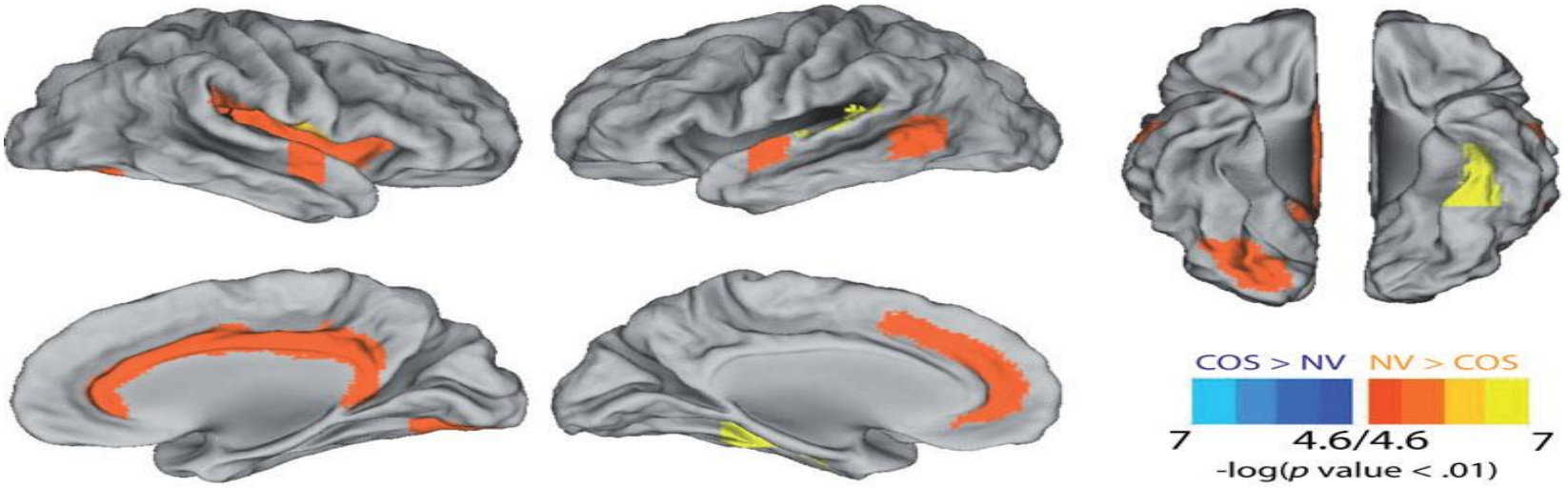


$C_{jih} = d(j, t) - d(j, i)$

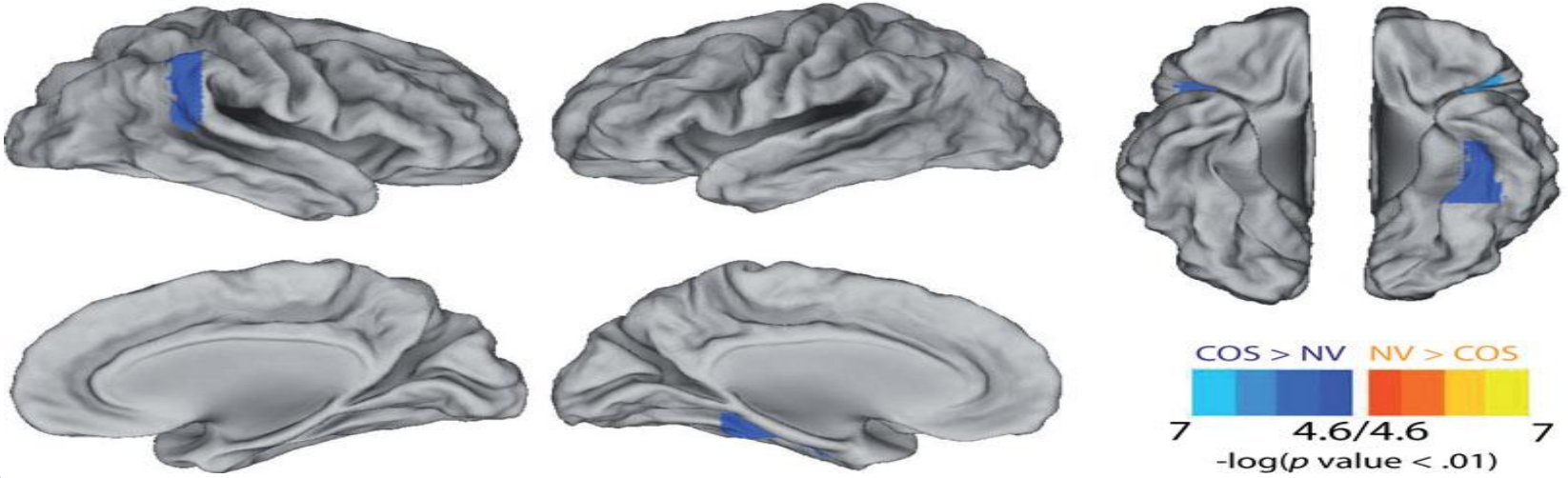


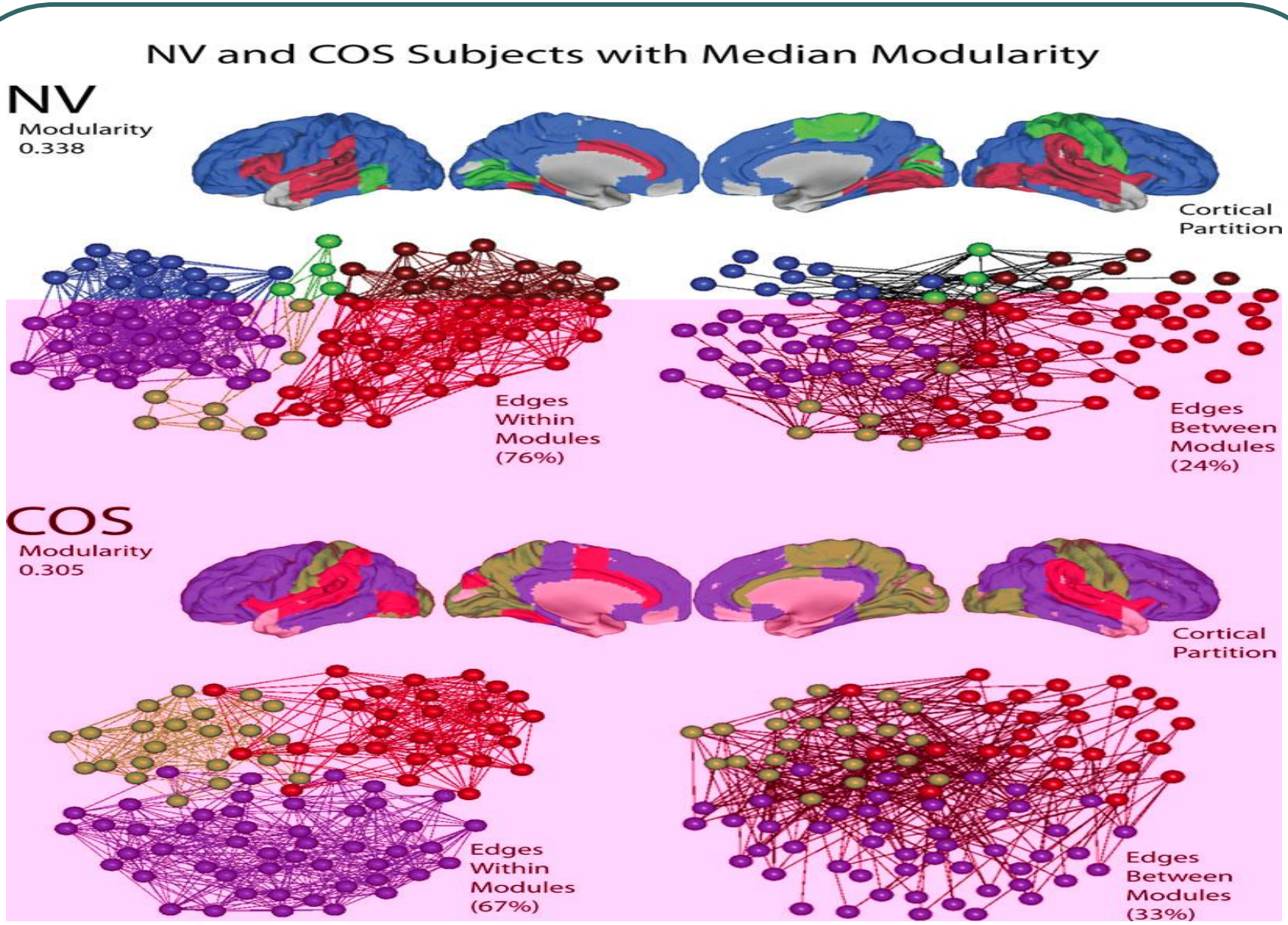
$C_{jih} = d(j, h) - d(j, t)$

Regional Differences in Clustering between COS and NV



Regional Differences in Efficiency between COS and NV




NV and COS Subjects with Median Modularity
NV
Modularity
0.338
Cortical
Partition
Edges
Within
Modules
(76%)
Edges
Between
Modules
(24%)
COS
Modularity
0.305
Cortical
Partition
Edges
Within
Modules
(67%)
Edges
Between
Modules
(33%)

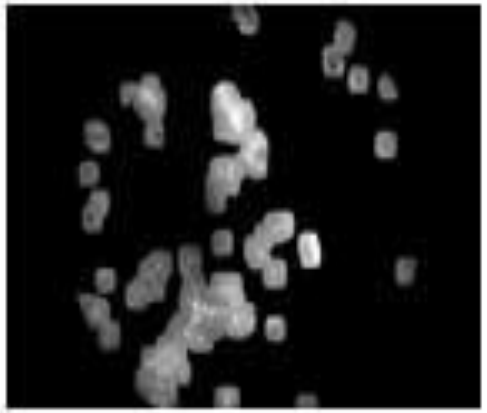
Fuzzy C-means(FCM)

- เป็นอัลกอริทึมที่ยอมให้ข้อมูลในแต่ละคลัสเตอร์มีการซ้อนทับกันหรือซ้ำกันได้
- วิธีการนี้เป็นการจัดกลุ่มที่มีใช้อย่างแพร่หลายในงานด้านต่างๆเช่น การแพทย์ วิทยาศาสตร์ วิศวกรรมศาสตร์
- โดยอาศัยการให้ค่าการเป็นสมาชิกของข้อมูลต่อกลุ่มข้อมูลต่างๆ การได้มาซึ่งค่าการเป็นสมาชิกส่วนหนึ่งมาจากการวัดระยะทางระหว่างข้อมูลและจุดศูนย์กลางของกลุ่มเหล่านั้น

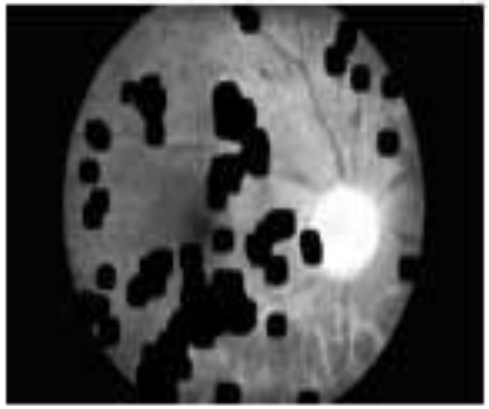
Fuzzy C-means(FCM)

ขั้นตอนการทำงานของฟัซซี่ซีมีน(fuzzy C-Means) ประกอบด้วย

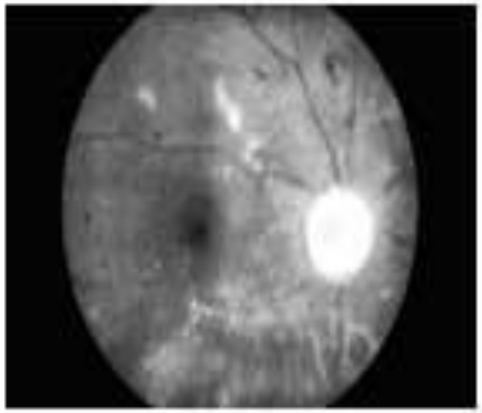
- กำหนดกลุ่มข้อมูลที่ต้องการจัดกลุ่ม เพื่อกำหนดค่าเพื่อเป็นเงื่อนไขในการให้ข้อมูลหยุดการจัดกลุ่ม() กำหนดค่าฟัซซี่พารามิเตอร์ (m) ซึ่งต้องมากกว่าหนึ่ง และกำหนดจุดศูนย์กลางเริ่มต้นของข้อมูล
- คำนวณค่าการเป็นสมาชิกของข้อมูลต่อกลุ่มข้อมูลต่างๆ
- คำนวณจุดศูนย์กลางกลุ่มข้อมูลใหม่และตรวจสอบเงื่อนไขโดยตรวจสอบค่าการเป็นสมาชิกใหม่ลบค่าการเป็นสมาชิกก่อนหน้า
- ถ้าเงื่อนไขเป็นจริงคำนวณค่าการเป็นสมาชิกและ objective function ถ้าเงื่อนไขเป็นเท็จ คำนวณค่าการเป็นสมาชิกจากจุดศูนย์กลางล่าสุด(วนรอบ)



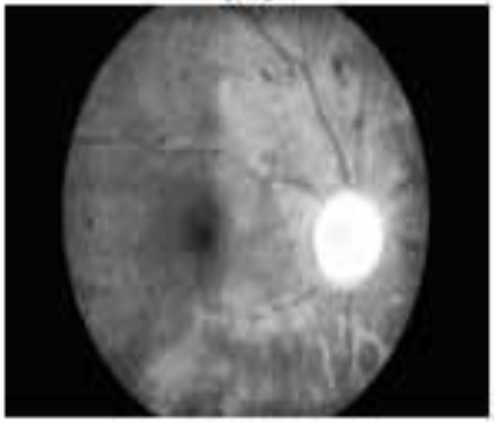
(A)



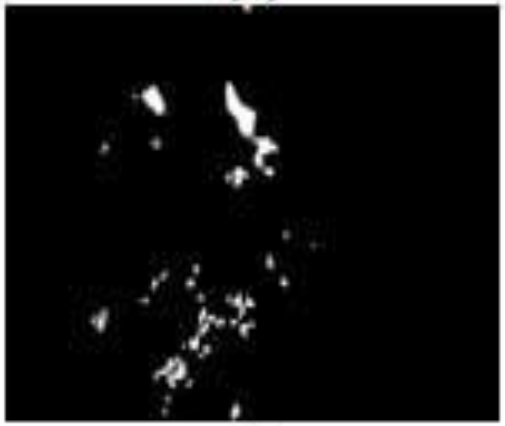
(B)



(C)



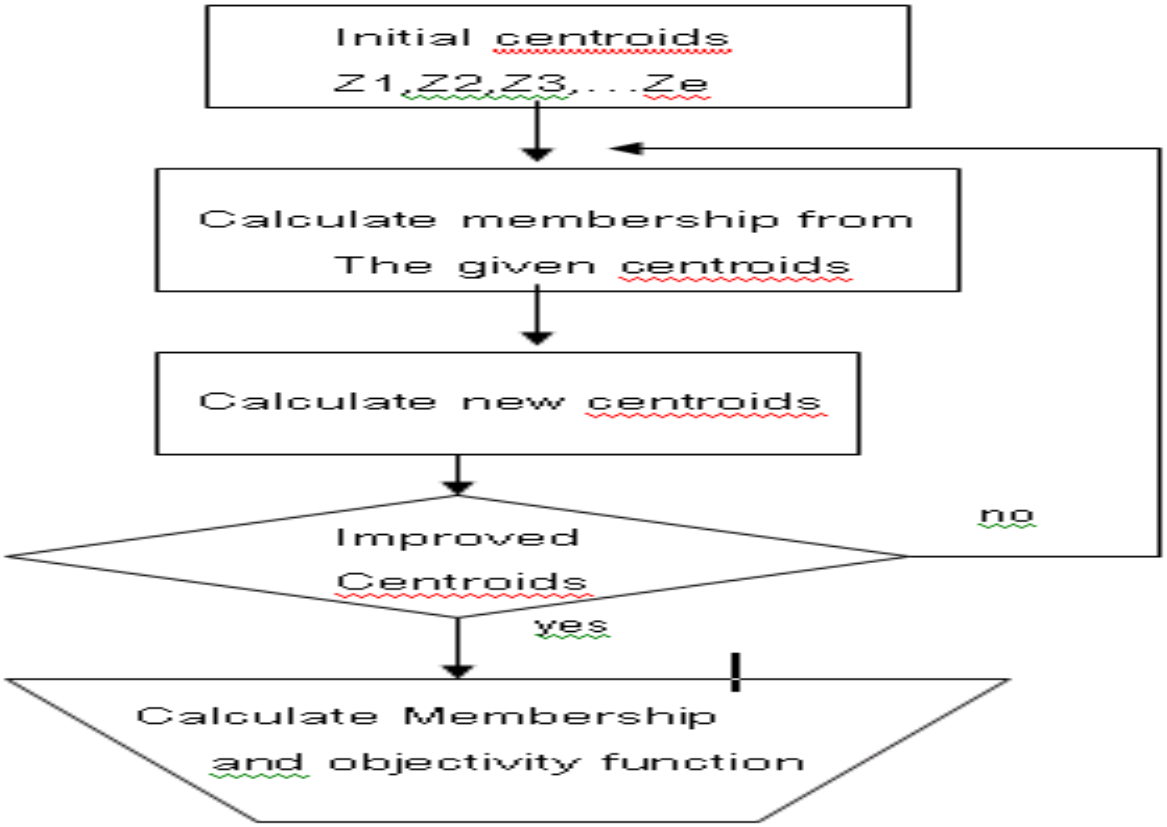
(D)



(E)

(F)

Fuzzy C-means(FCM)



วิธีการวัดระยะทาง

Euclidean distance

$$EDji = \sqrt{(X_j - Z_i)(X_j - Z_i)^T}$$

โดย EDji แทนระยะทางแบบยูคลิเดียนระหว่างข้อมูล X ที่ j และจุดศูนย์กลางข้อมูล Z กลุ่มที่ i และ T แทน Transpose matrix

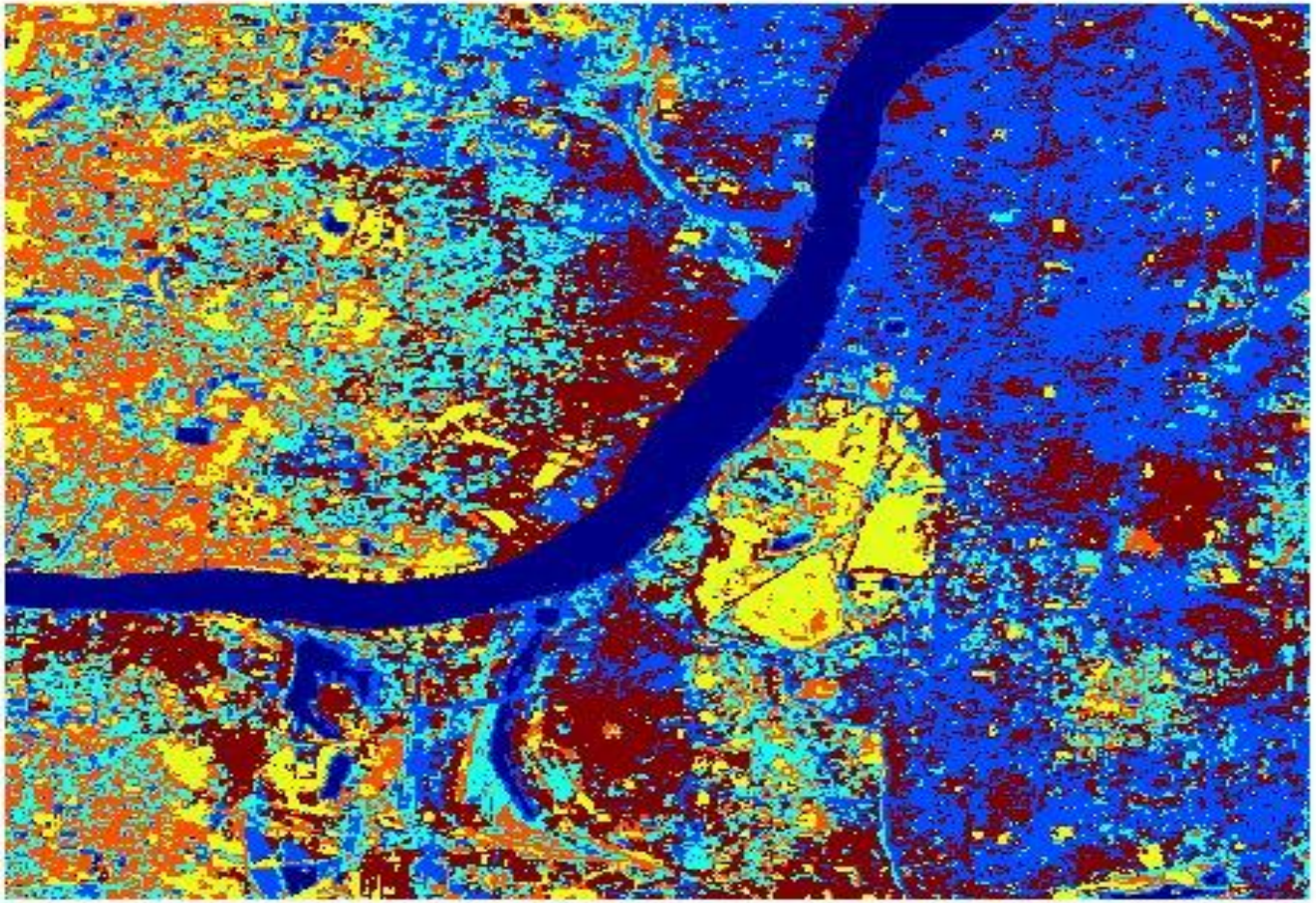
Mahalanobis distance

$$MDji = \sqrt{(X_j - Z_i)A^{-1}(X_j - Z_i)^T}$$

โดย MDji แทนระยะทางแบบมหาลาโนบิส ระหว่างข้อมูล X ตัวที่ j และจุดศูนย์กลางข้อมูล Z กลุ่มที่ i

A คือ variance-covariance matrix คำนวณจากสมการดังนี้

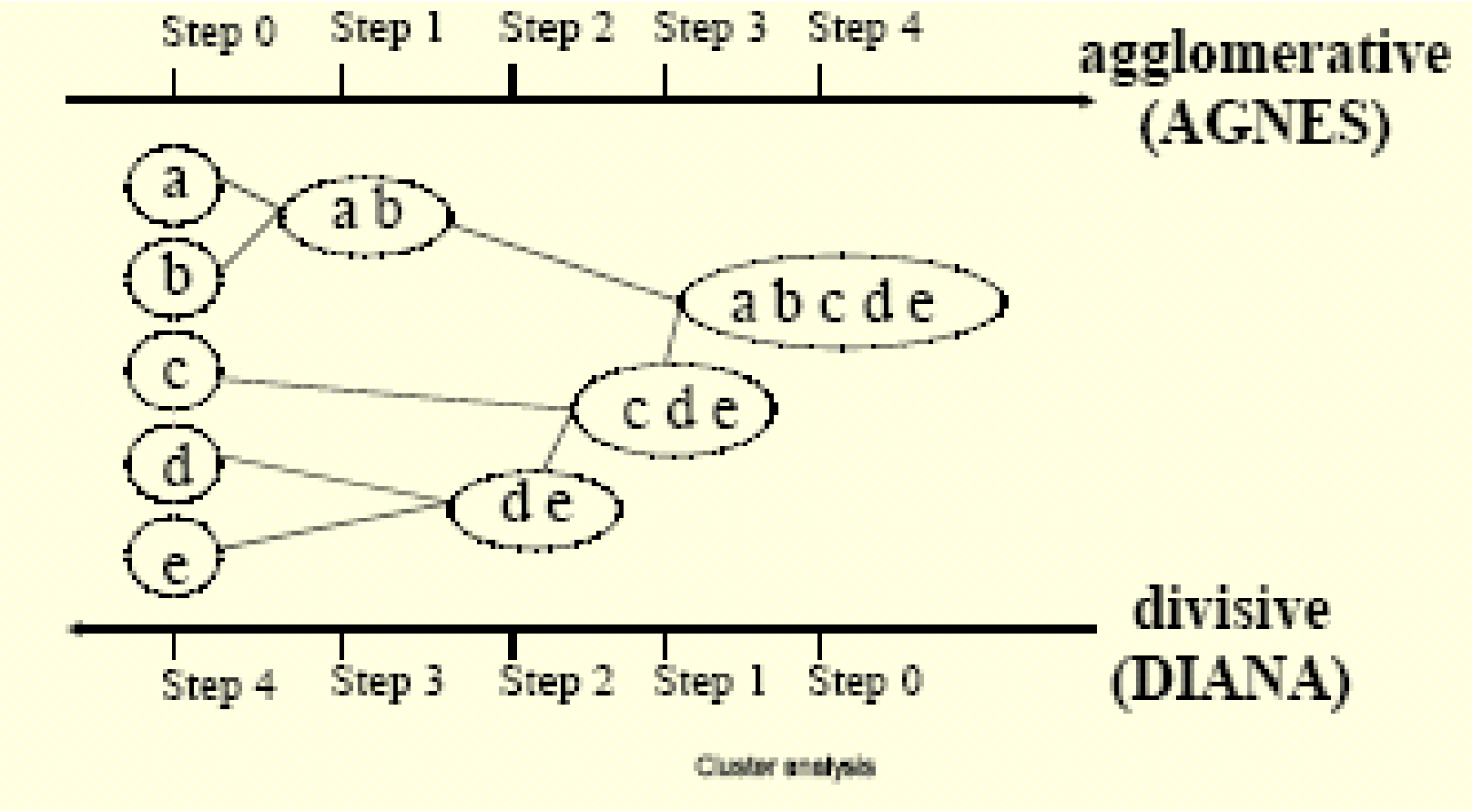
$$A = \frac{\sum_{j=1}^{n}(X_j - Z_i)^T(X_j - Z_i)}{n-1}$$



Hierarchical Clustering

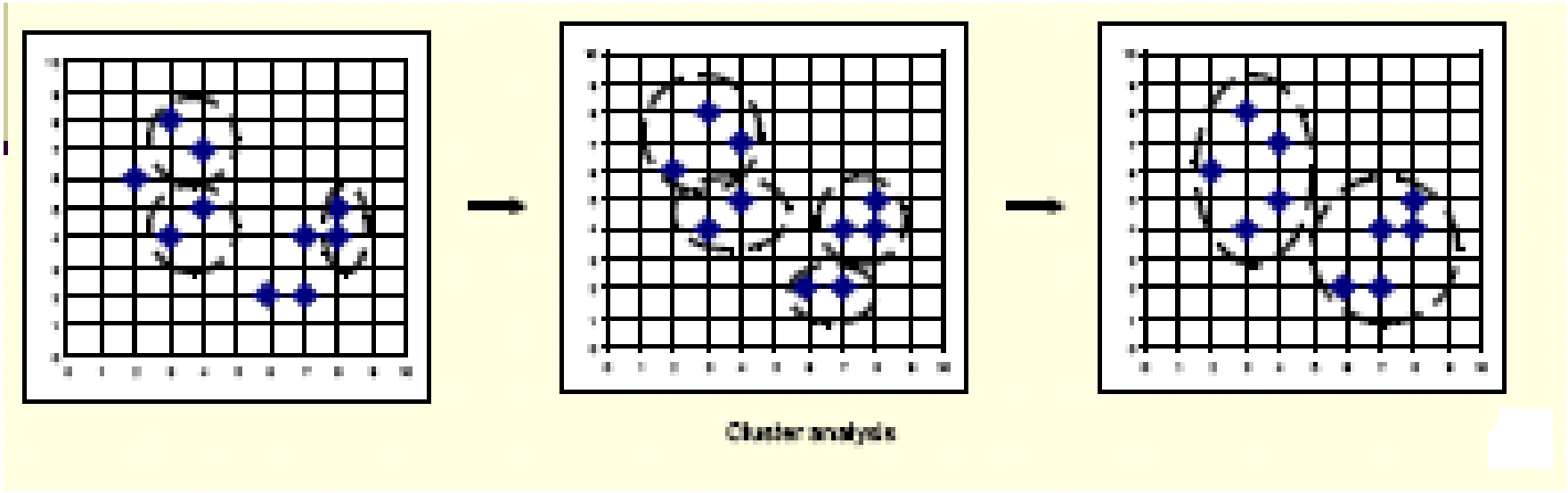
- Clustering เป็นการเกาะกลุ่มโดยระดับชั้น ซึ่งระเบียนข้อมูลถูกรวมกลุ่มโดยการใช้ระดับชั้น (Hierarchical decomposition)
- มีการใช้ distance matrix เป็นเงื่อนไขในการเกาะกลุ่ม

Hierarchical Clustering



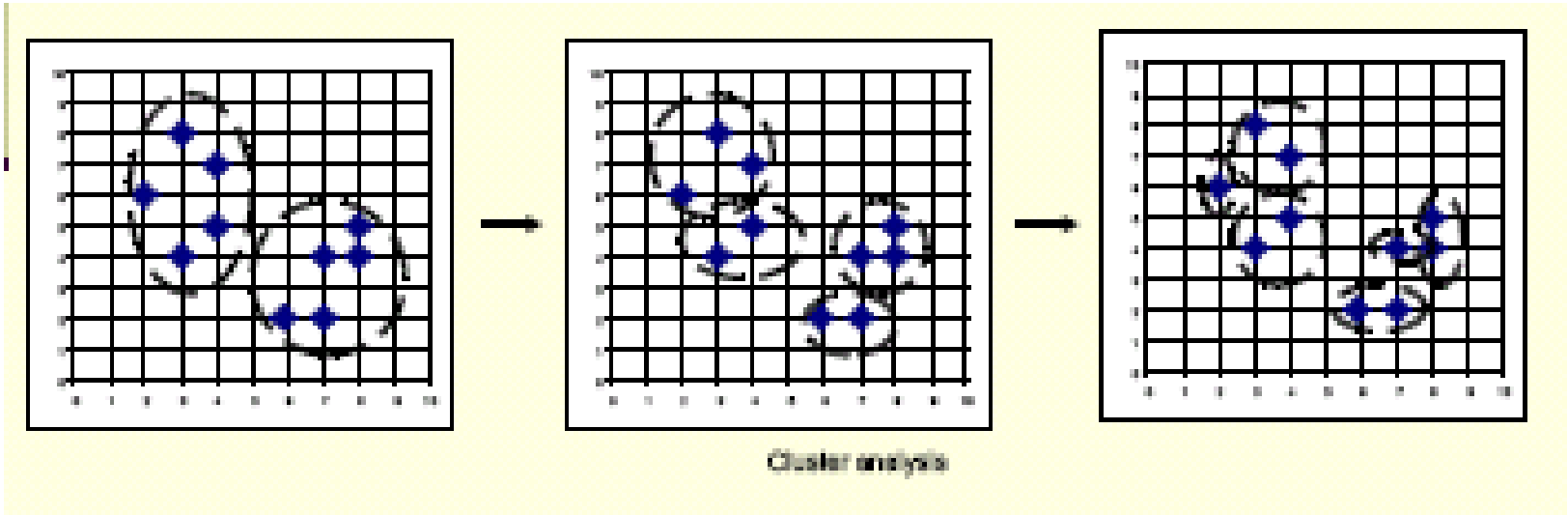
Hierarchical Clustering

AGNES(Agglomerative) นำเสนอโดย Kaufmann และ Rousseeuw(1990)
ใช้วิธีการ Single-link และเมตริกซ์แสดงความต่างของข้อมูล มีการรวมจุดที่มีความต่างน้อยที่สุดเข้าด้วยกัน
มีการทำซ้ำโดยทำการรวมซึ่งไม่มีการแยกออกจนกระทั่งสอดคล้องเงื่อนไขการหยุด
ในกรณีที่เราทำไปไม่หยุดจะได้ว่าทุกข้อมูลจะรวมอยู่กลุ่มเดียวกัน



Hierarchical Clustering

DIANA(Divisive Analysis) นำเสนอโดย Kaufmann และ Rousseeuw(1990) ใช้วิธี Single-Link และเมตริกซ์แสดงความต่างของข้อมูล เริ่มจากจัดข้อมูลทุกตัวให้อยู่ในกลุ่มเดียวกัน ทำซ้ำโดยแบ่งกลุ่มใหญ่ออกเป็นกลุ่มย่อย หยุดถ้าเงื่อนไขสอดคล้องการหยุด ถ้าไม่มีการตรวจสอบการหยุด ข้อมูลแต่ละตัวจะถูกจัดในกลุ่มที่ต่างกันหมด



ขั้นตอนพื้นฐานของ Hierarchical algorithm

กำหนดภายใน 1 เซตประกอบด้วย N คลัสเตอร์ และเมตริกซ์ของความแตกต่างมีขนาด N*N

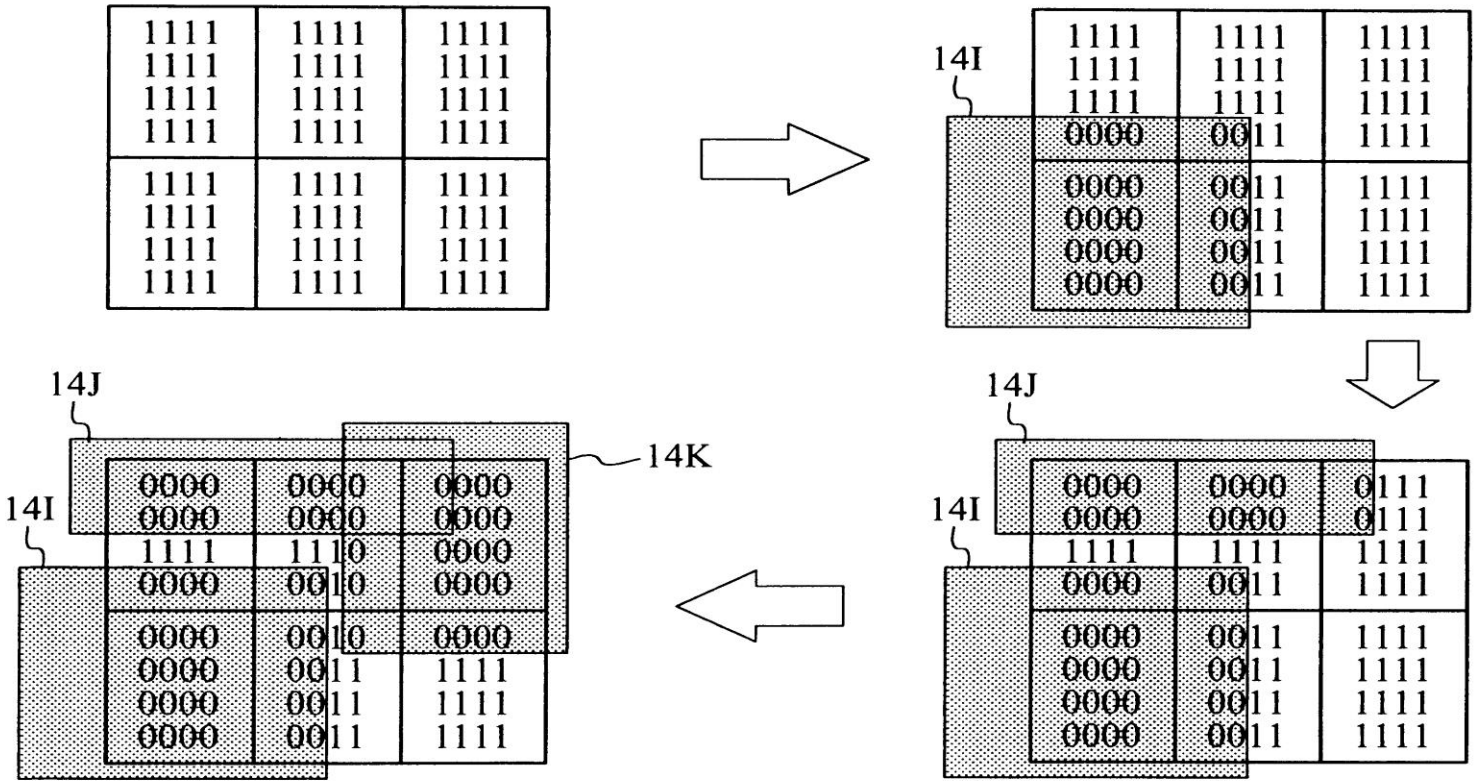
ขั้นตอนที่ 1 เริ่มต้นโดยการกำหนดแต่ละ item ให้ cluster ดังนั้นถ้าเรามี N ไอเท็ม ต้องมี cluster จำนวน N คลัสเตอร์เหมือนกัน เพื่อที่จะบรรจุ item ได้พอดี อนุญาตให้ความแตกต่างระหว่างคลัสเตอร์เหมือนกับความแตกต่างระหว่างคลัสเตอร์ที่บรรจุ item

ขั้นตอนที่ 2 เมื่อพบ คลัสเตอร์ 2 คลัสเตอร์ที่มีความใกล้เคียงกัน ให้ทำการรวมทั้งสองคลัสเตอร์เข้าด้วยกัน ดังนั้นเราจะได้คลัสเตอร์เพียงคลัสเตอร์เดียว

ขั้นตอนที่ 3 คำนวณความแตกต่างระหว่างคลัสเตอร์ใหม่และคลัสเตอร์เก่าของแต่ละคลัสเตอร์

ขั้นตอนที่ 4 ทำซ้ำขั้นตอนที่ 2 และ 3 จนกว่าคลัสเตอร์ทั้งหมด จะกลายเป็นเพียงคลัสเตอร์เดียว

integrated circuit using hierarchical algorithm



Single-Linkage Clustering

- นำเมตริกที่มีความใกล้เคียงกันมาแยกแถวและคอลัมภ์ออกจะเห็นเป็นคลัสเตอร์เดิมก่อนที่จะนำมารวมกันเป็นคลัสเตอร์ใหม่
- เมตริกซ์ N*N ที่มีลักษณะที่ใกล้เคียงกันคือ D =[d(i,j)]
- ซึ่งภายในคลัสเตอร์นี้จะกำหนดลำดับของตัวเลขเป็น 0,1,2,…,(n-1) และ L(k) เป็น level ของ k clustering m จะแสดงถึงตัวเลขที่ต่อเนื่องกันและความใกล้เคียงกันระหว่าง cluster (r) และ (s)

Single-Linkage Clustering

ขั้นตอนที่ 1 เริ่มต้นด้วยการแยกคลัสเตอร์ที่มี level L(0) =0 และเลขที่อยู่ในลำดับที่ m=0

ขั้นตอนที่ 2 หาคู่ลำดับของคลัสเตอร์ที่มีความแตกต่างกันน้อยที่สุด โดยที่คู่ลำดับจะขึ้นอยู่กับ

$$d[(r)(s)] = \min d[(i)(j)]$$

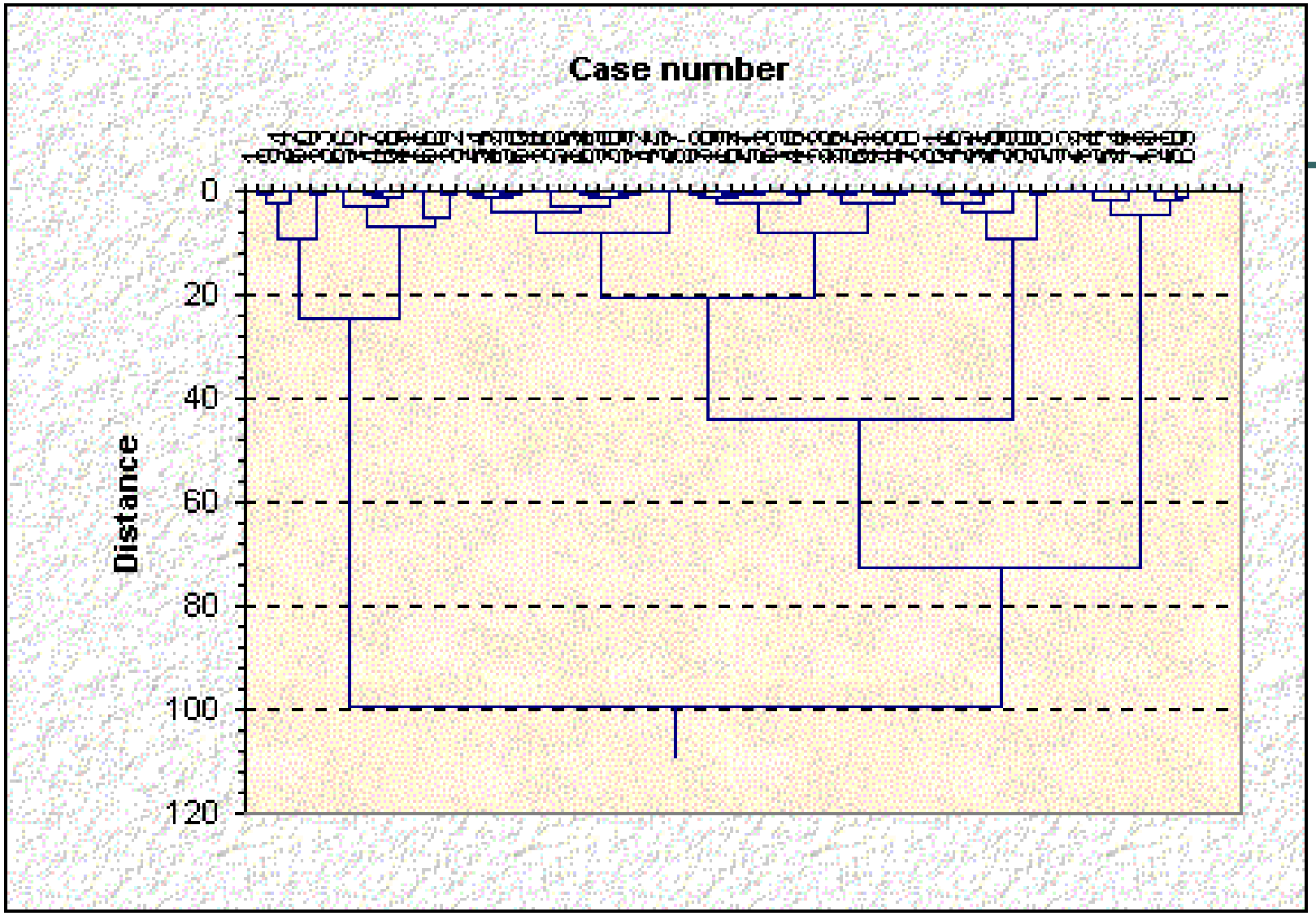
โดยที่จะต้องเป็นค่าที่น้อยที่สุดของคู่ลำดับทั้งหมดของทุกๆคลัสเตอร์ที่อยู่ในนั้น

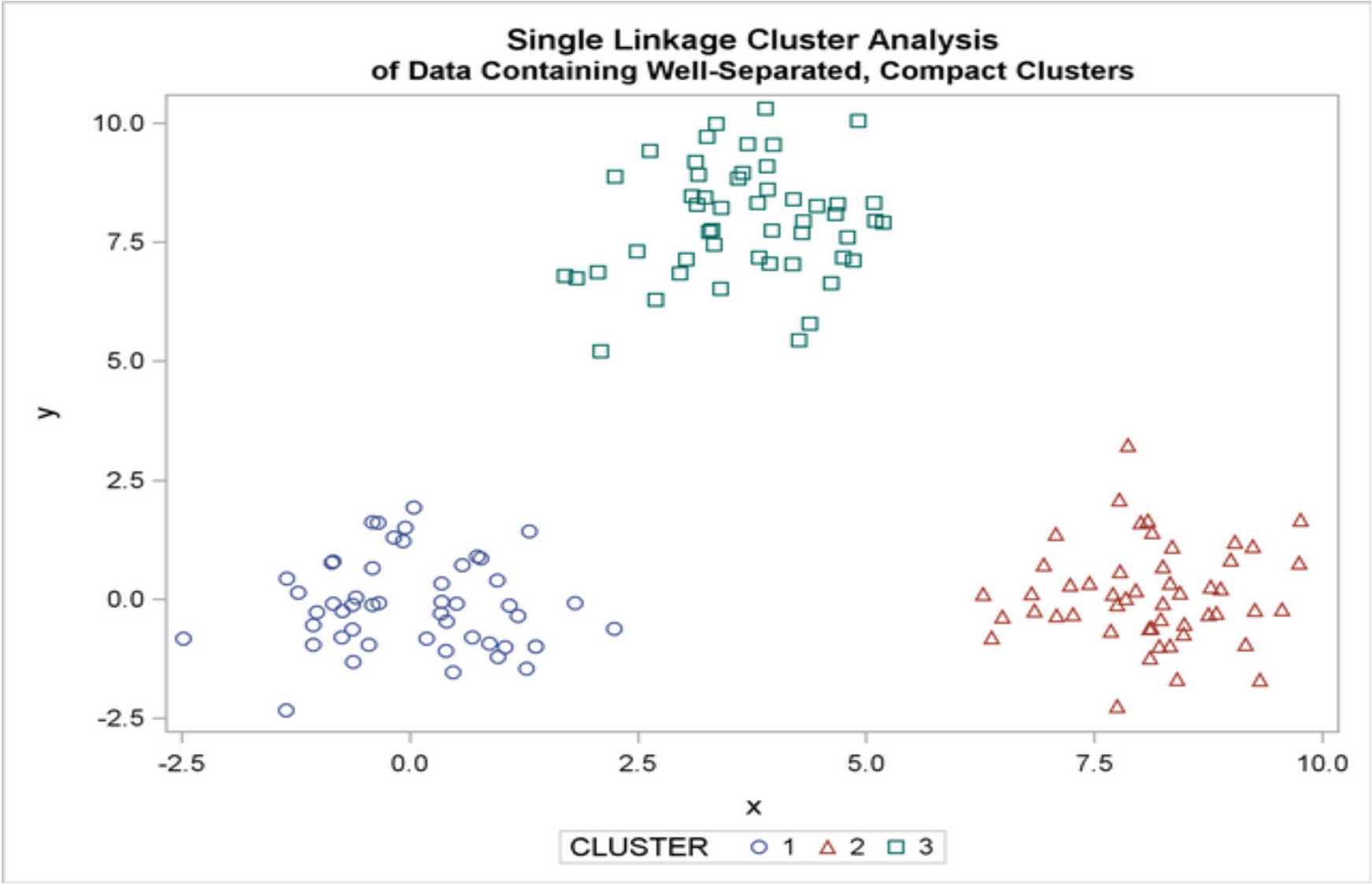
ขั้นตอนที่ 3 เพิ่มเลขลำดับจาก m = m+1 และทำการรวมคลัสเตอร์ (r) และ (s) ให้เป็นคลัสเตอร์เดียวกัน เป็นคลัสเตอร์ m ใหม่ กำหนด level ของคลัสเตอร์เป็น L(m) = d[(r)(s)]

ขั้นตอนที่ 4 แก้ไข proximity เมตริกซ์ D โดยตัดแถวและคอลัมภ์ที่ตรงกันของคลัสเตอร์ (r) และ (s) ออกและทำการเพิ่มแถวและคอลัมภ์ที่ตรงกันนี้ไปยังคลัสเตอร์ใหม่ ความใกล้เคียงกันระหว่างคลัสเตอร์ใหม่ denoted(r,s) และคลัสเตอร์ (k) เก่าสามารถนิยามได้ใหม่ดังนี้

$$d[(k)(r,s)] = \min d[(k)(r), d[(k)(s)]$$

ขั้นตอนที่ 5 ถ้ามีเพียงคลัสเตอร์เดียว ให้จบการคำนวณแต่ถ้าไม่ ก็ให้ทำซ้ำตั้งแต่ขั้นตอนที่ 2-5


Case number
Distance
0
20
40
60
80
100
120


Single Linkage Cluster Analysis
of Data Containing Well-Separated, Compact Clusters
10.0
7.5
5.0
2.5
0.0
-2.5
y
-2.5
0.0
2.5
5.0
7.5
10.0
x
CLUSTER
1
2
3

ตัวอย่าง Single-Linkage Clustering

การใช้ Hierarchical clustering ในการหาระยะทางระหว่างเมืองต่างๆ โดยระยะทางมีหน่วยเป็นกิโลเมตร ภายในประเทศอิตาลี

Input distance matrix

โดยทุกๆคลัสเตอร์มีค่า L=0

	BA	FI	MI	NA	RM	TO
BA	0	662	877	255	412	996
FI	662	0	295	468	268	400
MI	877	295	0	754	564	138
NA	255	468	754	0	219	869
RM	412	268	564	219	0	669
TO	996	400	138	869	669	0

หาเมือง 2 เมืองที่มีระยะใกล้ที่สุด

	BA	FI	MI	NA	RM	TO
BA	0	662	877	255	412	996
FI	662	0	295	468	268	400
MI	877	295	0	754	564	138
NA	255	468	754	0	219	869
RM	412	268	564	219	0	669
TO	996	400	138	869	669	0

เมืองสองเมืองที่มีระยะทางใกล้กันที่สุดคือเมือง MI และ TO ซึ่งมีระยะทางห่างกัน 138 กิโลเมตร
ทำการรวมเมืองทั้งสองเข้าด้วยกันเป็นคลัสเตอร์เดียวกันคือ MI/TO
เพราะฉะนั้น Level ของคลัสเตอร์ใหม่น้มีค่าเท่ากับ L(MI/TO) = 138 และค่า m= 1

คำนวณหาระยะทางของเมืองอื่นๆ โดยรอบกฎอยู่ว่า ถ้าบริเวณโดยรอบออปเจ็กซ์นั้นมีค่าเท่ากันหรือน้อยกว่าแต่ต้องน้อยที่สุดของระยะทางของสมาชิกอื่นๆในคลัสเตอร์ดังนั้น จึงเลือกระยะทางจากเมือง **MI/TO** ถึง **RM** มีระยะทาง **564** กิโลเมตร ซึ่งเป็นระยะทางจากเมือง **MI** ถึง **RM**

	BA	FI	MI/TO	NA	RM
BA	0	662	877	255	412
FI	662	0	295	468	268
MI/TO	877	295	0	754	564
NA	255	468	754	0	219
RM	412	268	564	219	0

ซึ่ง **min d(i,j) = d(NA,RM) = 219**

จะนำ **NA** และ **RM** มารวมเข้าด้วยกันเป็นคลัสเตอร์ใหม่ เรียกว่า **NA/RM** โดยมีค่า **L(NA/RM) = 219** และ **m =2**

	BA	FI	MI/TO	NA/RM
BA	0	662	877	255
FI	662	0	295	268
MI/TO	877	295	0	564
NA/RM	255	268	564	0

ซึ่ง **min d(i,j) = d(BA,NA/RM) = 255** จะนำ **BA** และ **NA/RM** มารวมกันจะได้เป็นคลัสเตอร์ใหม่ เรียกว่า **BA/NA/RM** โดยมีค่า **L(BA/NA/RM)=**255และ**m= 3**

	BA/NA/RM	FI	MI/TO
BA/NA/RM	0	268	564
FI	268	0	295
MI/TO	564	295	0

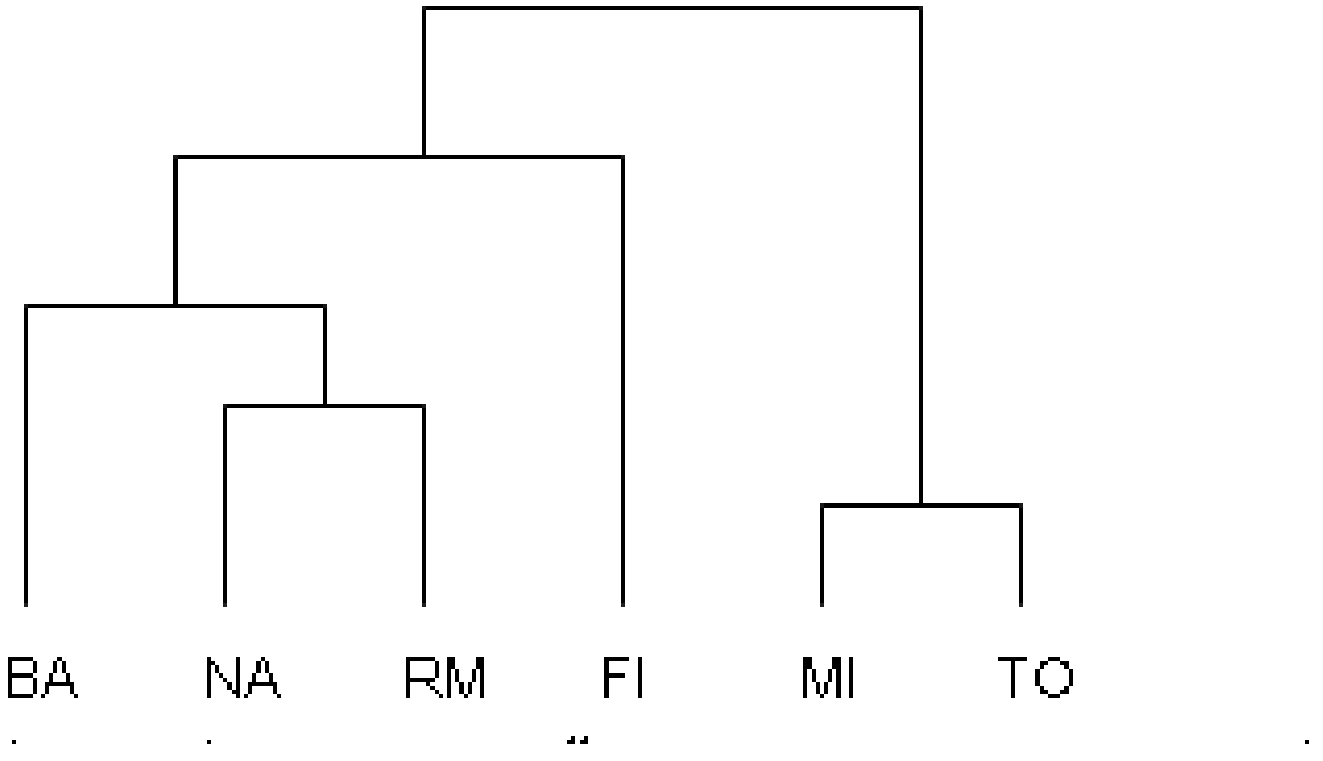
ซึ่ง **min d(i,j) = d(BA/NA/RM.FI) = 268**

จะนำ **BA/NA/RM** และ **FI** มารวมกันจะได้เป็นคลัสเตอร์ใหม่ เรียกว่า **BA/NA/RM/FI**

โดยมีค่า **L(BA/NA/RM/FI) =** 268 และ **m = 4**

	BA/NA/RM/FI	MI/TO
BA/NA/RM/FI	0	295
MI/TO	295	0

สุดท้ายเราจะทำการรวม 2 คลัสเตอร์สุดท้ายได้ **level = 295**
สามารถเขียนเป็น **hierarchical tree** ได้ดังนี้



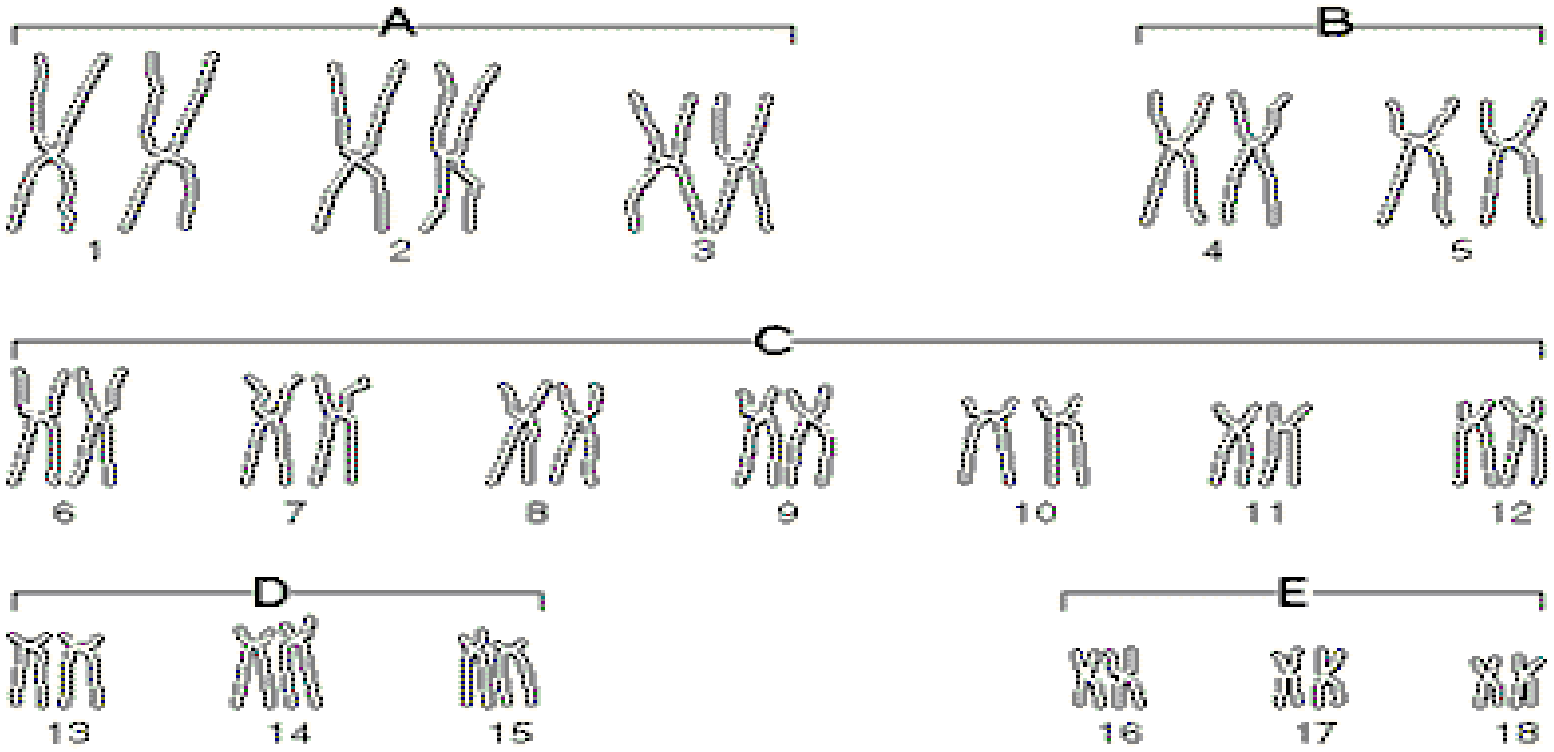
Genetic Algorithm

- ถูกคิดค้นขึ้นโดย John Holland ในปีค.ศ. 1975 เป็นการนำขบวนการวิวัฒนาการของสิ่งมีชีวิตมาประยุกต์ใช้ในงานปัญญาประดิษฐ์ เพื่อใช้สำหรับหาคำตอบที่ดีที่สุด (Optimization) ของปัญหาต่างๆจากจำนวนคำตอบที่เป็นไปได้ทั้งหมดของการแก้ปัญหานั้น

Genetic Algorithm

- รูปแบบโครโมโซมที่ใช้ในการนำเสนอทางเลือกที่สามารถจะเป็นได้ของแต่ละปัญหา
- วิธีสร้างประชากรต้นกำเหนิดของทางเลือกที่สามารถจะเป็นไปได้
- ฟังก์ชันสำหรับประเมินค่าความเหมาะสมเพื่อให้คะแนนแต่ละทางเลือก
- จีเนติกโอเปอเรเตอร์ซึ่งใช้ในการปรับเปลี่ยนองค์ประกอบของข้อมูลตลอดกระบวนการ ได้แก่ การคัดเลือก การครอสโอเวอร์ และการมิวเตชั่น
- ค่าพารามิเตอร์ต่างๆที่ต้องใช้สำหรับจีเนติกอัลกอริทึม เช่น ขนาดของประชากร ,ความน่าจะเป็นของการใช้จีเนติกโอเปอเรอเตอร์ และจำนวนรุ่น เป็นต้น

Biological Chromosomes were the incentive for Genetic Algorithms



ตัวอย่าง การค้นคืนสารสนเทศโดยใช้จีเนติกอัลกอริทึม

สมมุติมีเอกสาร 5 ฉบับประกอบด้วยคำสำคัญดังนี้

DOC1 ={Database, Query, Data Retrieval ,
Computer,Network, DBMS}
DOC2={Artificial Intelligence, Internet, Indexing, Natural
Language Processing}
DOC3={Database , Expert System, Information Retrieval
System, Multimedia}
DOC4={Fuzzy Logic, Neural Network, Computer Networks}
DOC5-{Object-Oriented, DBMS , Query ,Indexing}

นำคำสำคัญทั้งหมดมาจัดเรียงลำดับจากน้อยไปมากรวม 16 คำดังนี้

Artificial Intelligence
Computer Network
Data Retrival
Database
DBMS
Expert System ,
Fuzzy Logic
Indexing
Information Retrieval System
Internet
Multimedia
Natural Language Processing
Neural Network
Object Oriented
Query
Relational Database

DOC1 ={Database, Query, Data Retrieval , Computer,Network, DBMS}

DOC2={Artificial Intelligence, Internet, Indexing, Natural Language Processing}

DOC3={Database , Expert System, Information Retrieval System, Multimedia}

DOC4={Fuzzy Logic, Neural Network, Computer Networks}

DOC5={Object-Oriented, DBMS , Query ,Indexing}

นำเสนอรูปแบบโครโมโซมดังนี้

DOC1=0110100000000011

DOC2=1000000101010000

DOC3=0001010010100000

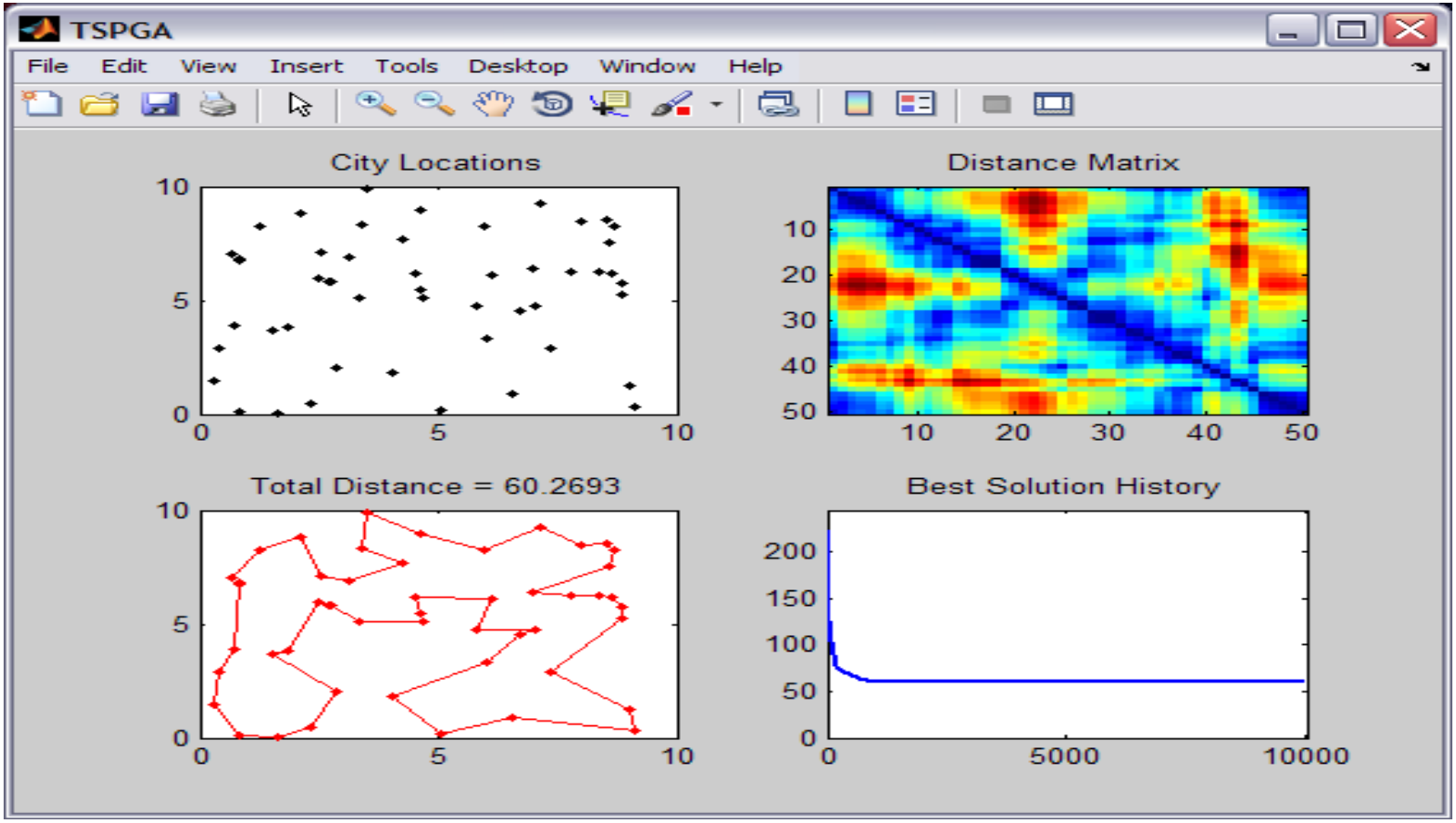
DOC4=0100001000001000

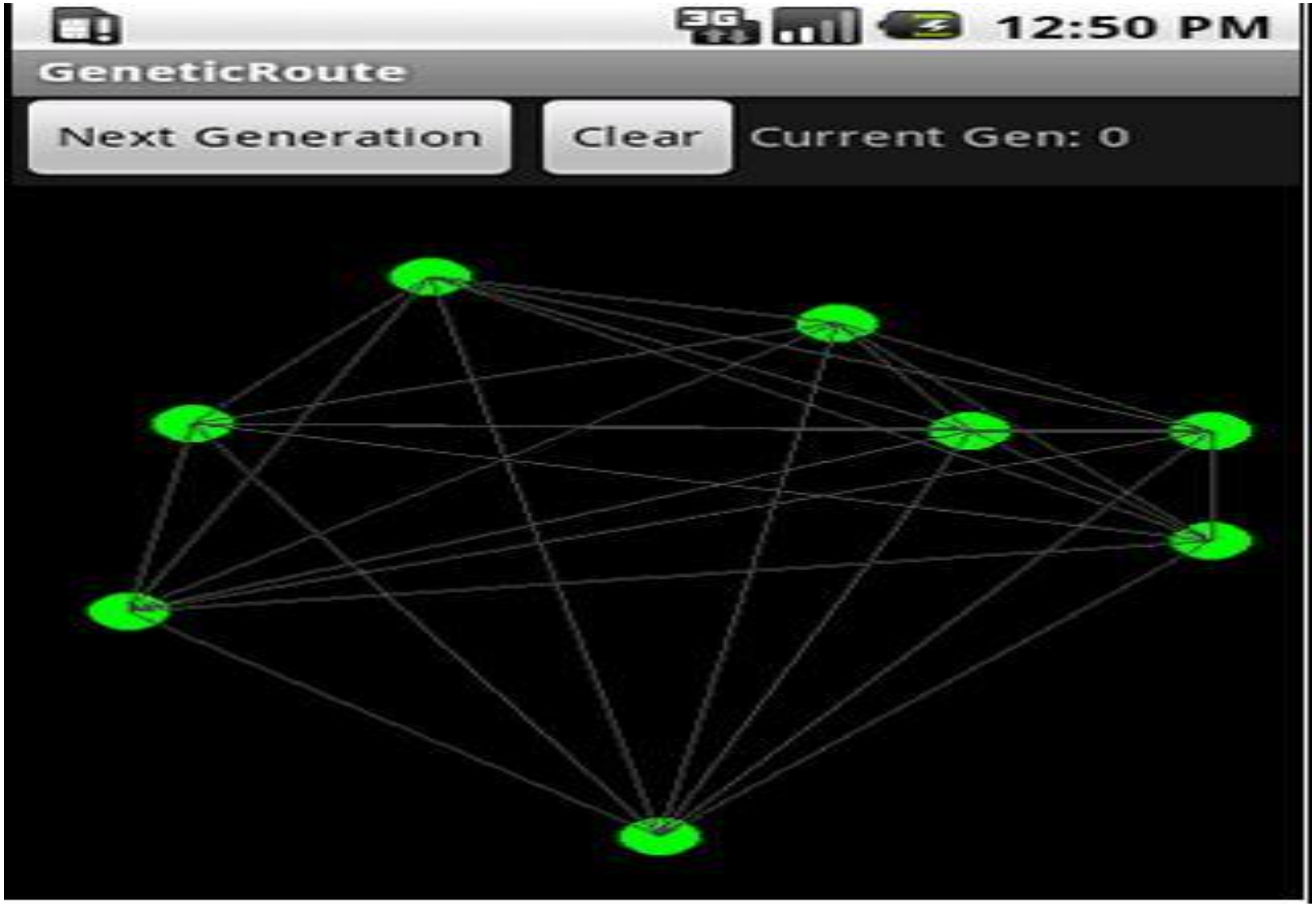
DOC5=0000100100000110

Genetic Algorithm

- โครโมโซมชุดแรกที่ได้มานี้จะเรียกว่าประชากรต้นกำเนิด ซึ่งจะนำไปผ่านกระบวนการจีแนติกต่อไป ความยาวของโครโมโซมเหล่านี้จะขึ้นอยู่กับจำนวนคำสำคัญของชุดเอกสารทั้งหมดที่ตรงตามข้อเรียกร้อง(query) จากตัวอย่างนี้มีความยาวเท่ากับ 16 บิต
- จีเนติกอัลกอริทึม จะทำเป็นวัฏจักรหมุนเวียนอยู่จนกระทั่งถึงจุดหนึ่งที่ตรงตามเงื่อนไขตามที่กำหนด หรือสิ้นสุดเมื่อพบคำตอบที่ดีที่สุดแล้ว หรือถึง threshold ตามที่ได้กำหนดไว้ล่วงหน้า

Traveling Salesman Problem - Genetic Algorithm Finds a near-optimal solution




12:50 PM
GeneticRoute
Next Generation
Clear
Current Gen: 0

การจำแนกหมวดหมู่เอกสารภาษาไทยอัตโนมัติโดยใช้อัลกอริทึม FPTC

- ภาควิชาวิทยาการคอมพิวเตอร์และสารสนเทศ
- บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ

วัตถุประสงค์

- เพื่อพัฒนาเครื่องมือในการจำแนกหมวดหมู่เอกสารข้อความภาษาไทยด้วยอัลกอริทึม Feature Projection Text Categorization (FPTC)
- ซึ่งเป็นอัลกอริทึมที่ปรับมาจาก k-Nearest Neighbor ลักษณะเด่นของ FPTC คือ การแทนคุณลักษณะในแบบภาพฉายของแต่ละคุณลักษณะ
- การจำแนกหมวดหมู่จะใช้วิธีการเปรียบเทียบความคล้ายของคำที่ปรากฏในเอกสารที่ใช้ทดสอบกับคำที่ปรากฏในเอกสารที่ใช้ในกระบวนการเรียนรู้ เพื่อหาเอกสารที่คล้ายกับเอกสารทดสอบมากที่สุด และกำหนดหมวดหมู่ของเอกสารนั้นให้กับเอกสารทดสอบ
- ใช้เอกสารข่าวภาษาไทยจากหนังสือพิมพ์ออนไลน์เป็นกรณีศึกษา
- จากผลการทดสอบพบว่า การจำแนกหมวดหมู่ด้วยอัลกอริทึม FPTC สามารถจำแนกหมวดหมู่เอกสารภาษาไทยได้อย่างมีประสิทธิผลดี สำหรับข้อมูลที่มีการกระจายตัวของหมวดหมู่เท่ากัน

อัลกอริทึม Feature Projection Text Categorization (FPTC)

- กระบวนการเรียนรู้เอกสารทั้งหมดพร้อมกันเพียงครั้งเดียว
- เอกสารทั้งหมดจะถูกเก็บในลักษณะของภาพฉาย (Projections) บนแต่ละ
- คุณลักษณะของเอกสาร เอกสารที่ไม่มีคุณลักษณะใดจะไม่ถูกเก็บบนคุณลักษณะนั้น ระยะห่าง
- ระหว่างเอกสารสองเอกสารก็จะถูกพิจารณาบนคุณลักษณะเดียว

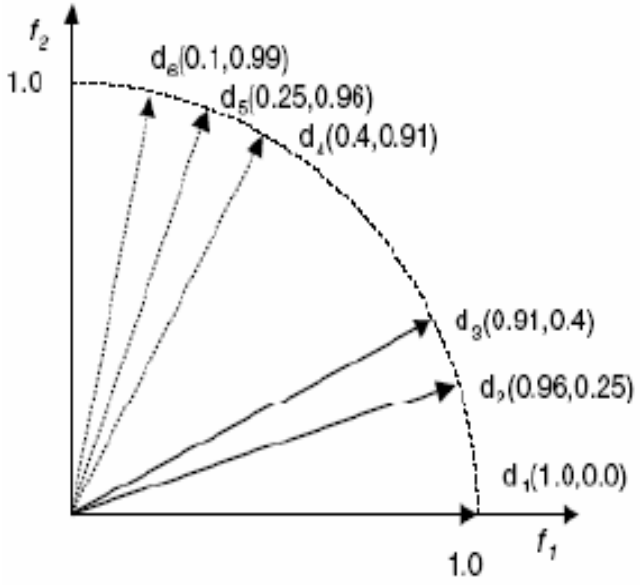
ระยะห่างระหว่างเอกสารคำนวณดังนี้

$$dist_m(t_m(i), t_m(j)) = \left| w(t_m, \vec{d}_i) - w(t_{(m}, \vec{d}_j) \right| \quad (3\text{-}1)$$

โดยที่ $t_m(i)$ คือ คุณลักษณะ t ลำดับที่ m ในเอกสาร d_i และ $w(t_m, \vec{d}_i)$ คือ ค่าน้ำหนักของคุณลักษณะ t ในเอกสาร d_i

การแทนเอกสารในรูปแบบเวกเตอร์

การกำหนดหมวดหมู่ให้กับเอกสารจะเป็นไปตามคะแนนเสียง (Vote) ของเอกสารที่มีค่าน้ำหนักใกล้เคียงกันมากที่สุดจำนวน k ลำดับ บนแต่ละคุณลักษณะ ถ้ามีจำนวนคุณลักษณะ f ชุด วิธีการนี้จะให้ค่าโหวตจำนวน $f \times k$ ชุด ในขณะที่ k-Nearest Neighbor จะให้ค่า k ชุด



การแทนคุณลักษณะในรูปแบบภาพฉายของคุณลักษณะ

w: weight, d: document, c: category

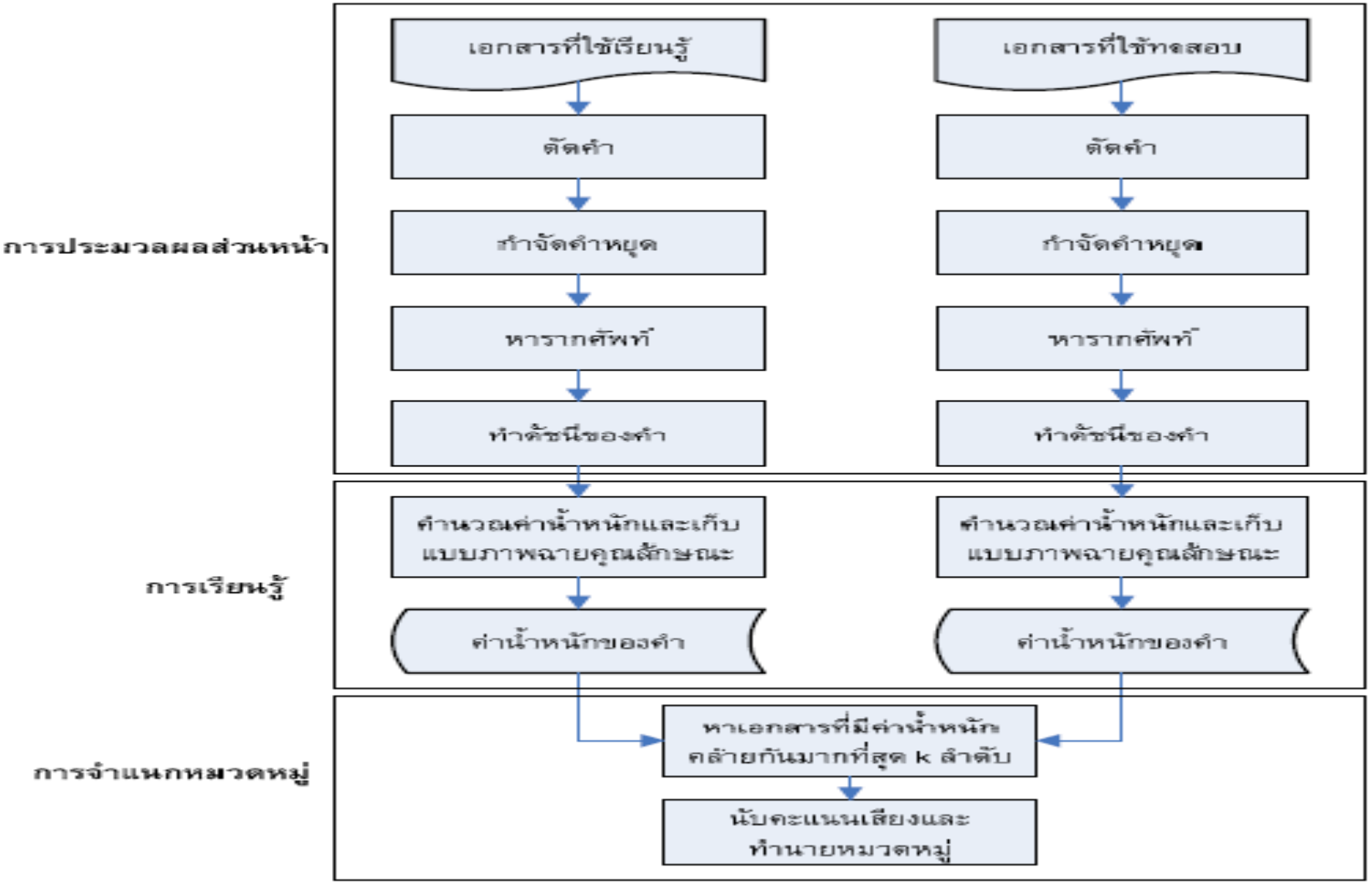
w, d, c
1.0, d_1, c_1
0.96, d_2, c_1
0.91, d_3, c_1
0.4, d_4, c_2
0.25, d_5, c_2
0.1, d_6, c_2

feature projections on feature f_1

w, d, c
0.99, d_6, c_2
0.96, d_5, c_2
0.91, d_4, c_2
0.4, d_3, c_1
0.25, d_2, c_1

feature projections on feature f_2

ขั้นตอนของการจำแนกหมวดหมู่



ปัญหาอุปสรรคในการทำวิจัย

- การตัดคำภาษาไทยไม่ถูกต้องมีผลอย่างมากต่อความถูกต้องในการจำแนกเอกสาร
- การเขียนข่าวภาษาไทยมีการใช้ศัพท์แสลงหรือสำนวนเป็นจำนวนมาก คำที่ใช้ในหมวดหมู่หนึ่งอาจมีความหมายที่แตกต่างไปสำหรับอีกหมวดหมู่หนึ่ง ซึ่งมีผลทำให้จำแนกเอกสารได้ไม่ถูกต้อง
- เช่น ข่าวในหมวดหมู่อาชญากรรมพบคำว่า บุก ยิง ทะลุ จากประโยคตัวอย่างที่ว่า “ผู้ร้ายบุกยิงเหยื่ออย่างอุกอาจ กระสุนทะลุท้ายทอยหนึ่งนัด แล้วยังตามแทงซ้ำนับสิบแผล”
- ข่าวในหมวดกีฬา ก็พบคำเดียวกันในความหมายของคำแสลงในประโยคที่ว่า “ผีแดงบุกทะลุทะลวงกองหน้าสาริกาดง ยิงกระหน่ำ 3 ต่อ 0” เป็นต้น