

Lecture 8: Document Similarity & Clustering

Pilsung Kang

School of Industrial Management Engineering
Korea University

AGENDA

01 Document Similarity

02 Clustering Overview

03 Partition-based Clustering

04 Hierarchical Clustering

05 Density-based Clustering

06 Self-Organizing Map

Document Similarity: Backgrounds

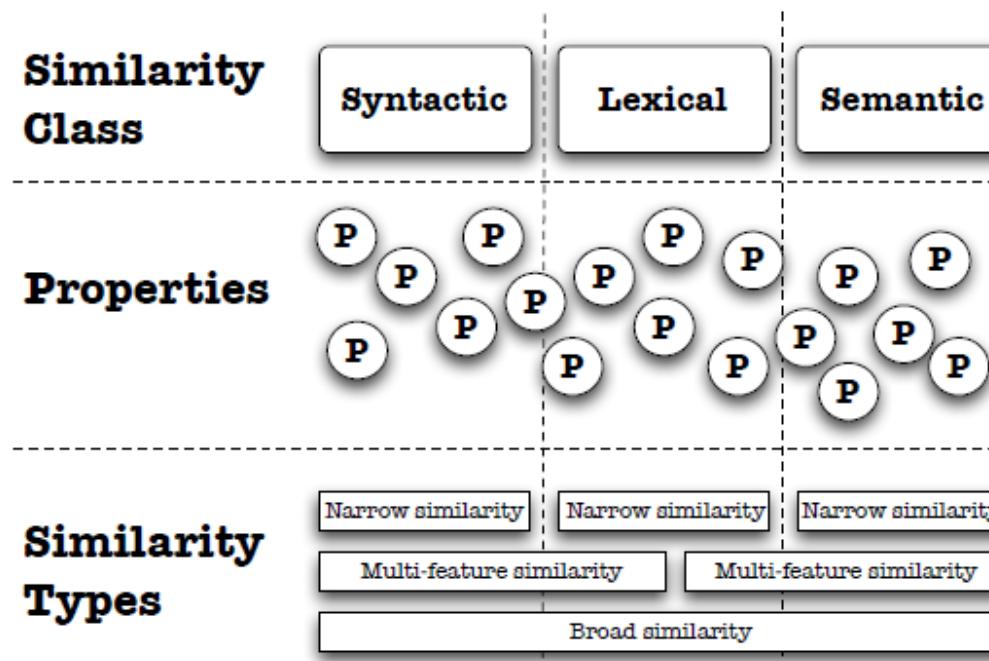
- Similarity in the Cognitive Sciences
 - ✓ Two objects falling under different concepts are similar because they have many **properties** in common
 - Similarity between two objects increases as a function of the number of properties they share
 - Properties can be treated as independent and additive
 - **The properties determining similarity are all roughly the same level of abstractness**
 - Similarities should be sufficient to describe conceptual structure

Document Similarity

- Problem
 - ✓ Given two text documents, how similar are they?
 - ✓ Example
 - d1: ant, ant, bee
 - d2: dog, bee, dog, hog, dog, ant, dog
 - d3: cat, gnu, dog, eel, fox
- Similarity between text documents
 - ✓ Used for search engines, text corpus visualizations, filtering, sorting, and retrieving rely fundamentally on similarity measures
 - ✓ A number of candidate models exist, not been assessed in terms of their ability to emulate human judgments of similarity

Document Similarity

- Similarity Measure
 - ✓ A function which computes the **degree of similarity** between a pair of vectors (documents)
 - ✓ There are a large number of similarity measures proposed in the literature, because the **best** similarity measure **does not exist** (yet!)



Document Similarity

- Term similarity
 - ✓ Two documents are similar if they contain some of the same terms
 - ✓ Possible measures of similarity might take into consideration
 - The length of the document
 - The number of terms in common
 - Whether the terms are common or unusual
 - How many times each term appears
- Vector space model
 - ✓ A document is represented in n-dimensional space, where n is the number of different terms/features used to index a set of documents

Document Similarity I: Binary Similarity Models

- Notation

- ✓ x_{ik} : the frequency of the term k in the document i
- ✓ t_{ik} : 1 if $x_{ik} > 0$, 0 otherwise
- ✓ $a_{ij} : \sum_k t_{ik} t_{jk}$, the number of words/N-grams in the corpus that are common to both documents
- ✓ $b_{ij} : \sum_k t_{ik} (1 - t_{jk})$, the number of words/N-grams that are presented in the document i but not in the document j
- ✓ $c_{ij} : \sum_k (1 - t_{ik}) t_{jk}$, the number of words/N-grams that are presented in the document j but not in the document i
- ✓ $d_{ij} : \sum_k (1 - t_{ik})(1 - t_{jk})$, the number of words/N-grams that are presented in neither documents

Document Similarity I: Binary Similarity Models

- Example

Freq.	D ₁	D ₂	D ₃
T ₁	3	0	2
T ₂	0	0	1
T ₃	5	3	0
T ₄	0	2	1
T ₅	0	1	2

Binary	D ₁	D ₂	D ₃
T ₁	1	0	1
T ₂	0	0	1
T ₃	1	1	0
T ₄	0	1	1
T ₅	0	1	1

✓ Between D₁ and D₂

- $a_{12} = 1, b_{12} = 1, c_{12} = 2, d_{12} = 1$

Document Similarity I: Binary Similarity Models

- Common Features Model

- ✓ Similarity is measured by the proportion of common features

$$s_{ij}^{common} = \frac{a_{ij}}{a_{ij} + b_{ij} + c_{ij} + d_{ij}}$$

- Ratio Model (Jaccard coefficient in binary format)

- ✓ The ratio of common to common and distinctive features

$$s_{ij}^{ratio} = \frac{a_{ij}}{a_{ij} + b_{ij} + c_{ij}}$$

Common	D ₁	D ₂	D ₃
D ₁	-	1/5	1/5
D ₂		-	2/5
D ₃			-

Ratio	D ₁	D ₂	D ₃
D ₁	-	1/4	1/5
D ₂		-	2/5
D ₃			-

Document Similarity I: Binary Similarity Models

- Simple Matching Coefficient

- ✓ Two documents become more dissimilar to the extent that one document has a feature that the other does not

$$S_{ij}^{smc} = \frac{a_{ij} + d_{ij}}{a_{ij} + b_{ij} + c_{ij} + d_{ij}}$$

SMC	D ₁	D ₂	D ₃
D ₁	-	2/5	1/5
D ₂		-	2/5
D ₃			-

- For more binary similarity measures, see the Appendix A

Document Similarity 2: Count Similarity Models

- Jaccard similarity

- ✓ The ratio of common to common and distinctive features

$$s_{ij}^{Jaccard} = \frac{\sum_k \min(x_{ik}, x_{jk})}{\sum_k \max(x_{ik}, x_{jk})}$$

- Overlap similarity

- ✓ Measures the overlap between two document defined as the size of the intersection divided by the smaller of the size of the two documents

$$s_{ij}^{overlap} = \frac{\sum_k \min(x_{ik}, x_{jk})}{\min(\sum_k x_{ik}, \sum_k x_{jk})}$$

Jaccard	D1	D2	D3
D1	-	3/11	2/12
D2		-	2/9
D3			-

Overlap	D1	D2	D3
D1	-	3/6	2/6
D2		-	2/6
D3			-

Document Similarity 2: Count Similarity Models

- Cosine similarity

- ✓ The similarity between two document is a function of the angle between their vectors in the vector space

$$S_{ij}^{cosine} = \frac{\sum_k (x_{ik} \times x_{jk})}{\sqrt{(\sum_k x_{ik}^2)(\sum_k x_{jk}^2)}}$$

Cosine	D ₁	D ₂	D ₃
D ₁	-	0.6875	0.3254
D ₂		-	0.3380
D ₃			-

AGENDA

01 Document Similarity

02 Clustering Overview

03 Partition-based Clustering

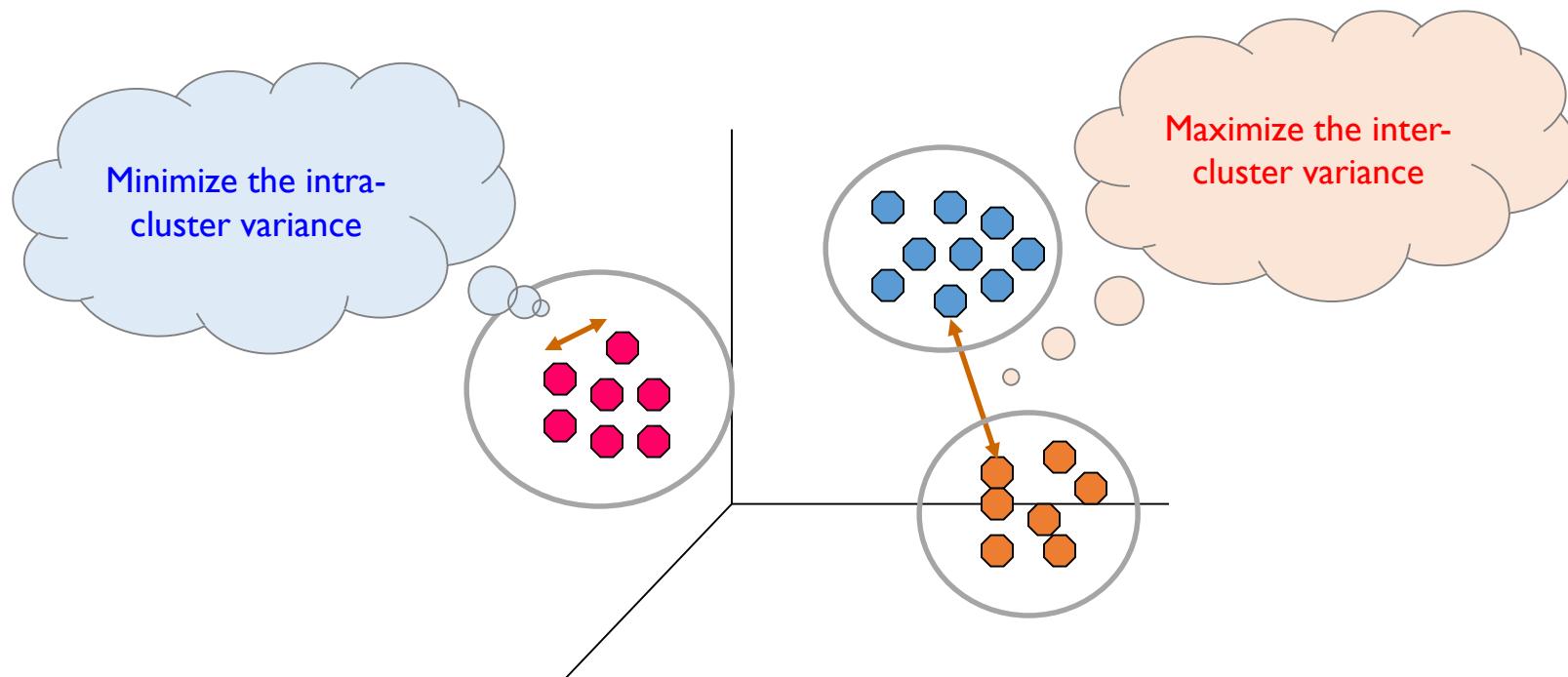
04 Hierarchical Clustering

05 Density-based Clustering

06 Self-Organizing Map

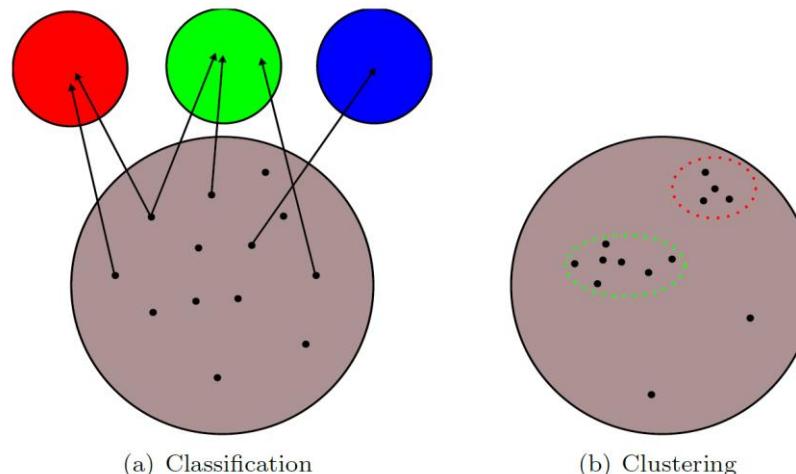
Type of Clustering

- What is clustering?
 - ✓ Find groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups



Document Clustering

- Document clustering
 - ✓ Aims to discover natural grouping, presenting an overview of the classes (topics) in a collection of documents
- Classification (Categorization) vs. Clustering
 - ✓ Classification: The number of classes (and their properties) is known a priori, and documents are assigned to one of them (supervised learning)
 - ✓ Clustering: neither the number, properties, or membership of classes known in advance (unsupervised learning)



Clustering Overview

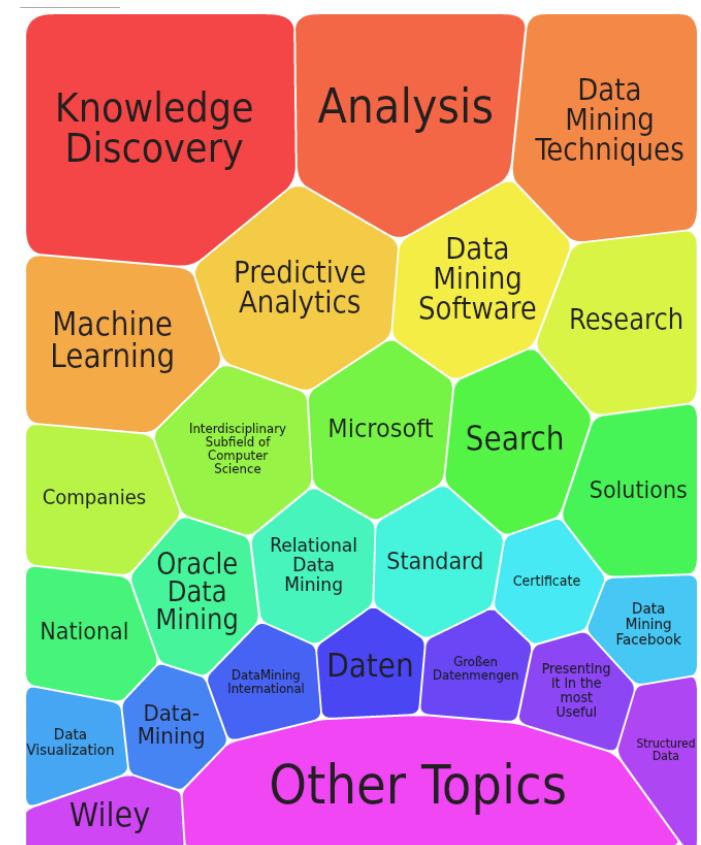
- Where are clustering used?

✓ “Understanding”

- Related documents for browsing
- Genes and proteins for similar functionalities
- Stocks with similar price fluctuation

Query: israel
 Documents: 272, Clusters: 15, Average Cluster Size: 15.1 documents

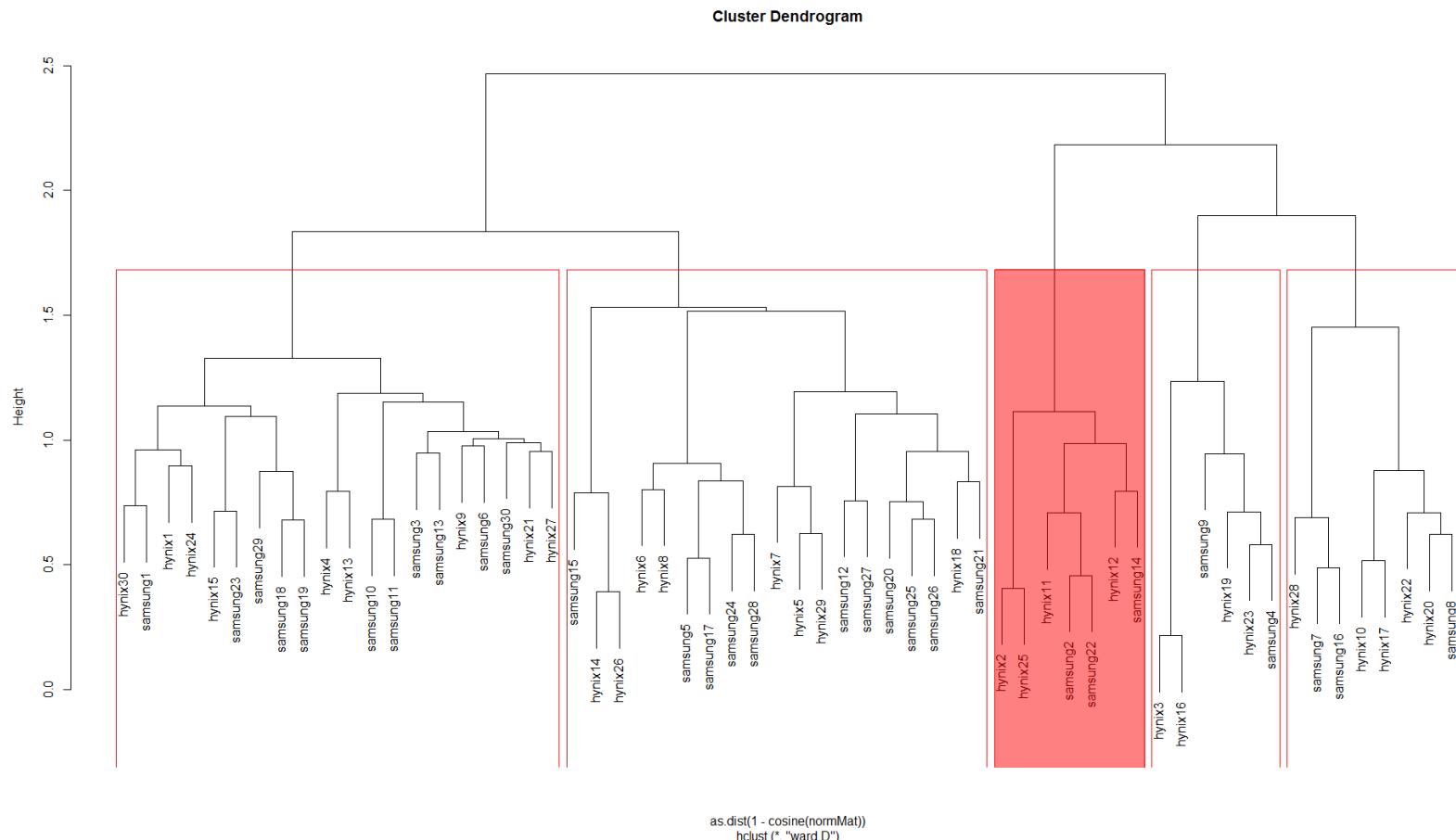
Cluster	Size	Shared Phrases and Sample Document Titles
1 View Results Refine Query Based On This Cluster	16	Society and Culture (56%), Faiths and Practices (56%), Judaism (69%), Spirituality (56%); Religion (56%), organizations (43%) Ahavat Israel - The Amazing Jewish Website! Israel and Judaism Judaica Collection
2 View Results Refine Query Based On This Cluster	15	Ministry of Foreign Affairs (33%), Ministry (87%) Publications and Data of the BANK OF ISRAEL Consulate General of Israel to the Mid-Atlantic Region The Friends of Israel Gospel Ministry
3 View Results Refine Query Based On This Cluster	11	Israel Tourism (36%), Comprehensive Israel (36%), Tourism (64%) Interactive Israel tourism guide - Jerusalem Ambassade d'Israël Travel to Israel Opportunities
4 View Results Refine Query Based On This Cluster	7	Middle East (57%), History (57%); WAR (42%), Region (42%), Complete (42%), Listing (42%), country (42%) Israel at Fifty: Our Introduction to The Six Day War Machal - Volunteers in the Israel's War of Independence HISTORY: The State of Israel
5 View Results Refine Query Based On This Cluster	22	Economy (68%), Companies (55%), Travel (55%) Israel Hotel Association Israel Association of Electronics Industries Focus Capital Group - Israel



Clustering Overview

- Where are clustering used?

✓ Strategic planning: Patent Analysis



Clustering Overview

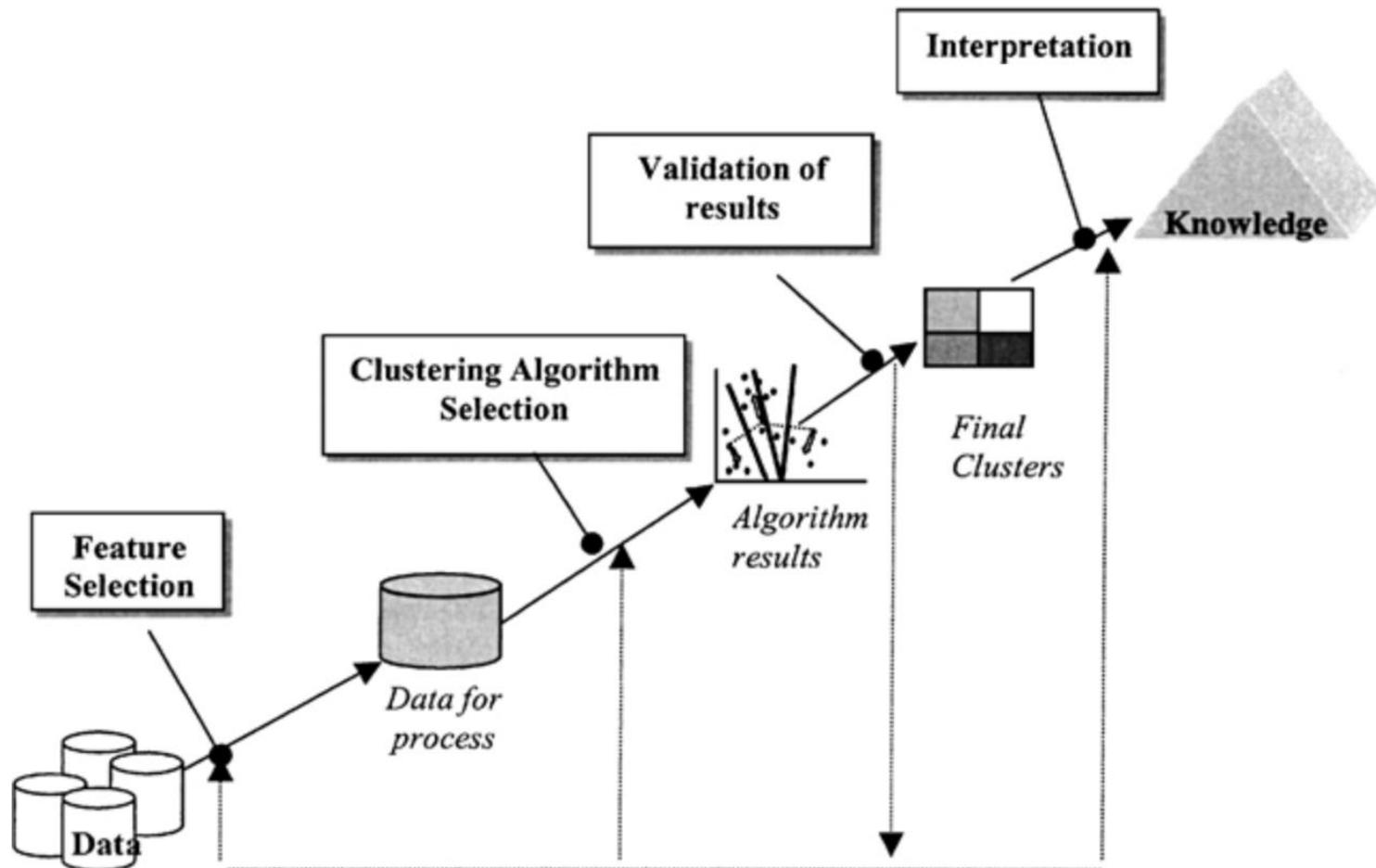
- Where are clustering used?

✓ Strategic planning: Patent Analysis

1	회사	일련번호	특허명	초록
2	SK하이닉스	2	멀티 레귤레이터 회로 및 이를 구비한 집적회로	본 기술에 따른 레귤레이터 회로는, 입력전압을 일정한 전압 레벨로 레귤레이팅하여 출력하도록 구성된 레귤레이터 및 복수개의 전압 생성 코드 들에 의해 결정되는 내부 저항값들에 따라 상기 레귤레이터의 출력 전압을 분배한 분배전압들을 각각 출력하도록 구성된 복수개의 전압 분배회로를 포함한다.
3	SK하이닉스	11	내부 전압 생성 회로 및 그의 동작 방법	펌핑 동작을 통해 내부 전압을 생성하는 내부 전압 생성 회로에 관한 것으로, 다수의 펌핑부를 포함하며, 목표 전압 레벨에 대응하는 최종 펌핑 전압을 생성하기 위한 펌핑 전압 생성부, 및 상기 목표 전압 레벨에 대응하여 상기 다수의 펌핑부의 활성화 개수를 제어하기 위한 활성화 제어부를 구비하는 내부 전압 생성 회로가 제공된다.
4	SK하이닉스	12	자기 메모리 장치를 위한 라이트 드라이버 회로 및 자기 메모리 장치	비트라인과 소스라인 간에 접속되며, 비트라인 방향으로 인접하는 한 쌍의 자기 메모리 셀이 소스라인을 공유하는 복수의 자기 메모리 셀로 이루어진 메모리 셀 어레이를 포함하는 자기 메모리 장치를 위한 라이트 드라이버 회로로서, 정의의 기록전압 공급단자와 부의 기록전압 공급단자 간에 접속되어, 라이트 인에이블 신호 및 데이터 신호에 따라 정의의 기록전압 또는 부의 기록전압에 의한 전류를 비트라인에 선택적으로 공급하는 스위칭부를 포함하는 자기 메모리 장치를 제공한다.
5	SK하이닉스	25	전압 레귤레이터 및 전압 레귤레이팅 방법	전압 레귤레이터는 출력전압을 전압 출력단으로 출력하는 전압 출력부와, 제1 제어코드의 제어에 따라 분배 저항값을 조절하는 제1 저항분배 스테이지와, 제1 저항분배 스테이지에서 결정된 분배 저항값을 제2 제어코드의 제어에 따라 조절하는 제2 저항분배 스테이지를 포함하며, 전압 출력단을 통해서 출력되는 출력전압의 전압레벨은 제1 및 제2 저항분배 스테이지를 통해서 결정된 상기 분배 저항값과, 기준저항의 저항값 비율에 따라 조절되는 것을 특징으로 한다.
6	삼성전자	2	전압 공급 장치 및 그것을 포함한 불휘발성 메모리 장치	본 발명에 따른 전압 공급 장치는 전원 전압을 승압하고, 상기 승압된 전압을 출력 라인으로 제공하기 위한 전하 펌프 및 상기 출력 라인의 전압 레벨을 목표 전압 레벨로 유지하기 위한 전압 제어 회로를 포함한다. 본 발명에 따른 상기 전압 제어 회로는 웰 상에 형성된 제 1 영역 및 제 2 영역을 포함하고, 상기 제 1 영역 및 제 2 영역 사이의 리치 스루(reach through)를 이용하여 상기 출력 라인의 전압 레벨을 제어하기 위한 리치 스루 소자를 포함한다.
7	삼성전자	14	파워 공급 회로 및 이를 구비하는 상 변화 메모리 장치	파워 공급 회로 및 이를 구비하는 상 변화 메모리 장치는 파워 공급 회로, 스위치들 및 선택기들을 구비한다. 파워 공급 회로는 상기 블록들의 메모리 셀 들에 사용되는 제 1 전압 및 제 2 전압을 생성한다. 스위치들은 상기 파워 공급 회로와 상기 제 1 전압이 전달되는 제 1 라인 및 상기 제 2 전압이 전달되는 제 2 라인으로 연결되고, 제어 신호에 응답하여 상기 제 1 전압 및 제 2 전압 중 하나를 대응되는 블록으로 인가한다. 선택기들은 블록 선택 신호 및 디스차아지 성공 신호에 응답하여, 상기 제어 신호를 생성한다. 본 발명에 따른 파워 공급 회로 및 이를 구비하는 상 변화 메모리 장치는 블록마다 별도의 파워 스위치를 구비함으로써 파워 공급 회로의 동작 시간 및 동작 전류를 감소시킬 수 있다. 또한, 기입 전압을 디스차아지한 후 다른 레벨의 전압을 공급함으로써, 상 변화 메모리 장치의 오작동이 방지될 수 있다.
8	삼성전자	22	전압 안정화 장치 및 그것을 포함하는 반도체 장치 및 전압 생성 방법	본 발명은 전압 안정화 장치 및 그것을 이용하는 반도체 장치에 관한 것이다. 본 발명의 기술적 사상의 실시 예에 따른 전압 안정화 장치는 제 1 전압을 생성하는 제 1 레귤레이터 및 상기 제 1 전압보다 낮은 제 2 전압을 생성하는 제 2 레귤레이터를 포함하여, 상기 제 2 레귤레이터는 상기 제 1 전압의 레벨과 미리 정해진 기준 전압의 레벨의 비교 결과에 기초하여 상기 제 1 전압 또는 상기 제 1 전압보다 높은 제 3 전압을 선택적으로 이용하여 상기 제 2 전압을 생성한다. 본 발명의 기술적 사상의 실시 예에 따르면 제 1의 전압>제 2의 전압의 관계를 유지하면서, 동시에 제 2의 전압을 고속으로 전위 변환 시킬 수 있다.

Clustering Overview

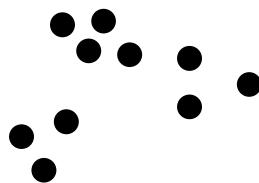
- Clustering procedure



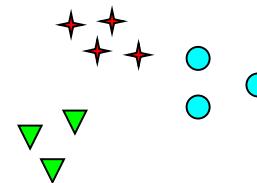
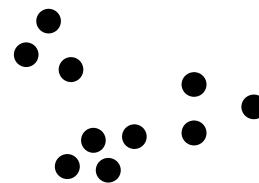
Clustering Overview

- Issues on clustering

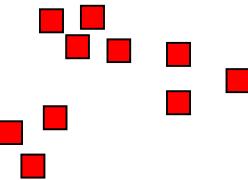
✓ How many clusters are optimal?



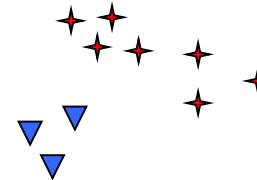
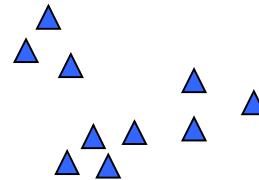
How many clusters?



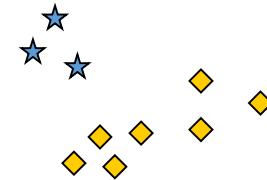
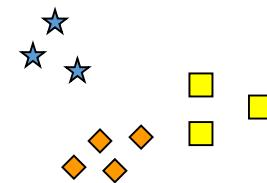
Six Clusters



Two Clusters



Four Clusters

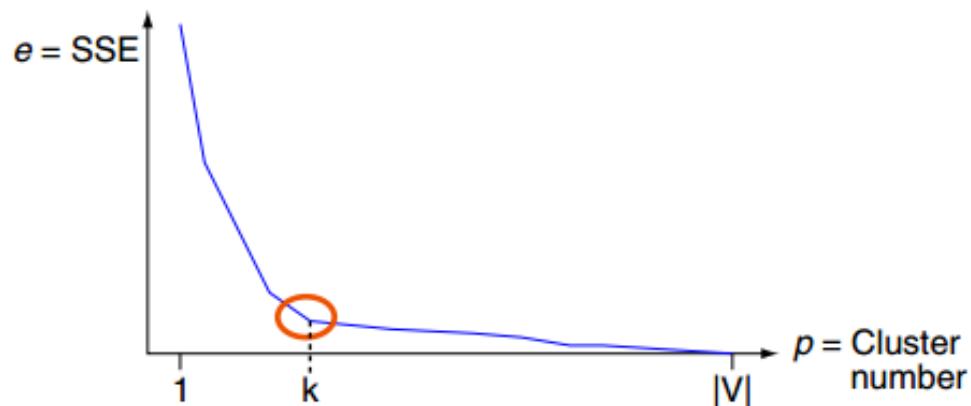
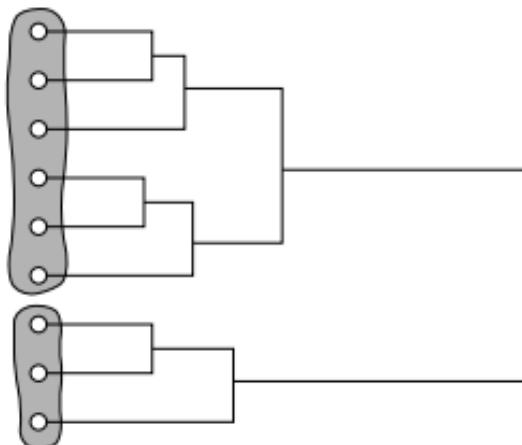


Clustering Overview

- Issues on clustering

- ✓ How many clusters are optimal?

- Use the validity plot for various number of clusters
 - In general, set the elbow point as the optimal number of clusters



Clustering: Issues

- How to evaluate the clustering result?
 - ✓ There is no globally accepted validity measure
 - ✓ Because clustering is an unsupervised learning task, we do not know the exact answer
- Three categories for clustering validity measures
 - ✓ External: Compare the clustering structure with the known answer (**unrealistic**)
 - ✓ Internal: Focusing on the **compactness** of clusters
 - ✓ Relative: Focusing on both the **compactness** of clusters and **separation** between clusters

Clustering Overview

- Issues on clustering

- ✓ How to evaluate the clustering results?

- External: Analyze how close is a clustering to a reference
- Internal: Analyze intrinsic characteristic of a clustering
- Analyze the sensitivity (of internal measures) during clustering generation

External	Internal	Relative
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Rand Statistic	<input type="checkbox"/> Cophenetic Correlation Coefficient	<input type="checkbox"/> Dunn family of indices
<input type="checkbox"/> Jaccard Coefficient	<input type="checkbox"/> Sum of Squared error (SSE)	<input type="checkbox"/> Davies-Bouldin (DB) index
<input type="checkbox"/> Folks and Mallows index	<input type="checkbox"/> Cohesion and separation	<input type="checkbox"/> Semi-partial R-squared
<input type="checkbox"/> (Normalized) Hurbert Γ statistic		<input type="checkbox"/> SD validity index
		<input type="checkbox"/> Silhouette

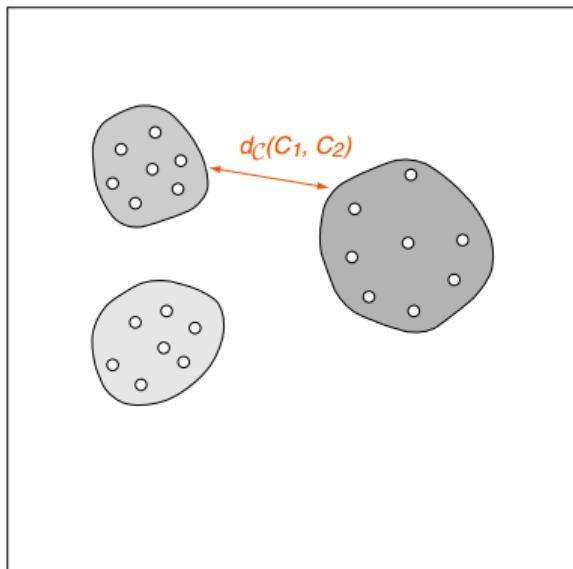
- ✓ There is no single best metric for all clustering tasks

- For more definitions of the clustering validity measures, see the Appendix B.

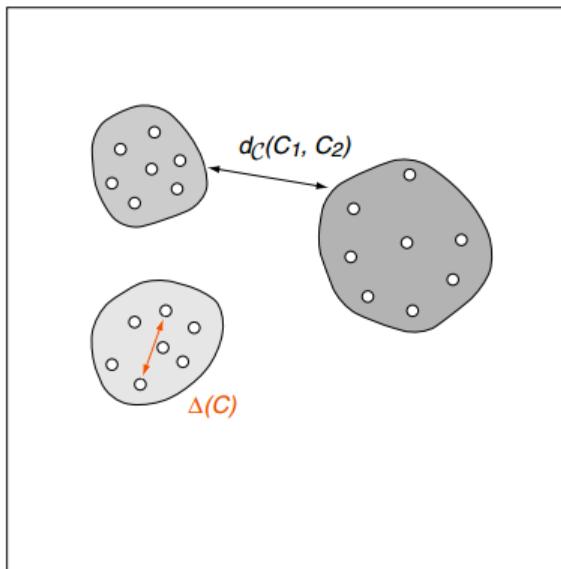
Clustering Validity Index

- Measurements

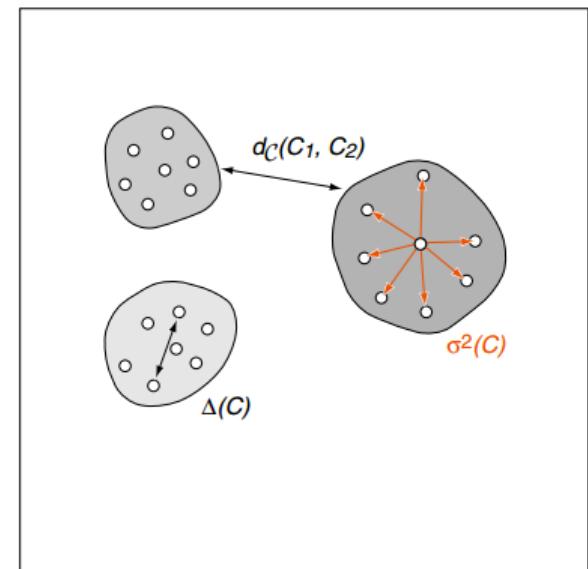
Distance between two clusters



Diameter of a cluster

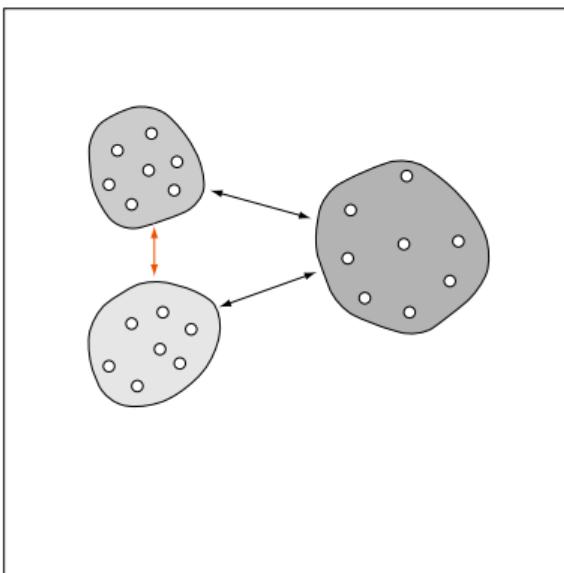


Scatter within a cluster (SSE)



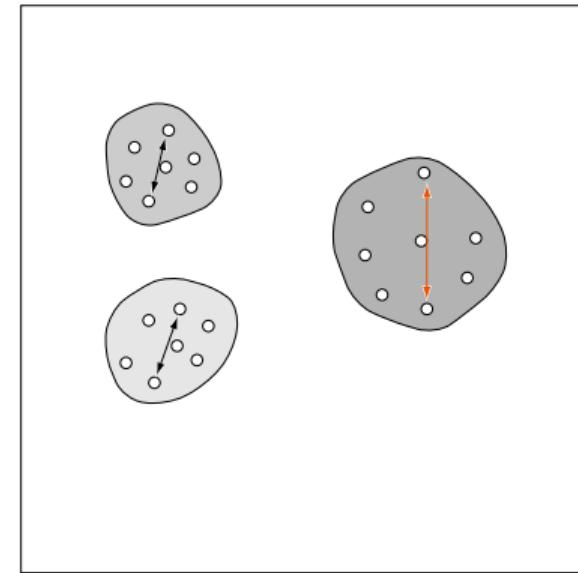
Clustering Validity Index

- Dunn Index



$$I(\mathcal{C}) = \frac{\min_{i \neq j} \{d_{\mathcal{C}}(C_i, C_j)\}}{\max_{1 \leq l \leq k} \{\Delta(C_l)\}},$$

$I(\mathcal{C}) \rightarrow \max$



$$I(\mathcal{C}) = \frac{\min_{i \neq j} \{d_{\mathcal{C}}(C_i, C_j)\}}{\max_{1 \leq l \leq k} \{\Delta(C_l)\}},$$

$I(\mathcal{C}) \rightarrow \max$

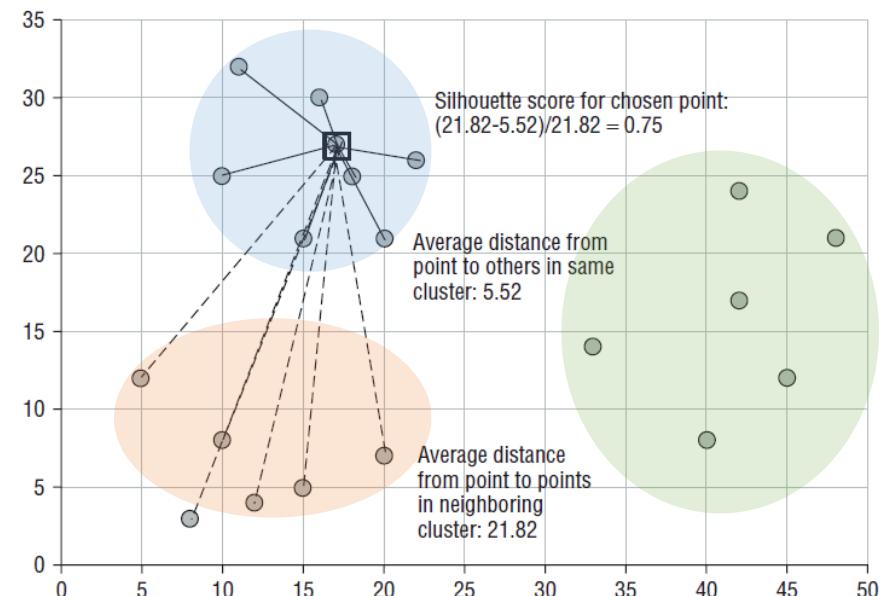
Clustering: Issues

- Clustering Validity Measure Example: Silhouette

- ✓ $a(i)$: the average distance between an instance i and the other instances in the same cluster
- ✓ $b(i)$ the minimum of the average distances between an instance i and the instances in a cluster to which the instance i does not belong

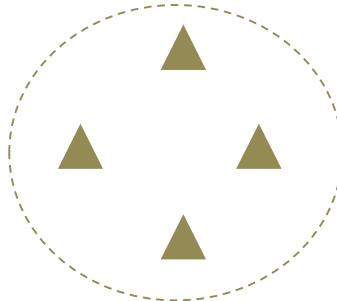
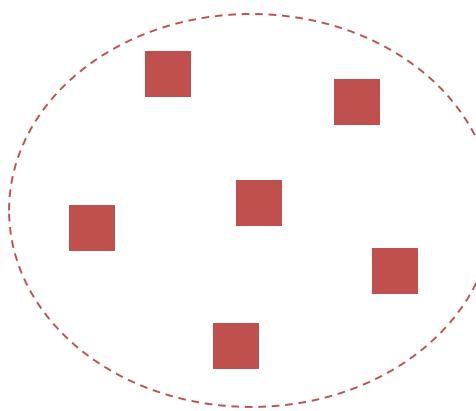
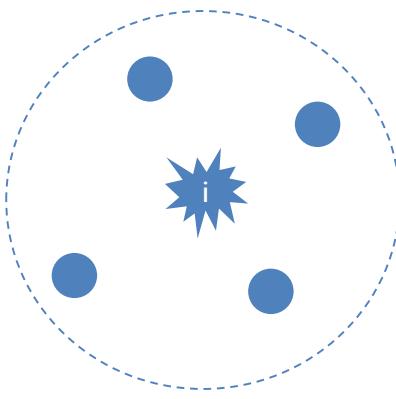
$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

$$s(i) = \begin{cases} 1 - a(i)/b(i), & \text{if } a(i) < b(i) \\ 0, & \text{if } a(i) = b(i) \\ b(i)/a(i) - 1, & \text{if } a(i) > b(i) \end{cases}$$



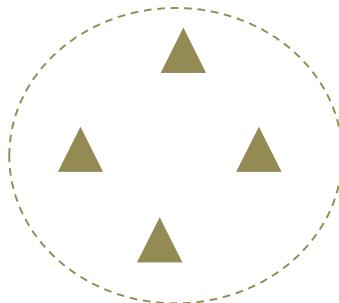
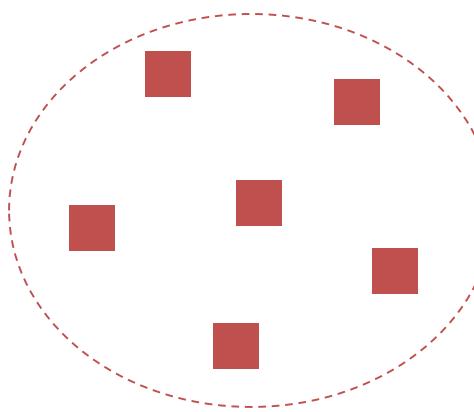
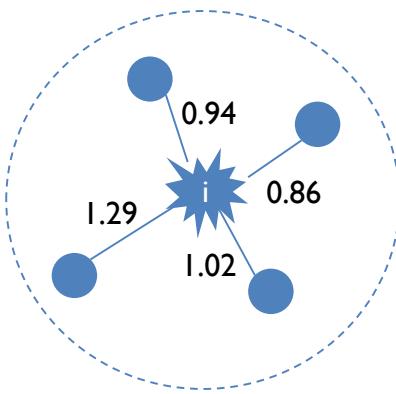
Clustering: Issues

- Clustering Validity Measure Example: Silhouette



Clustering: Issues

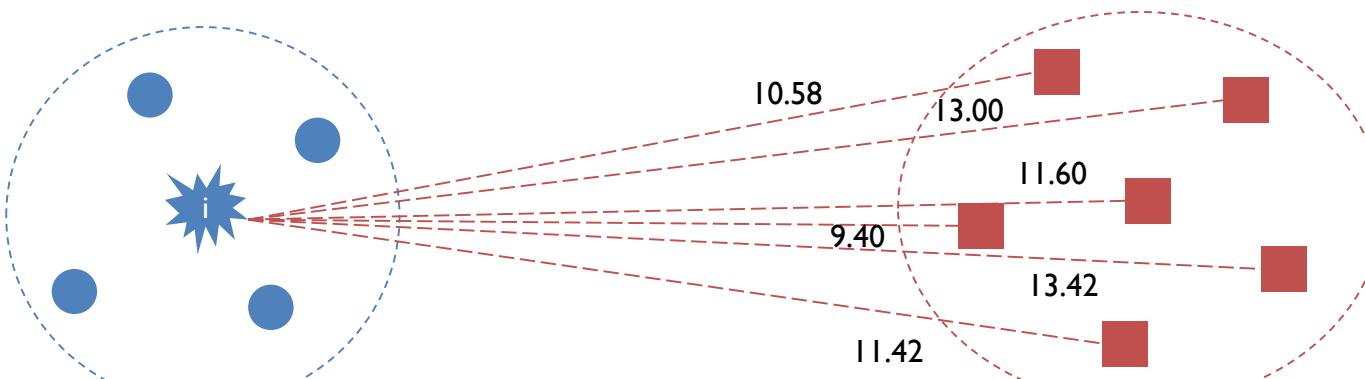
- Clustering Validity Measure Example: Silhouette



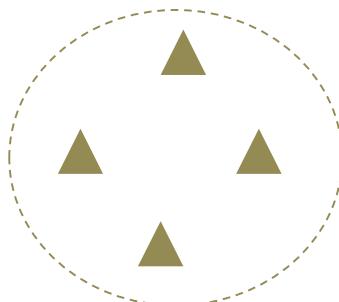
$$a(i) = \frac{1}{4} (0.94 + 0.86 + 1.02 + 1.29) = 1.03$$

Clustering: Issues

- Clustering Validity Measure Example: Silhouette

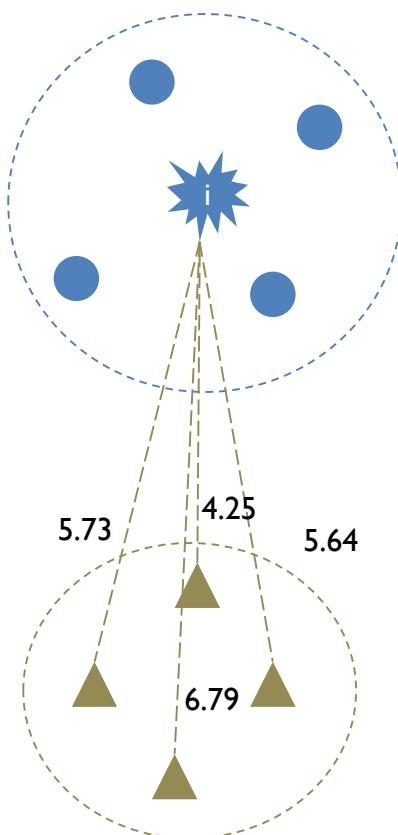


$$\frac{1}{6} (10.58 + 13.00 + 11.60 + 9.40 + 13.42 + 11.42) = 11.57$$

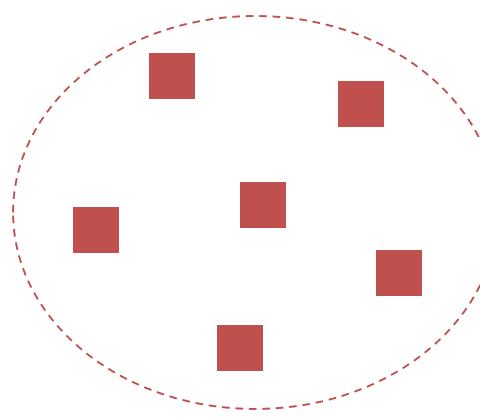


Clustering: Issues

- Clustering Validity Measure Example: Silhouette



$$\frac{1}{4} (5.73 + 6.79 + 4.25 + 5.64) = 5.60$$



$$b(i) = \min(11.57, 5.60) = 5.60$$

$$s(i) = \frac{5.60 - 1.03}{\max(1.03, 5.60)}$$

$$= \frac{4.57}{5.60} = 0.82$$

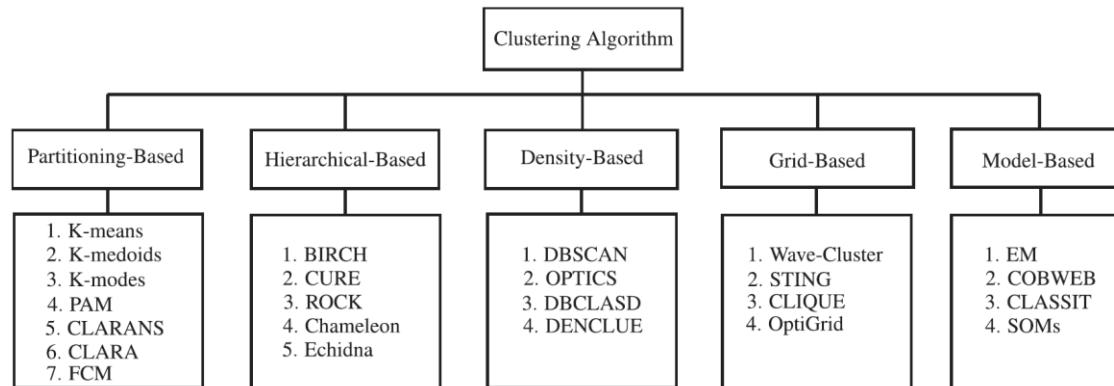
Type of Clustering

Fahad et al. (2014)

- Hard Clustering vs. Soft Clustering

- ✓ Hard Clustering (Crisp Clustering)

- Produces disjoint (non-overlapping) clusters
 - Each document belongs to only one cluster



- ✓ Soft Clustering (Fuzzy Clustering)

- Produces overlapping clusters
 - A document can have multiple cluster memberships
 - Techniques used in topic modeling (pLSI, LDA, etc.)

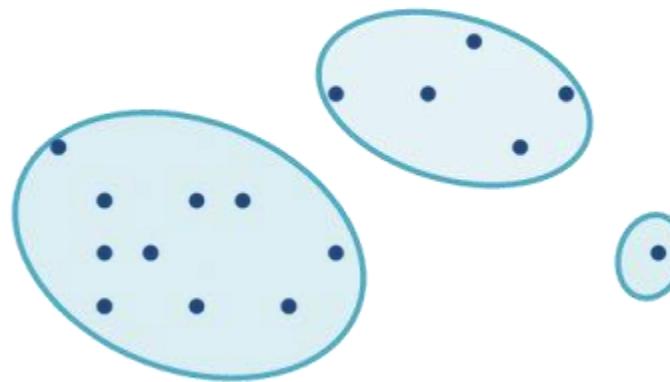
Type of Clustering

- Type of clustering techniques

- ✓ Partitional clustering

- Divide data into non-overlapping subsets such that each data object is in exactly one subset

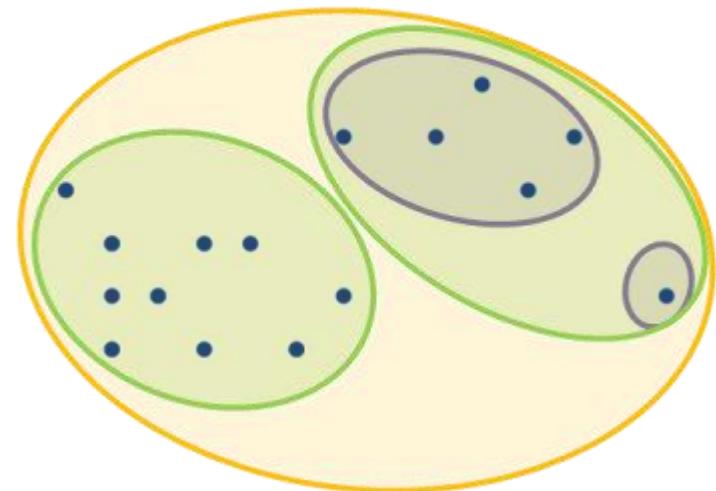
Partitional Clustering



- ✓ Hierarchical clustering

- A set of nested clusters organized as a hierarchical tree

Hierarchical Clustering

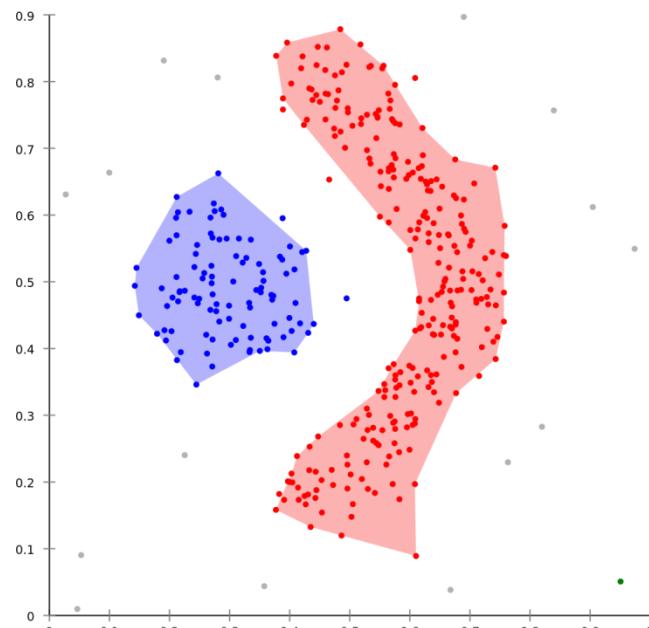


Type of Clustering

- Type of clustering techniques

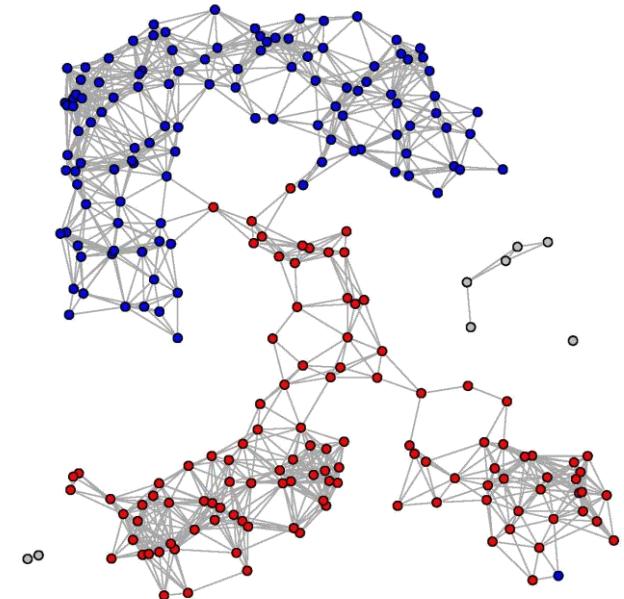
- ✓ Density-based Clustering

- Considers relative density of data space



- ✓ Graph-based Clustering

- Define a subset in a nodes in a graph such that nodes in the same cluster are densely connected whereas nodes in different clusters are sparsely connected

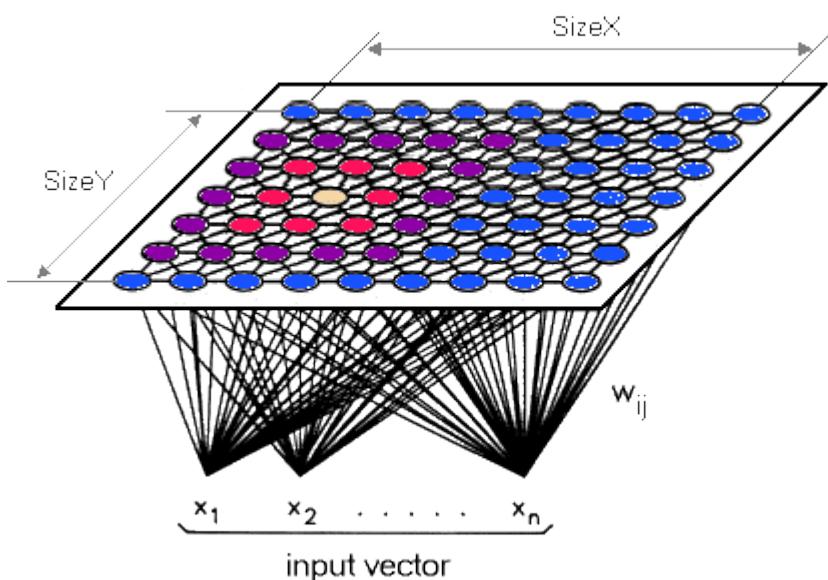


Type of Clustering

- Type of clustering techniques

- ✓ Self-Organizing Map (SOM)

- Clustering data and ordering of the cluster centers in a two dimensional grid through neural network-based learning methods



AGENDA

- 01 Document Similarity
- 02 Clustering Overview
- 03 Partition-based Clustering
- 04 Hierarchical Clustering
- 05 Density-based Clustering
- 06 Self-Organizing Map

K-Means Clustering

- K-Means Clustering (KMC)
 - ✓ Partitional clustering approach
 - Each cluster is associated with a centroid
 - Each point is assigned to the cluster with the closest centroid
 - Number of cluster, K, must be specified

$$\mathbf{X} = C_1 \cup C_2 \dots \cup C_K, \quad C_i \cap C_j = \emptyset, \quad i \neq j$$

$$\arg \min_{\mathbf{C}} \sum_{i=1}^K \sum_{\mathbf{x}_j \in C_i} \|\mathbf{x}_j - \mathbf{c}_i\|^2$$

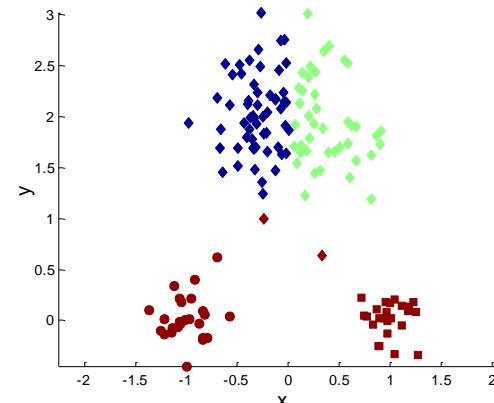
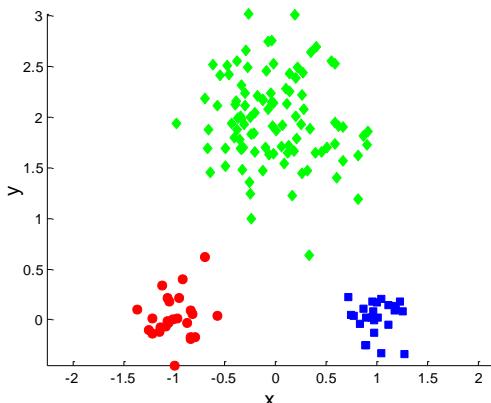
K-Means Clustering

- K-Means Clustering Procedure

- ✓ Algorithm

- 1: Select K points as the initial centroids.
 - 2: **repeat**
 - 3: Form K clusters by assigning all points to the closest centroid.
 - 4: Recompute the centroid of each cluster.
 - 5: **until** The centroids don't change
-

- Initial centroids are often **chosen randomly**: clustering results vary according to the initial centroid selection

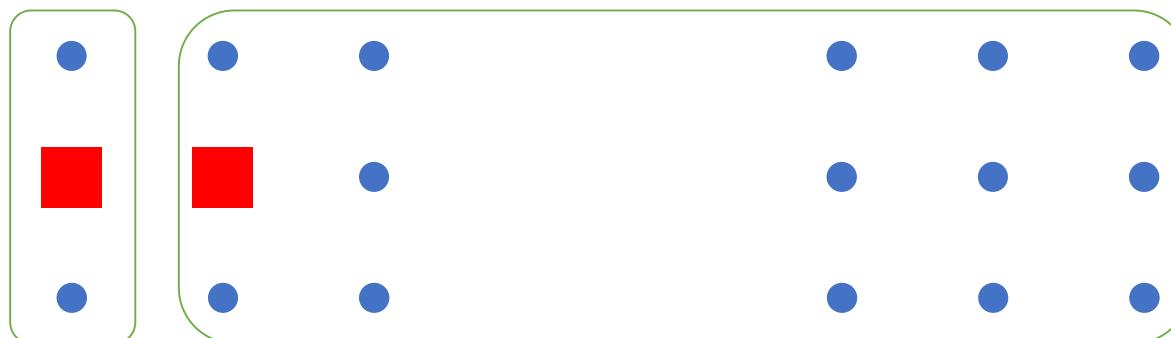


K-Means Clustering

- KMC Example
 - ✓ Step 1: Initialize K centroids



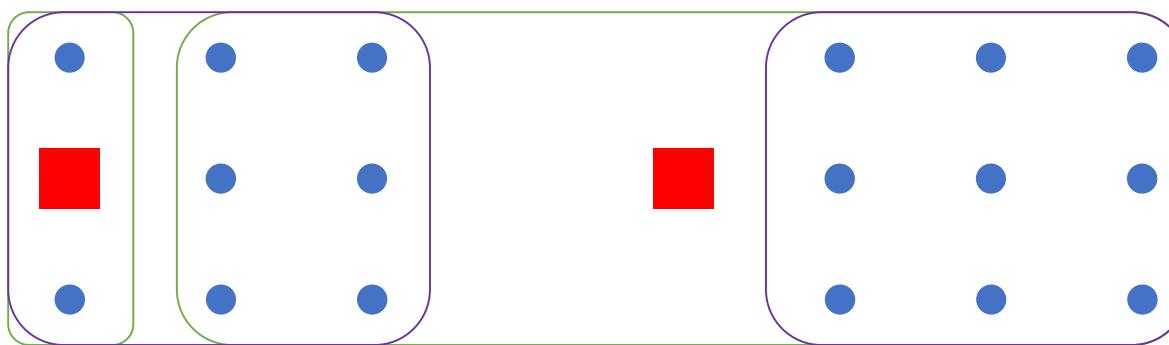
- ✓ Step 2-1 (1st round): Assign all instances to their closest centroid
- ✓ Step 2-2 (1st round): Re-compute the centroids with the assigned instances



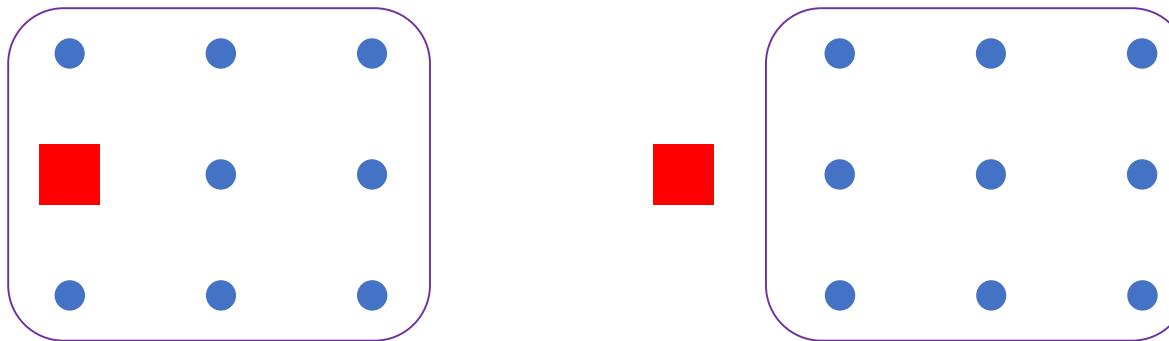
K-Means Clustering

- KMC Example

- ✓ Step 2-1 (2nd round): Assign all instances to their closest centroid



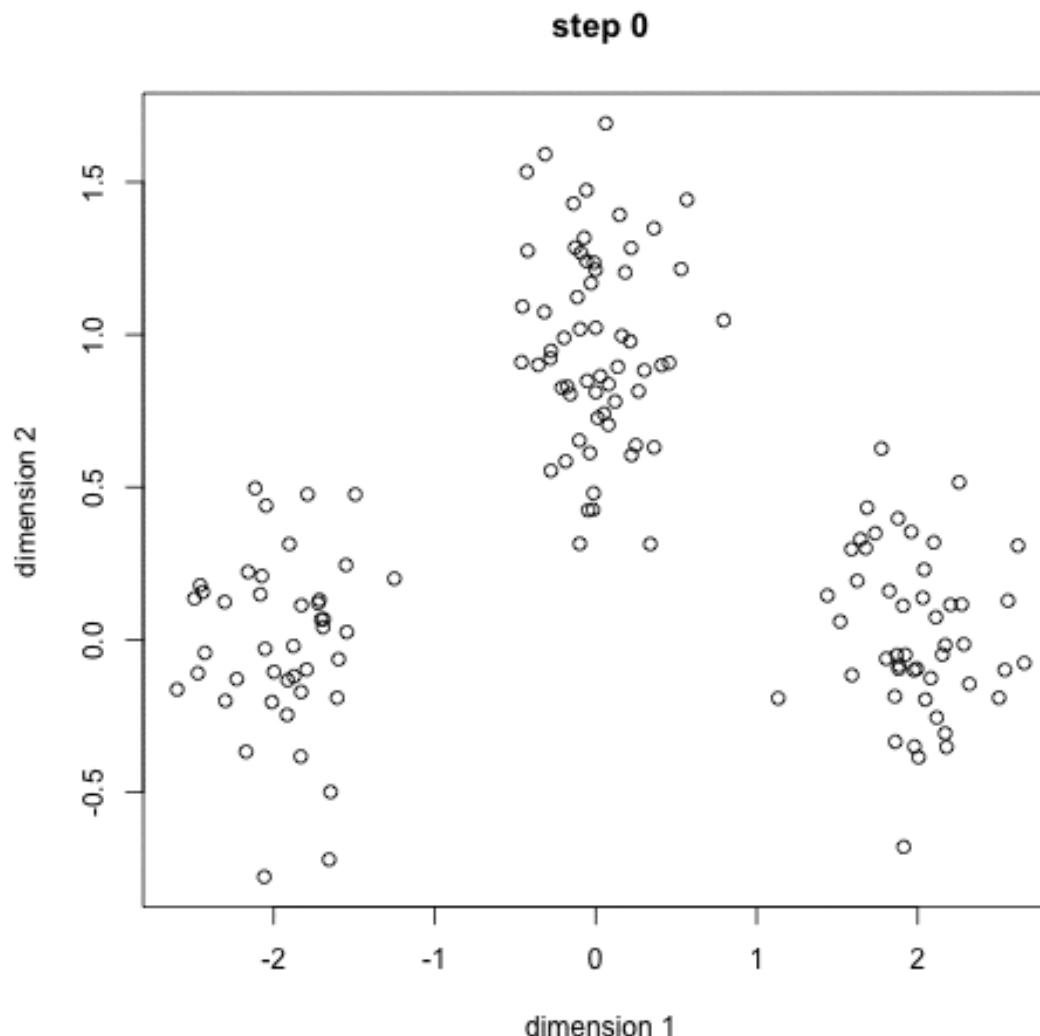
- ✓ Step 2-2 (2nd round): Re-compute the centroids with the assigned instances



- Centroids and cluster membership assignments are not changed → End the procedure

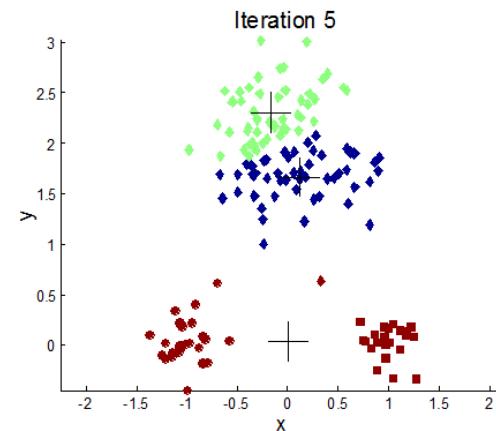
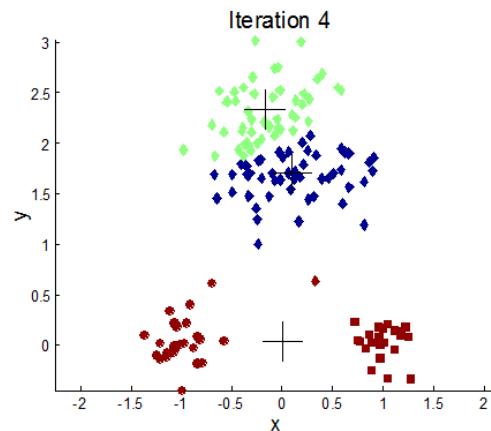
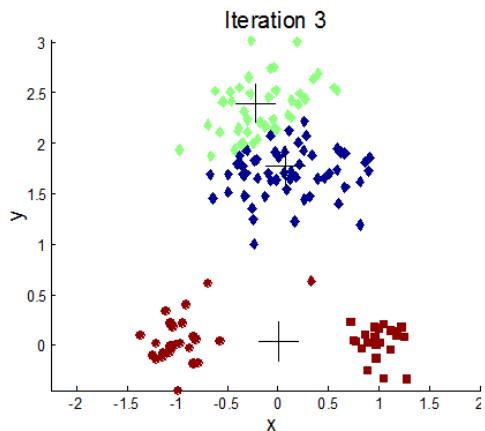
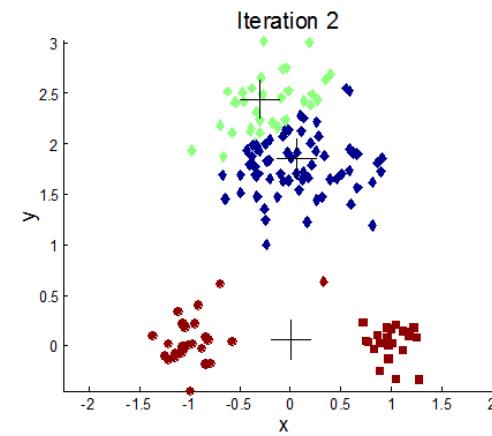
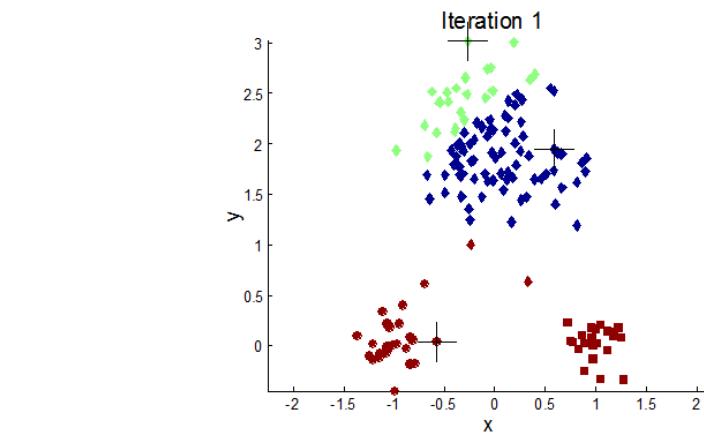
K-Means Clustering

- KMC example



K-Means Clustering

- Effects of initial centroids
 - ✓ Undesirable centroid selection



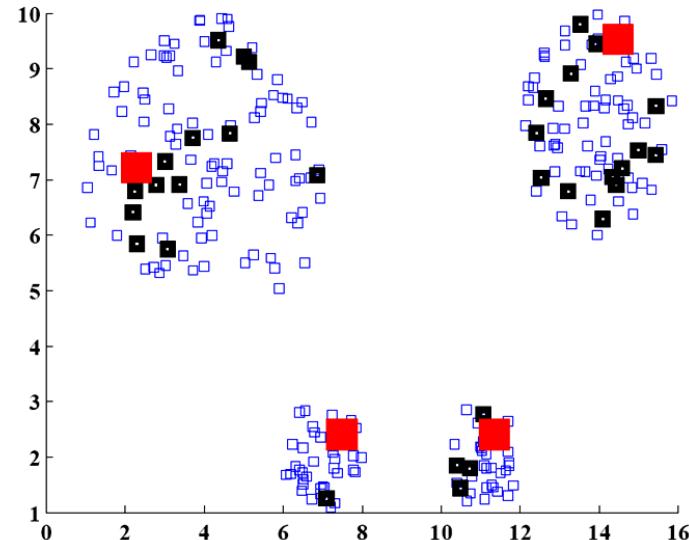
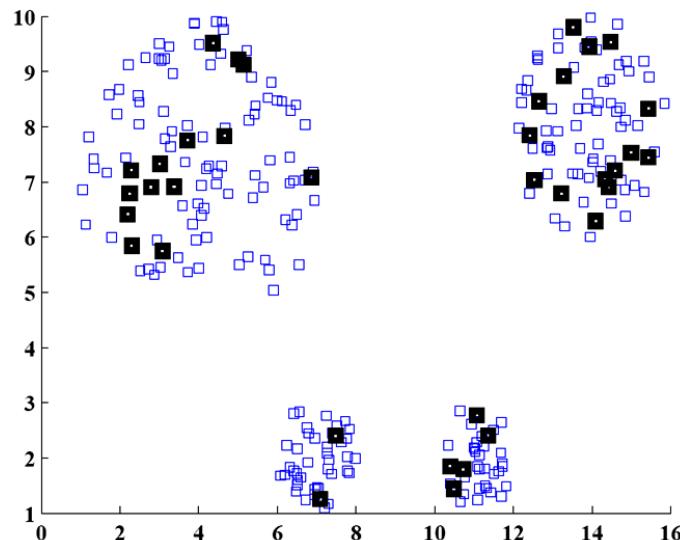
K-Means Clustering

Kang and Cho (2009)

- Some remedies for initial centroid selection

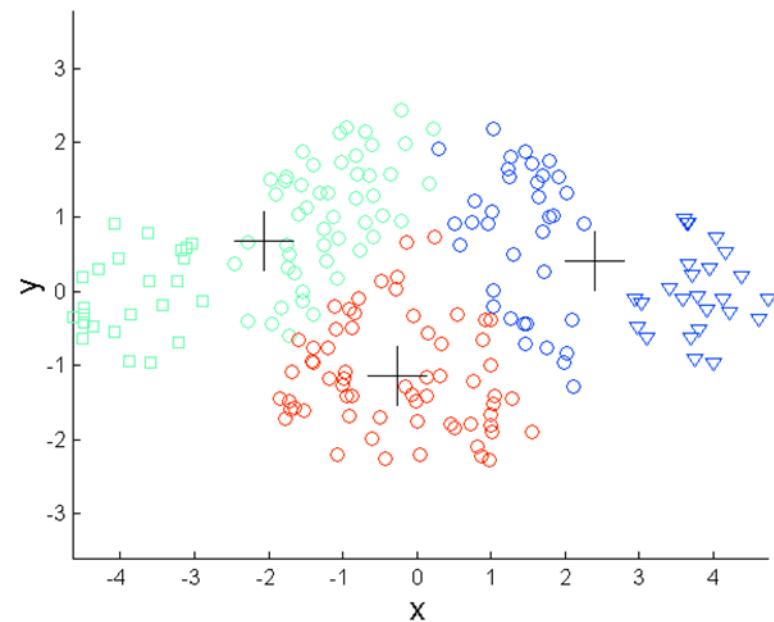
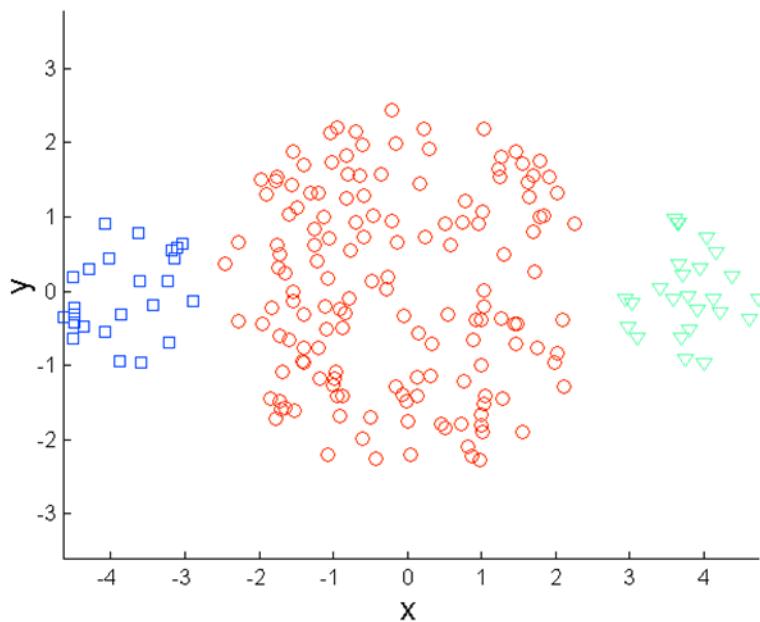
- ✓ Multiple runs
- ✓ Sample and use hierarchical clustering to determine initial centroids
- ✓ Preprocessing & Postprocessing

$$\mathcal{L}(\mathbf{x}_s | \mathbf{S}, \mathbf{C}) = d_G(\mathbf{x}_s, \mathbf{S}) \times \frac{1}{1 + \exp(-d_R(\mathbf{x}_s, \mathbf{S}))}$$



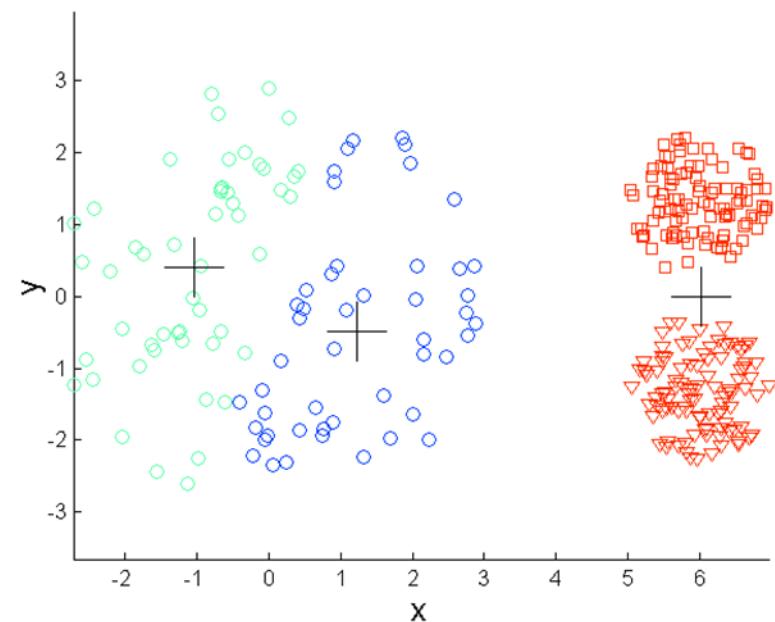
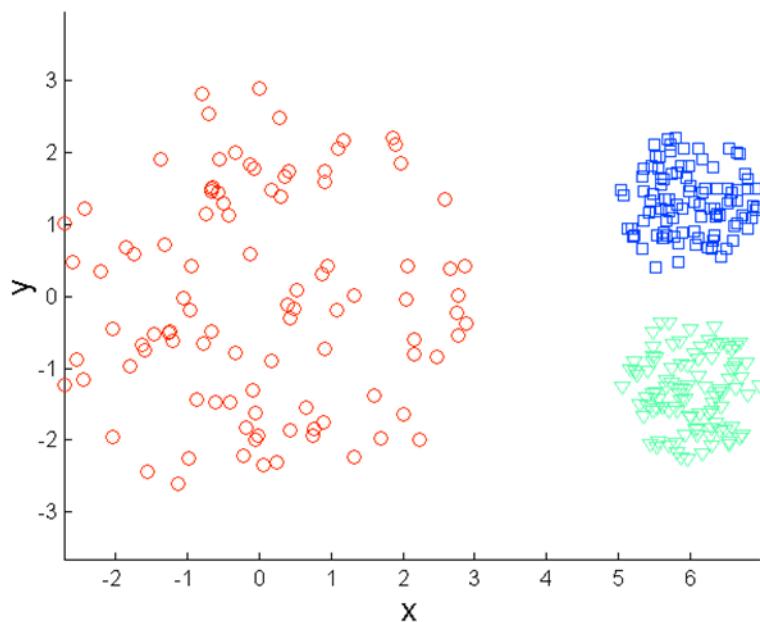
K-Means Clustering

- Limitations of K-Means Clustering
 - ✓ Cannot cope with different sizes



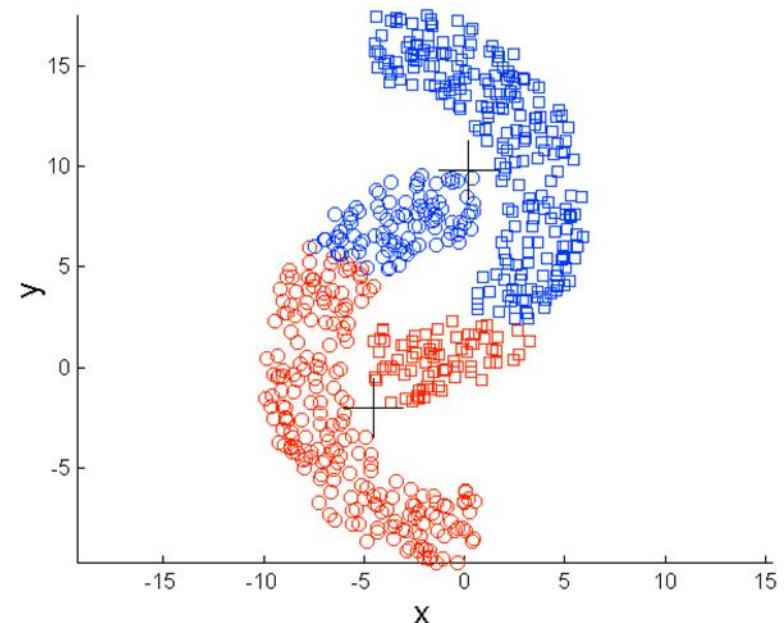
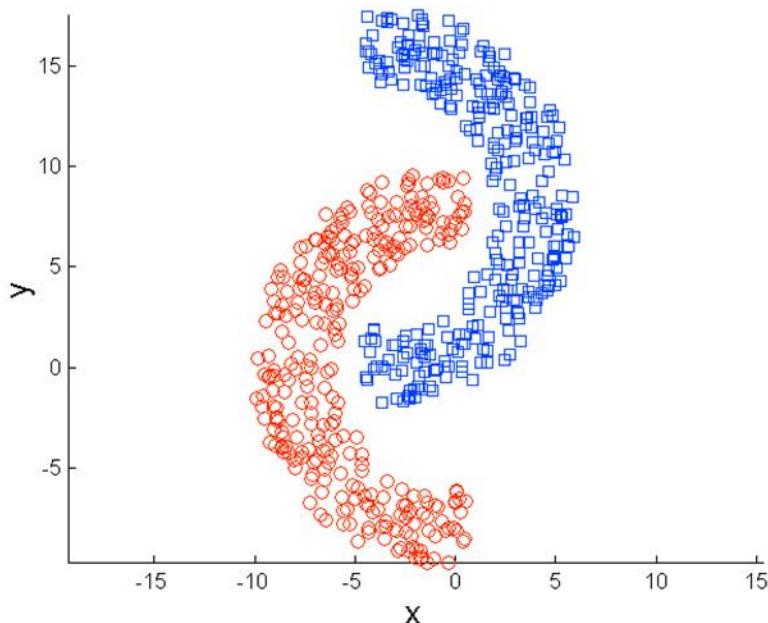
K-Means Clustering

- Limitations of K-Means Clustering
 - ✓ Cannot cope with different densities



K-Means Clustering

- Limitations of K-Means Clustering
 - ✓ Cannot cope with non-globular shapes



AGENDA

01 Document Similarity

02 Clustering Overview

03 Partition-based Clustering

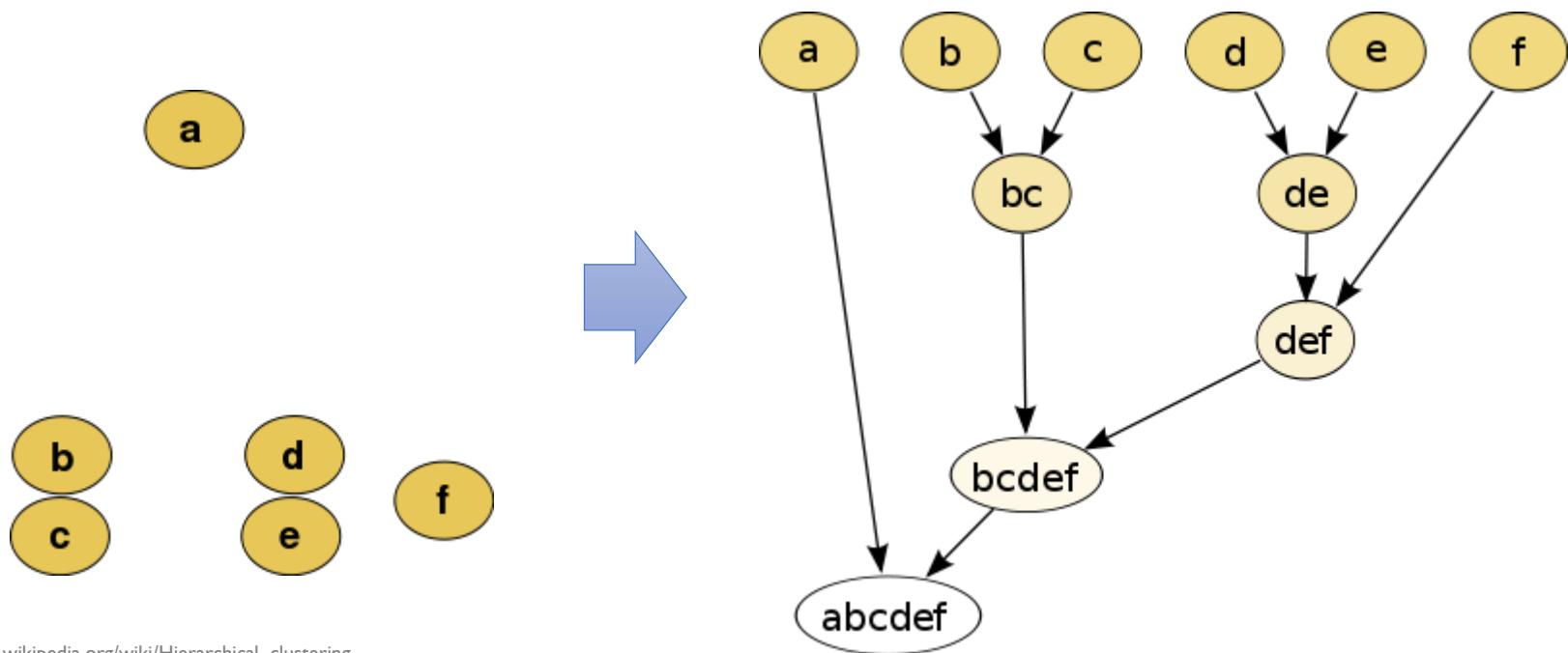
04 Hierarchical Clustering

05 Density-based Clustering

06 Self-Organizing Map

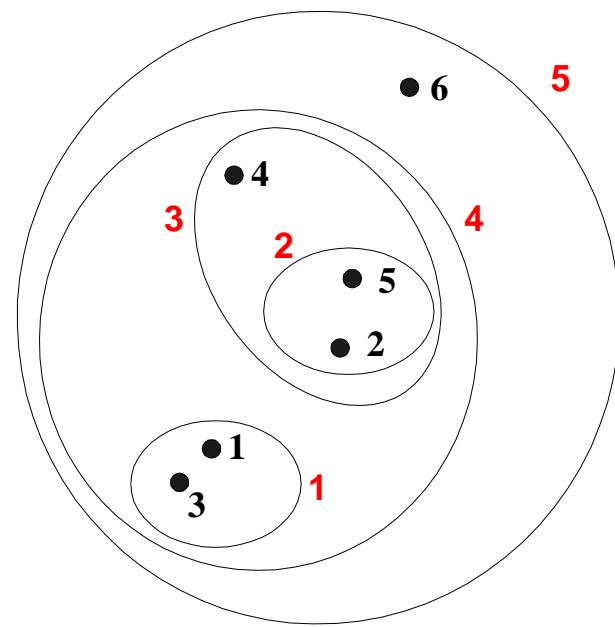
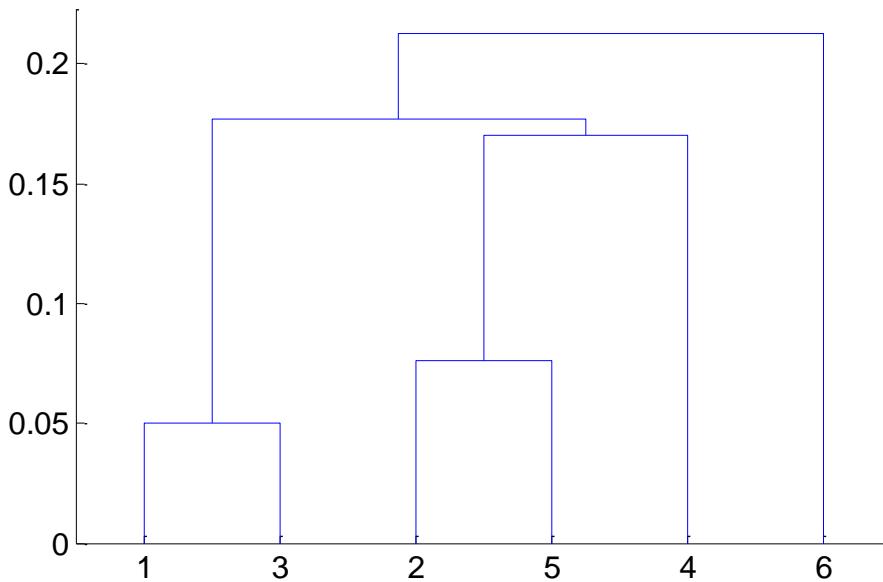
Hierarchical Clustering

- Hierarchical clustering
 - ✓ Produces a set of nested clusters organized as a hierarchical tree
 - ✓ Can be visualized as a dendrogram
 - A tree like diagram that records the sequences of merges or splits



Hierarchical Clustering

- Hierarchical clustering
 - ✓ Produces a set of nested clusters organized as a hierarchical tree
 - ✓ Can be visualized as a dendrogram
 - A tree like diagram that records the sequences of merges or splits

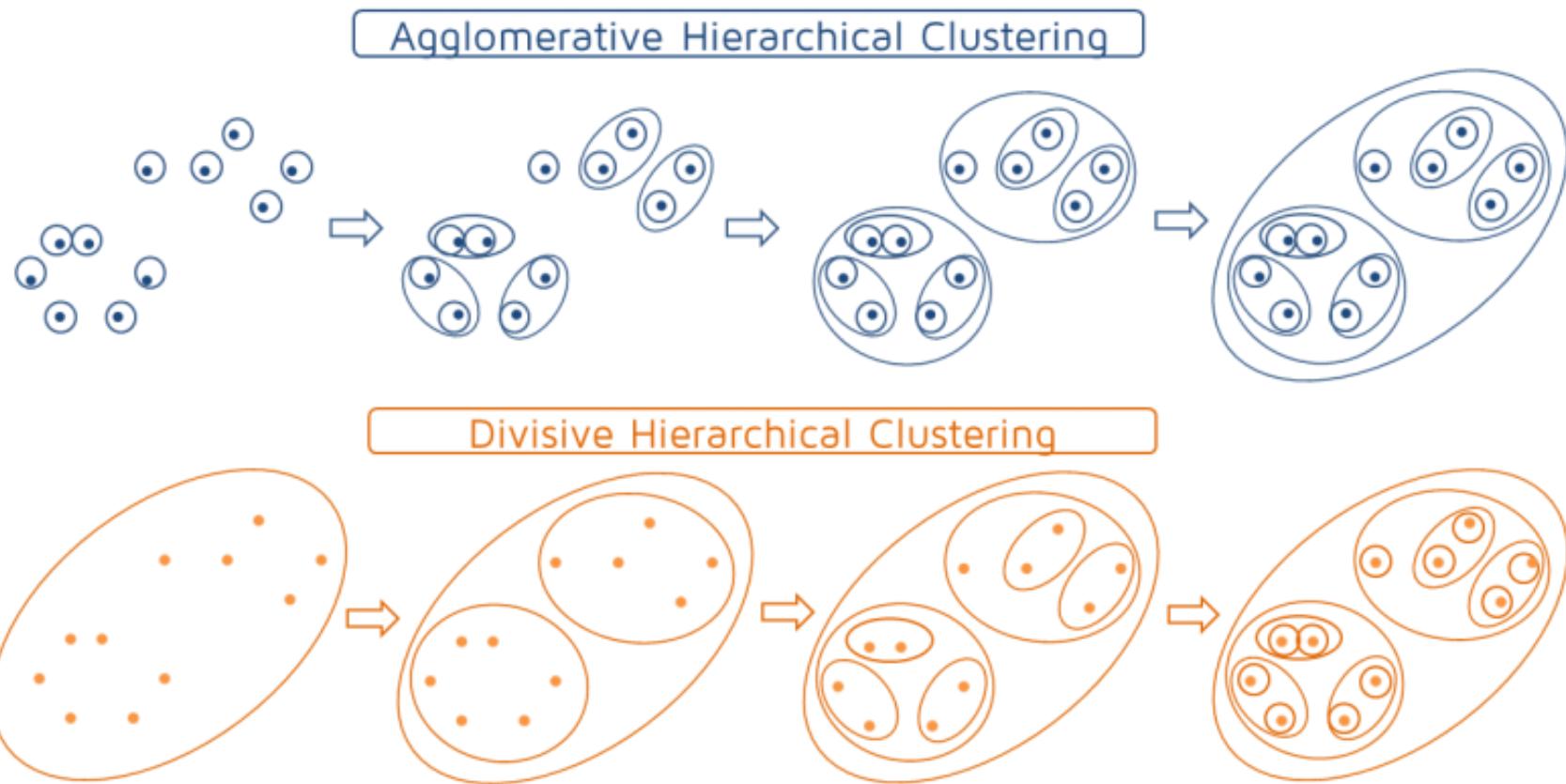


Hierarchical Clustering

- Strengths of Hierarchical clustering
 - ✓ Do not have to assume any particular number of clusters
 - Any desired number of clusters can be obtained by ‘cutting’ the dendrogram at the proper level
 - ✓ May correspond to meaningful taxonomies
- Two main types of hierarchical clustering
 - ✓ Agglomerative clustering
 - Start with the points as individual clusters
 - At each step, merge the closest pair of clusters until only one cluster left
 - ✓ Divisive clustering
 - Start with one, all-inclusive cluster
 - At each step, split a cluster until each cluster contains a point

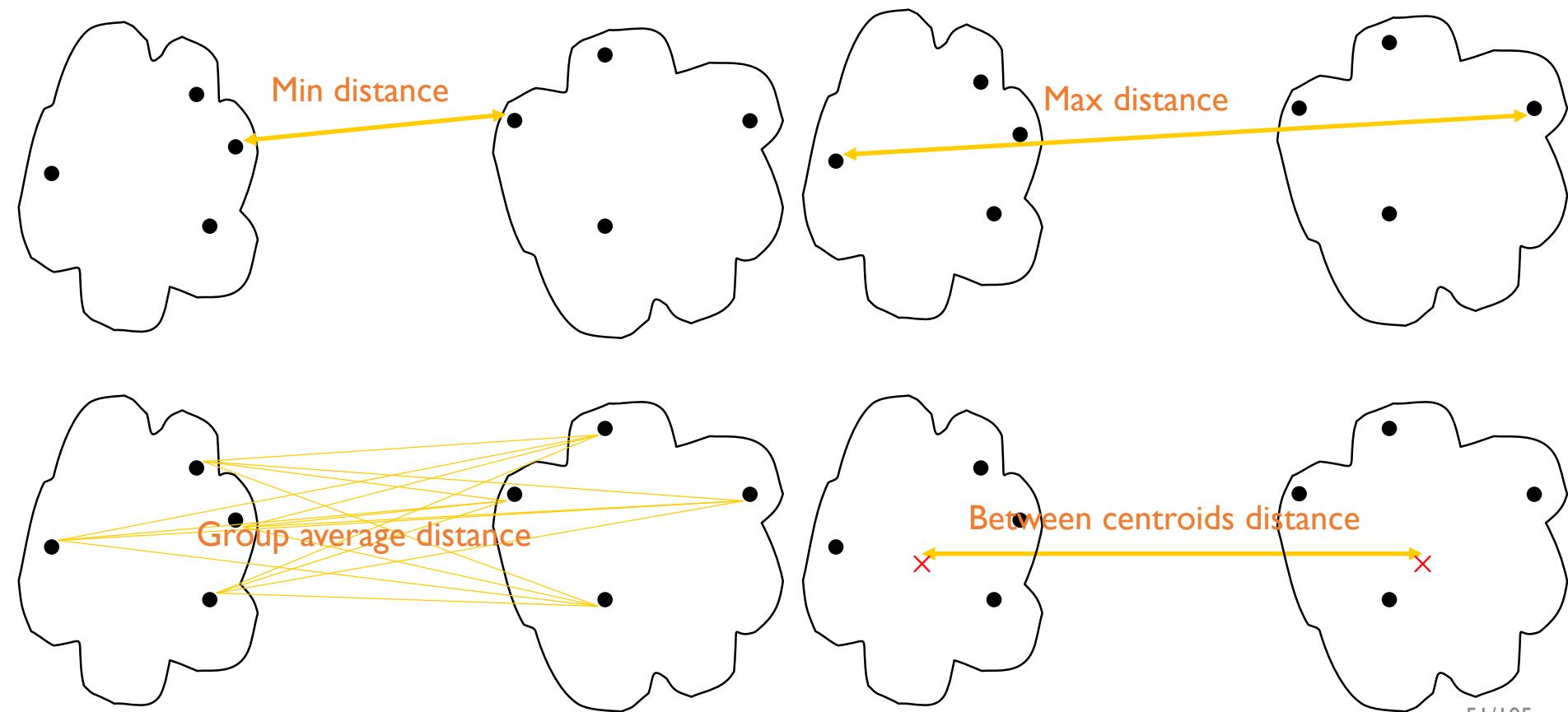
Hierarchical Clustering

- Strengths of Hierarchical clustering
 - ✓ Agglomerative clustering vs. Divisive clustering



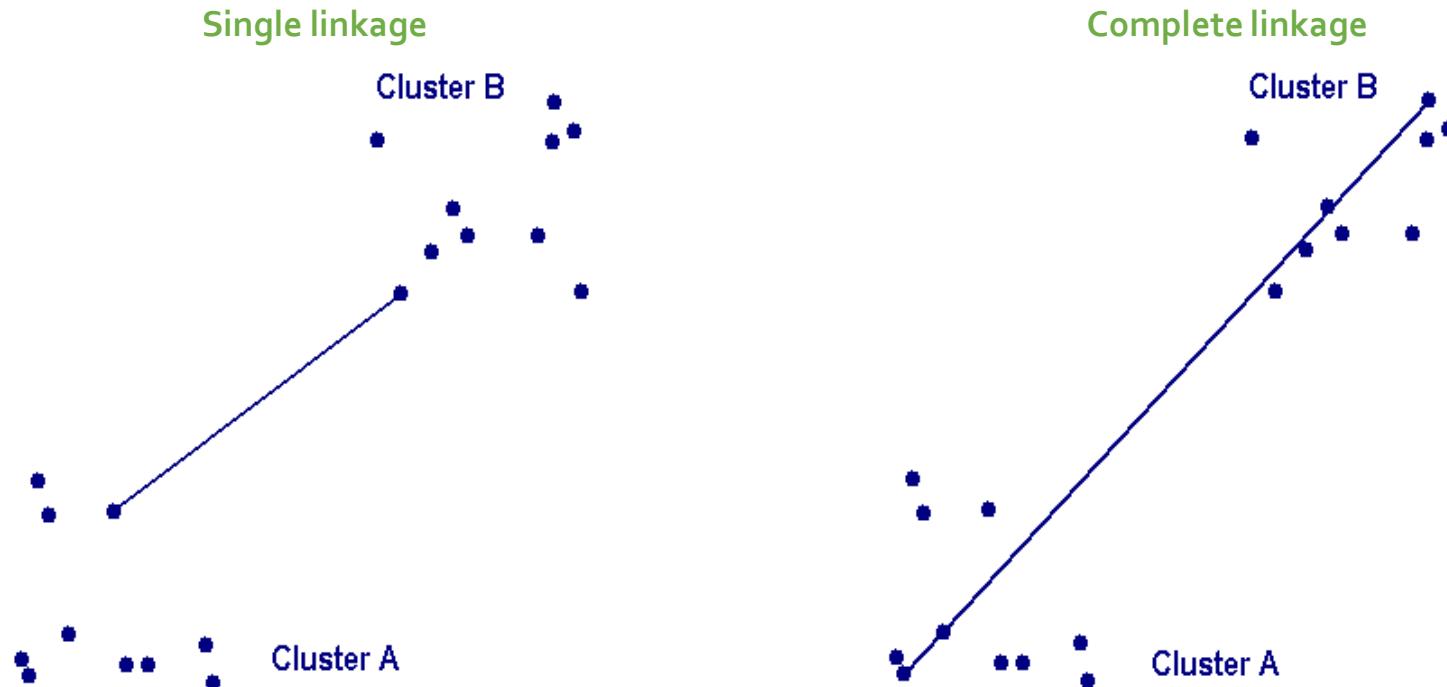
Hierarchical Clustering

- Agglomerative clustering algorithm
 - ✓ Key operation: computation of the proximity of two clusters
 - Min, max, group average, between centroid, etc.



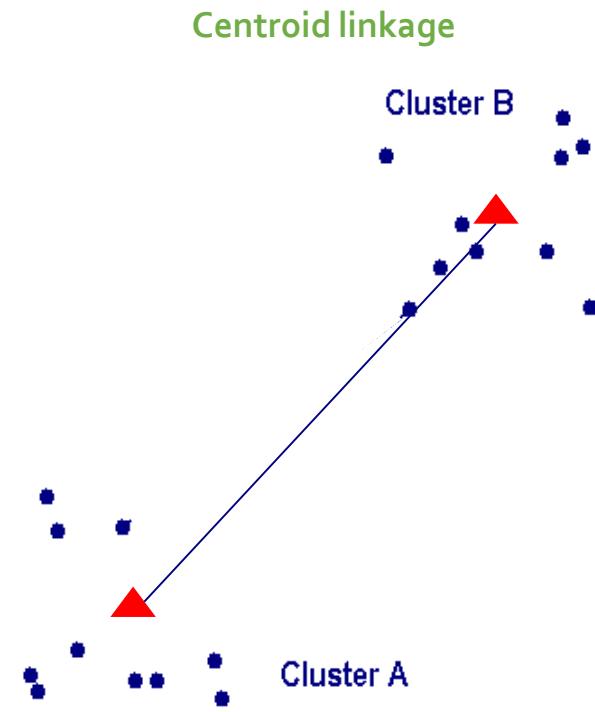
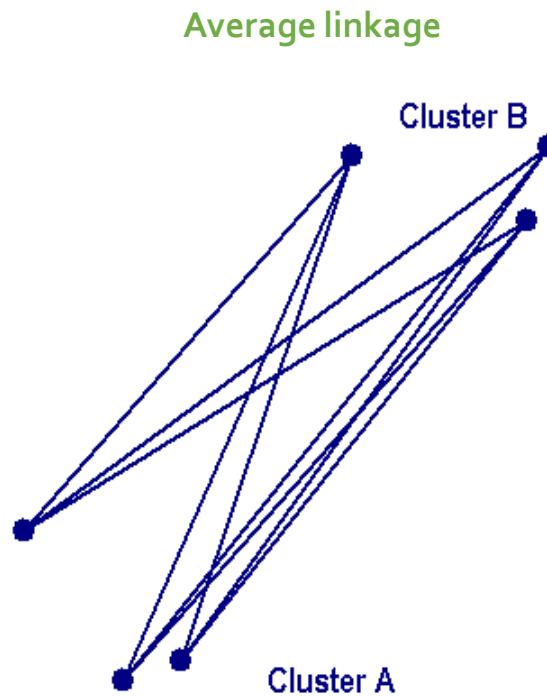
Hierarchical Clustering

- Agglomerative clustering algorithm
 - ✓ Single linkage: minimum distance between two data points in different clusters
 - ✓ Complete linkage: maximum distance between two data points in different clusters



Hierarchical Clustering

- Agglomerative clustering algorithm
 - ✓ Average linkage: mean distance between two data points in different clusters
 - ✓ Centroid linkage: distance between centroids in different clusters



Hierarchical Clustering

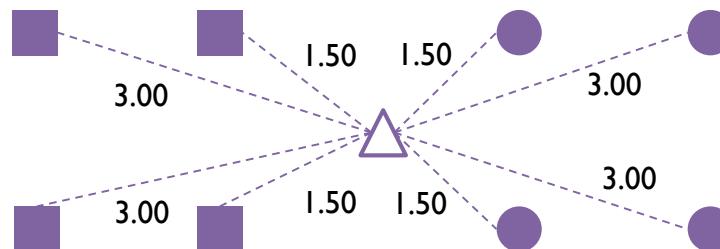
- Agglomerative clustering algorithm

✓ Ward method: Compare the sum of squared error (SSE) before and after the merge

- SSE before merge: $1^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2 = 8$



- SSE after merge: $4 \times 1.5^2 + 4 \times 3^2 = 45$



- Ward distance: $45 - 8 = 37$

Hierarchical Clustering

- Agglomerative clustering algorithm
 1. Compute the proximity matrix
 2. Let each data point be a cluster

3. Repeat

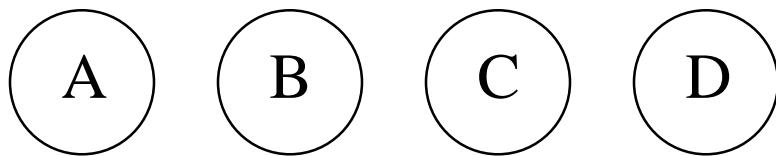
 1. Merge the two closest clusters
 2. Update the proximity matrix

4. Until only a single cluster remains

Hierarchical Clustering

- HC example

Initial Data Items



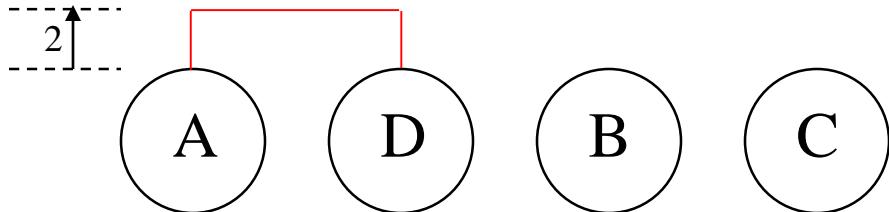
Distance Matrix

Dist	A	B	C	D
A		20	7	2
B			10	25
C				3
D				

Hierarchical Clustering

- HC example

Current Clusters



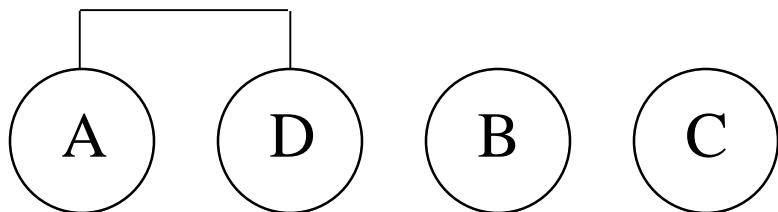
Distance Matrix

Dist	A	B	C	D
A		20	7	2
B			10	25
C				3
D				

Hierarchical Clustering

- HC example

Current Clusters



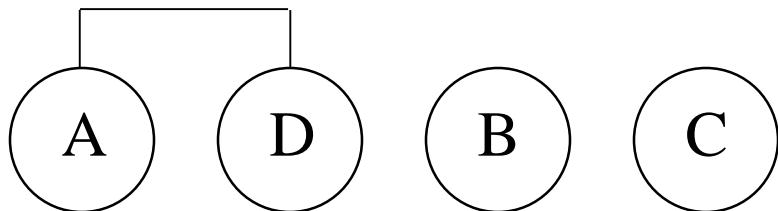
Distance Matrix

Dist	AD	B	C	
AD		20	3	
B			10	
C				

Hierarchical Clustering

- HC example

Current Clusters



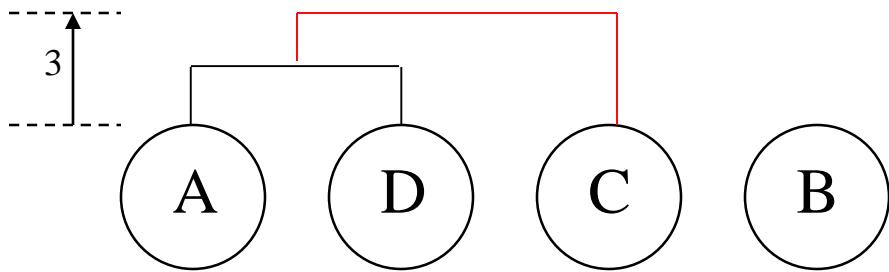
Distance Matrix

Dist	AD	B	C	
AD		20	3	
B			10	
C				

Hierarchical Clustering

- HC example

Current Clusters



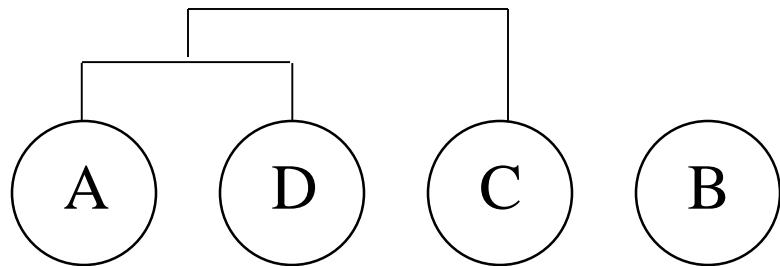
Distance Matrix

Dist	AD	B	C	
AD		20	3	
B			10	
C				

Hierarchical Clustering

- HC example

Current Clusters



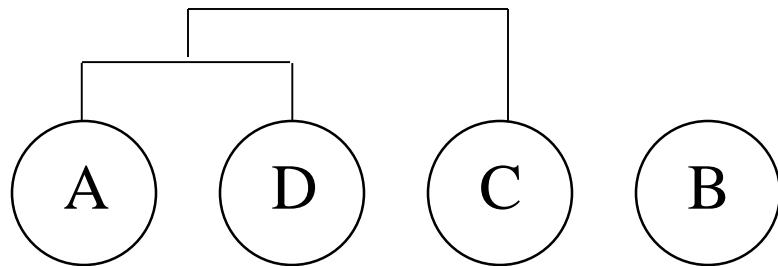
Distance Matrix

Dist	AD C	B		
AD C		10		
B				

Hierarchical Clustering

- HC example

Current Clusters



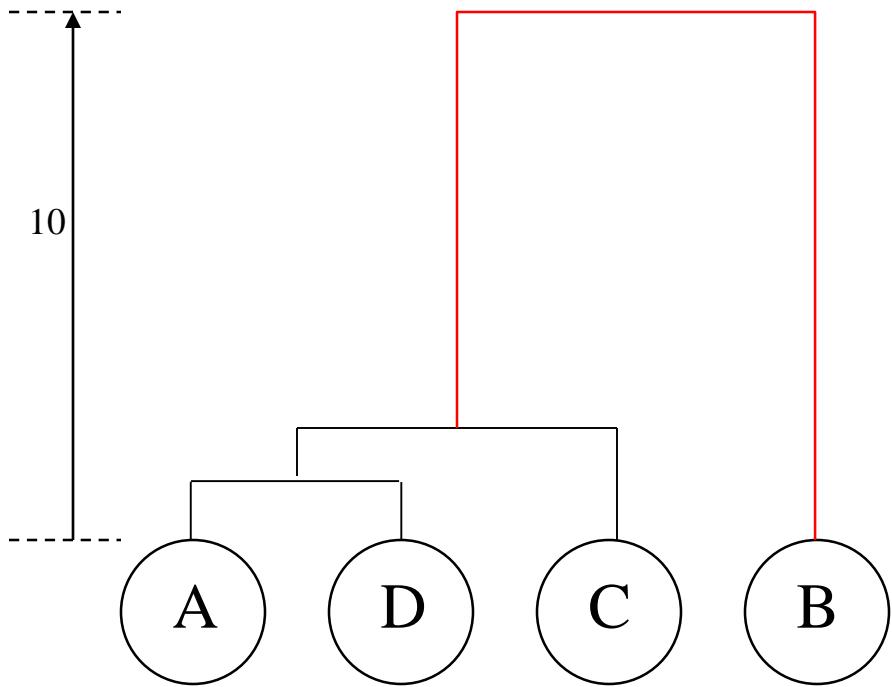
Distance Matrix

Dist	AD C	B		
AD C		10		
B				

Hierarchical Clustering

- HC example

Current Clusters



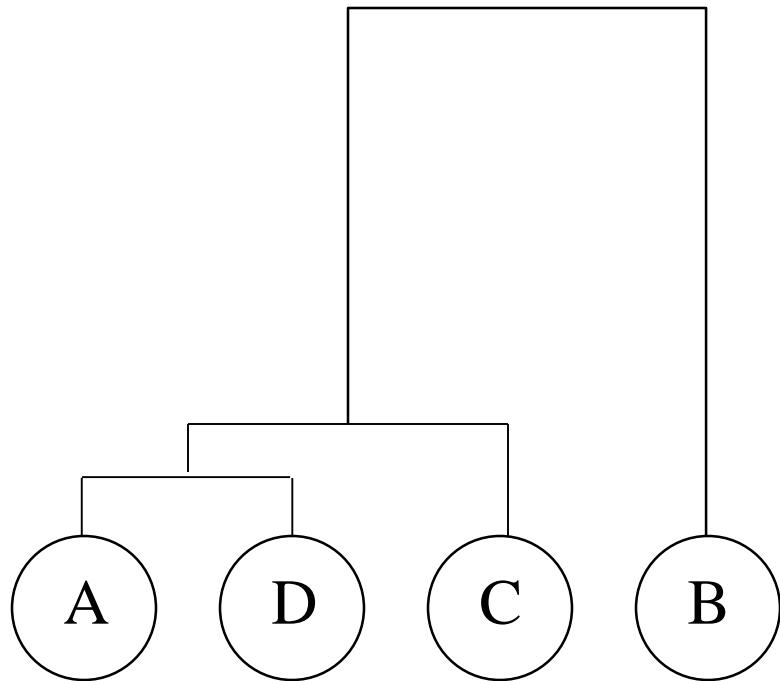
Distance Matrix

Dist	AD C	B		
AD C		10		
B				

Hierarchical Clustering

- HC example

Final Result

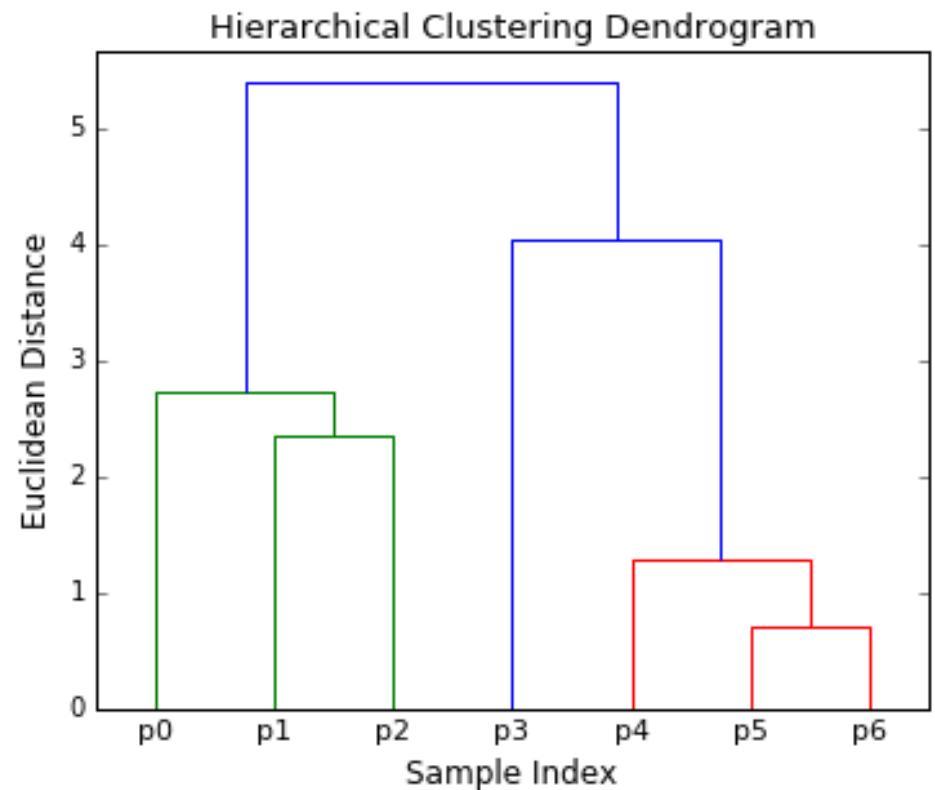
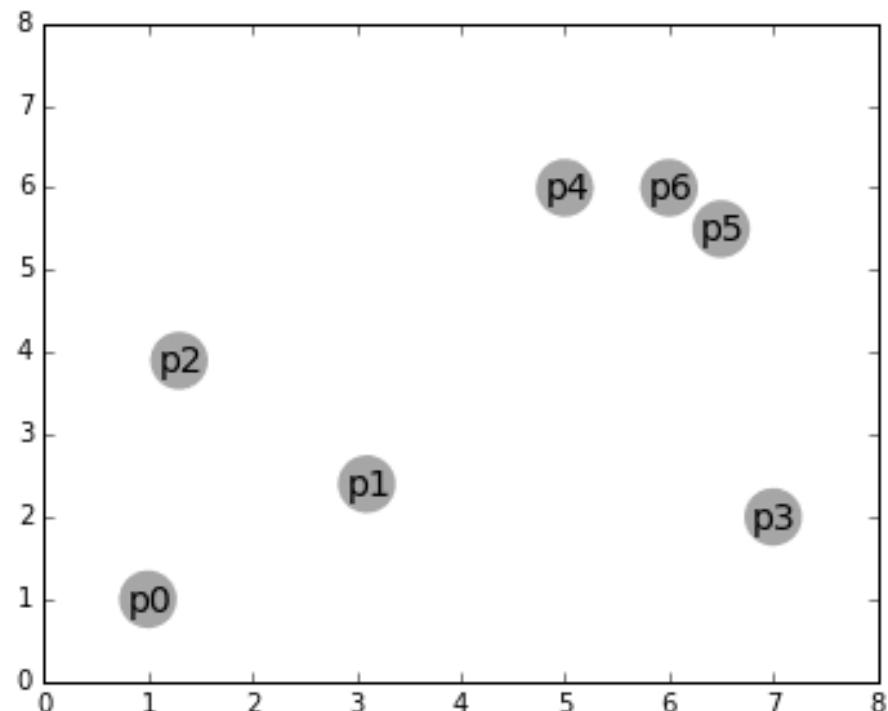


Distance Matrix

Dist	AD	CB			
AD					
CB					

Hierarchical Clustering

- HC example



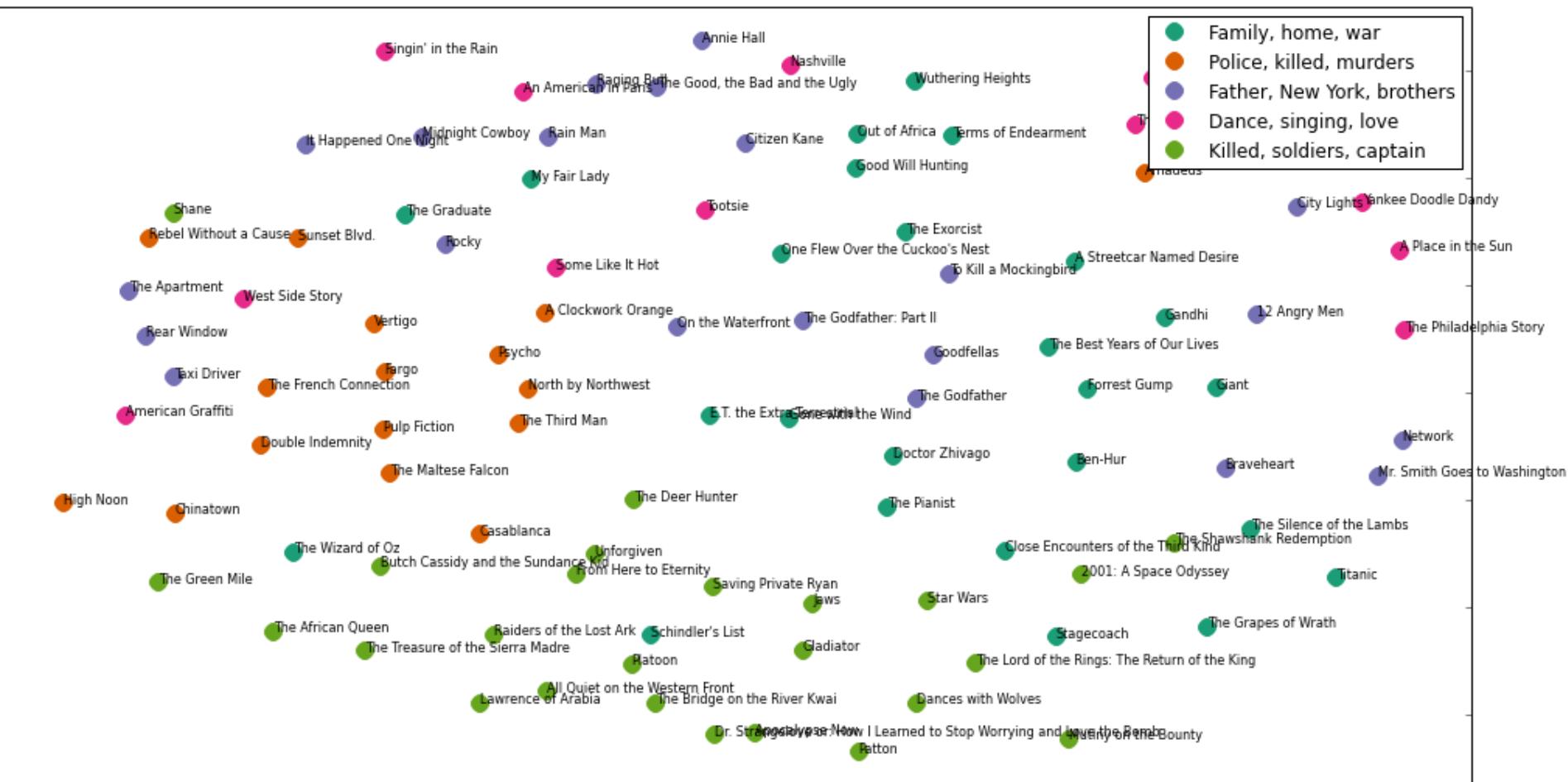
<https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68>

Hierarchical Clustering

- Clustering top 100 film synopses (<http://brandonrose.org/clustering>)
 - ✓ Tokenizing and stemming each synopsis
 - ✓ Transforming the corpus into vector space using [tf-idf](#)
 - ✓ Calculating cosine distance between each document as a measure of similarity
 - ✓ Clustering the documents using the [k-means algorithm](#)
 - ✓ Using [multidimensional scaling](#) to reduce dimensionality within the corpus
 - ✓ Plotting the clustering output using [matplotlib](#) and [mpld3](#)
 - ✓ Conducting a hierarchical clustering on the corpus using [Ward clustering](#)
 - ✓ Plotting a Ward dendrogram
 - ✓ Topic modeling using [Latent Dirichlet Allocation \(LDA\)](#)

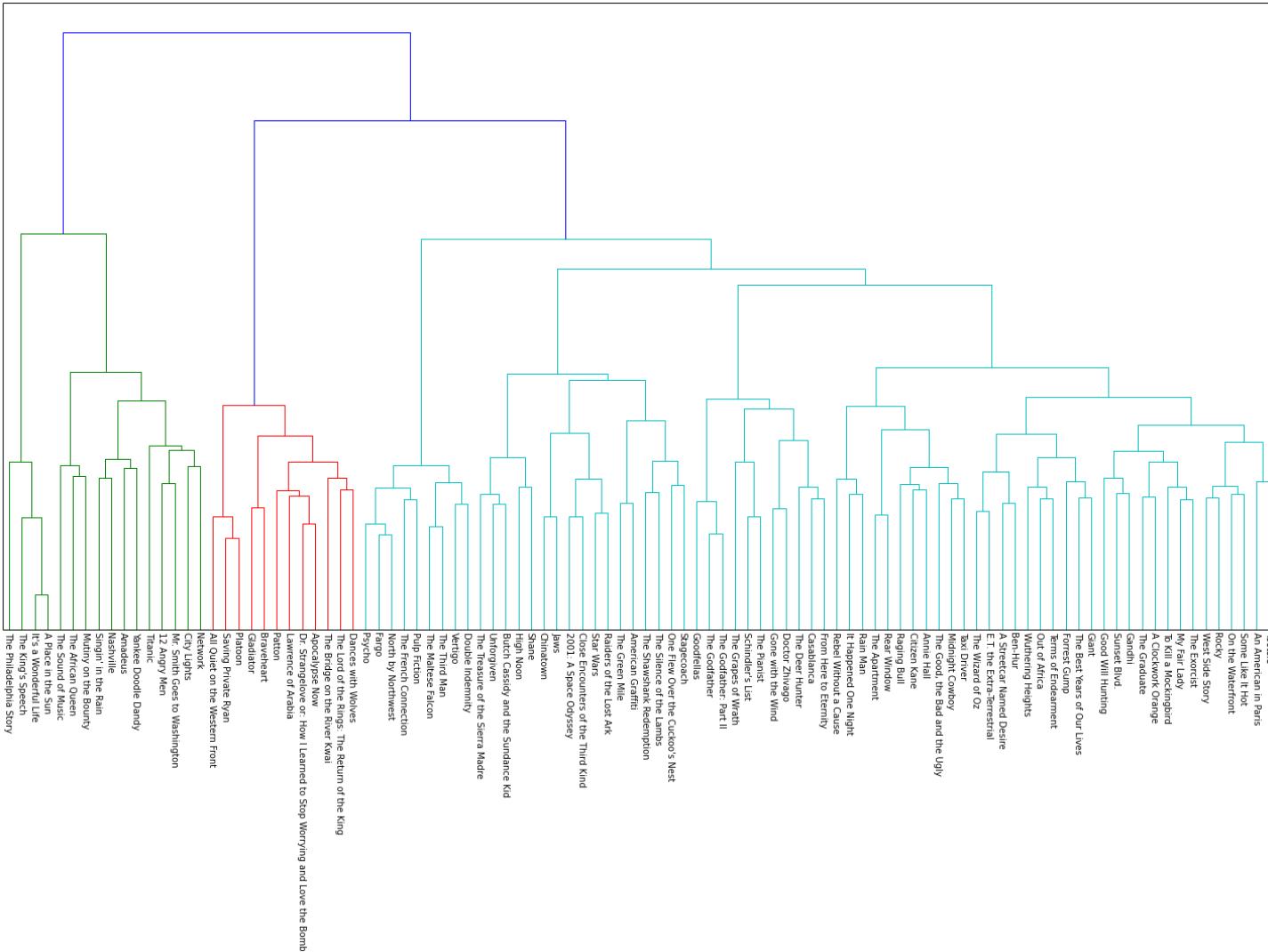
Hierarchical Clustering

- MDS result



Hierarchical Clustering

- Hierarchical clustering



AGENDA

01 Document Similarity

02 Clustering Overview

03 Partition-based Clustering

04 Hierarchical Clustering

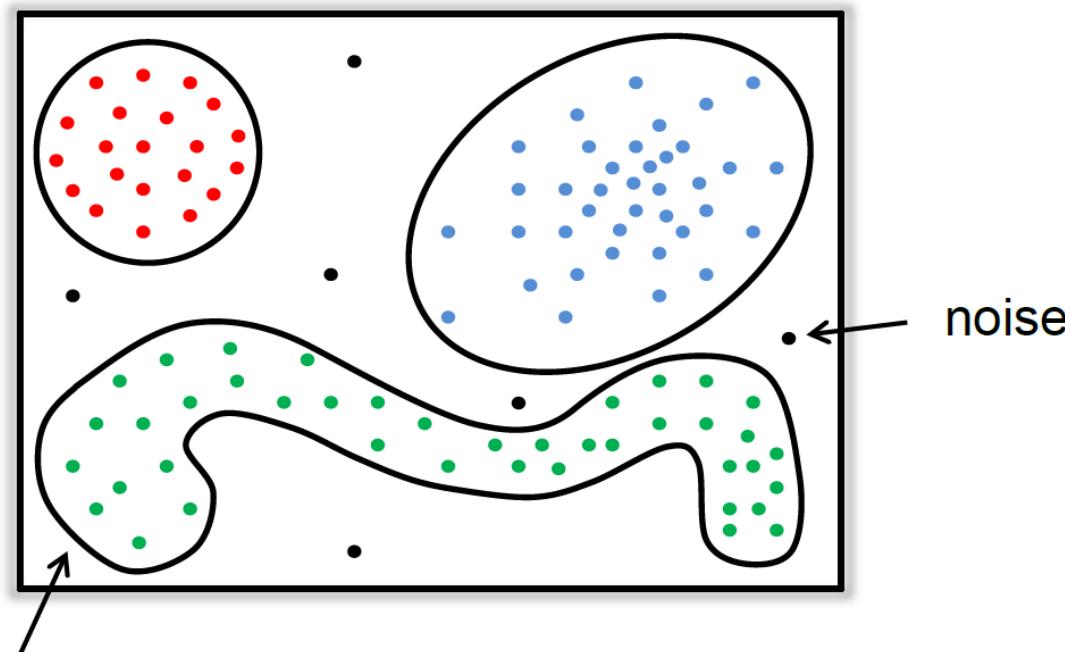
05 Density-based Clustering

06 Self-Organizing Map

Density-based Clustering

Ester et al. (1996)

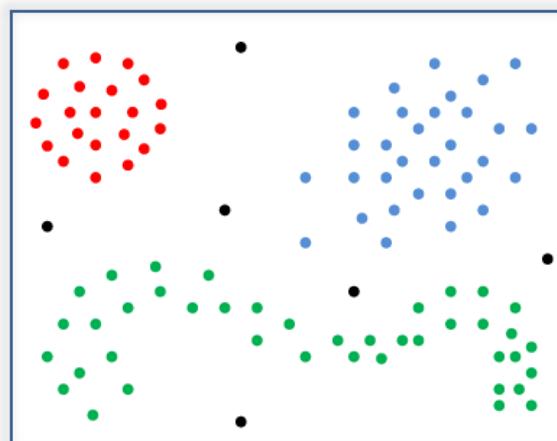
- Density-based clustering
 - ✓ Conduct a clustering by considering the density of data points
 - Can find an arbitrary shape of cluster
 - Can remove noise from clustering result



arbitrarily shaped clusters

Density-based Clustering

- DBSCAN
 - ✓ Most popular density-based clustering algorithm
- Idea
 - ✓ Clusters are the collections of data points with high density
 - ✓ Density around a noise point is very low
- Purpose
 - ✓ Quantify the features of clusters and noise points to find a set of valid clusters



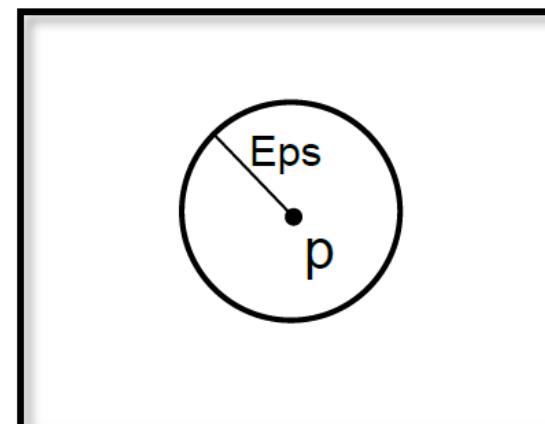
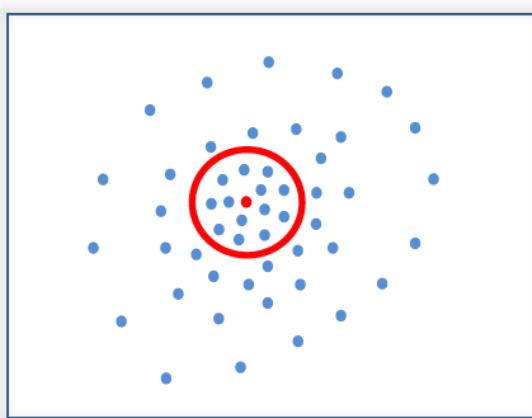
Density-based Clustering

- DBSCAN

- ✓ **Definition 1: ϵ -neighborhood of a point**

- The ϵ -neighborhood of a point, denoted by $N_\epsilon(p)$, is defined by

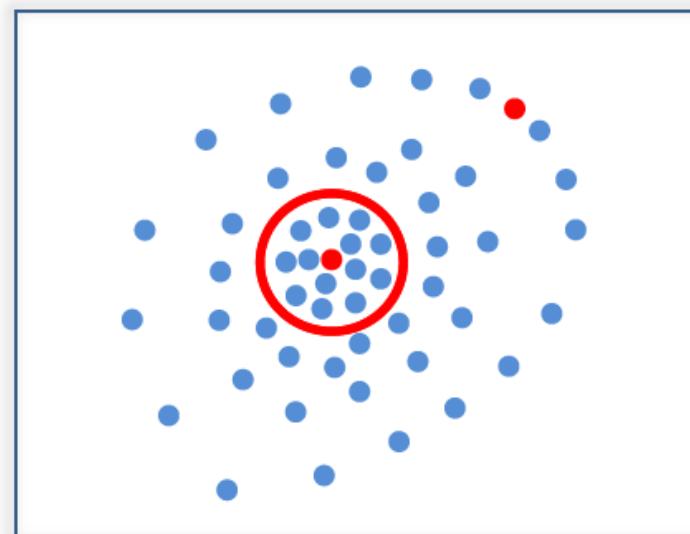
$$N_\epsilon(p) = \{q \in D \mid dist(p, q) \leq \epsilon\}$$



- ✓ **Naïve Approach:** require for each point in a cluster that there are at least a minimum number (**MinPts**) of points in an ϵ -neighborhood of that point

Density-based Clustering

- DBSCAN
 - ✓ Problem of Naïve Approach
 - There are two kinds of points in a cluster
 - Points inside of the cluster (core points)
 - Points on the border of the cluster (border points)
 - An ϵ -neighborhood of a border point contains significantly less points than an ϵ -neighborhood of a core point

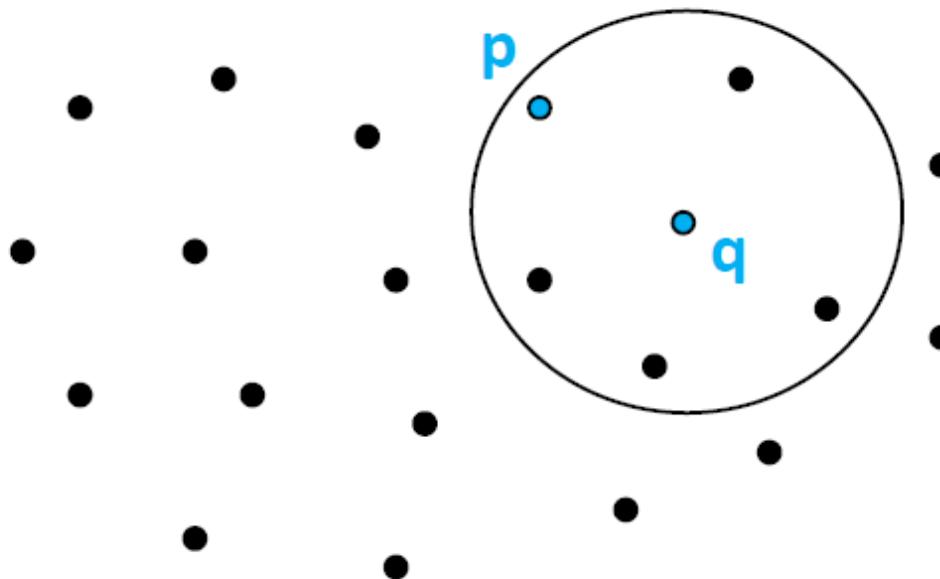


Density-based Clustering

- DBSCAN

- ✓ Better idea

- For every point p in a cluster C , there is a point q in C so that p is inside of the ϵ -neighborhood of q (Border points are connected to core points)
 - $N_\epsilon(q)$ contains at least MinPts points (Core points = high density)



Density-based Clustering

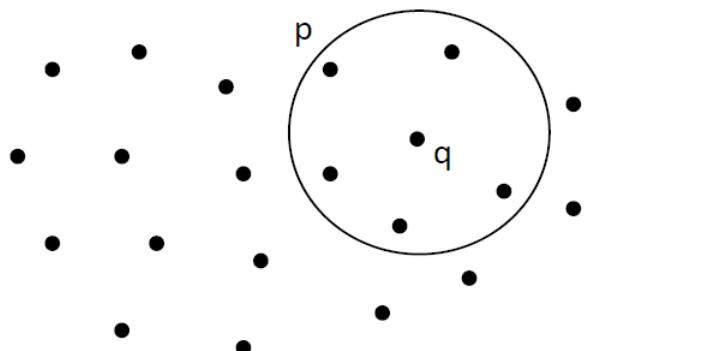
- DBSCAN

✓ **Definition 2: directly density-reachable**

- A point p is directly density-reachable from a point q with regard to the parameters ϵ and MinPts , if

1) $p \in N_\epsilon(q)$ (*reachability*)

2) $|N_\epsilon(q)| \geq \text{MinPts}$ (*core point condition*)



$$\text{MinPts} = 5$$

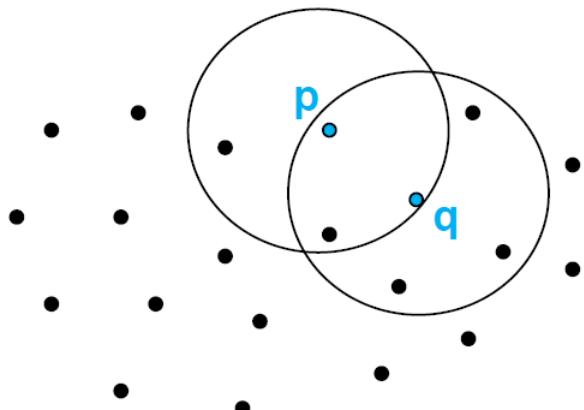
$$|N_{\epsilon}(q)| = 6 \geq 5 = \text{MinPts} \text{ (core point condition)}$$

Density-based Clustering

- DBSCAN

- ✓ Property

- Directly density-reachable is symmetric for pairs of core points
 - It is not symmetric if one core point and one border point are involved



Parameter: MinPts = 5

p directly density reachable from q

$$p \in N_{Eps}(q)$$

$|N_{Eps}(q)| = 6 \geq 5 = \text{MinPts}$ (core point condition)

q not directly density reachable from p

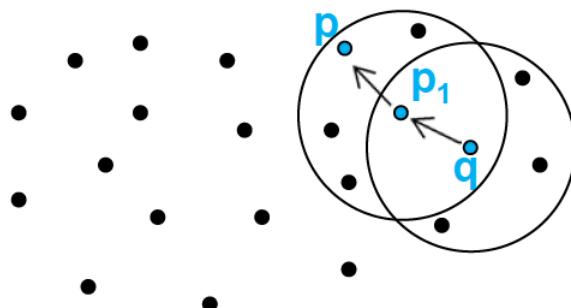
$|N_{Eps}(p)| = 4 < 5 = \text{MinPts}$ (core point condition)

Density-based Clustering

- DBSCAN

- ✓ **Definition 3: density-reachable**

- A point p is density-reachable from a point q with regard to the parameters ϵ and MinPts, if there is a chain of points p_1, p_2, \dots, p_s with $p_1 = q$ and $p_s = p$ such that p_{i+1} is directly density-reachable from p_i for all $1 < i < s-1$



$$\text{MinPts} = 5$$

$$|N_{\epsilon ps}(q)| = 5 = \text{MinPts} \quad (\text{core point condition})$$

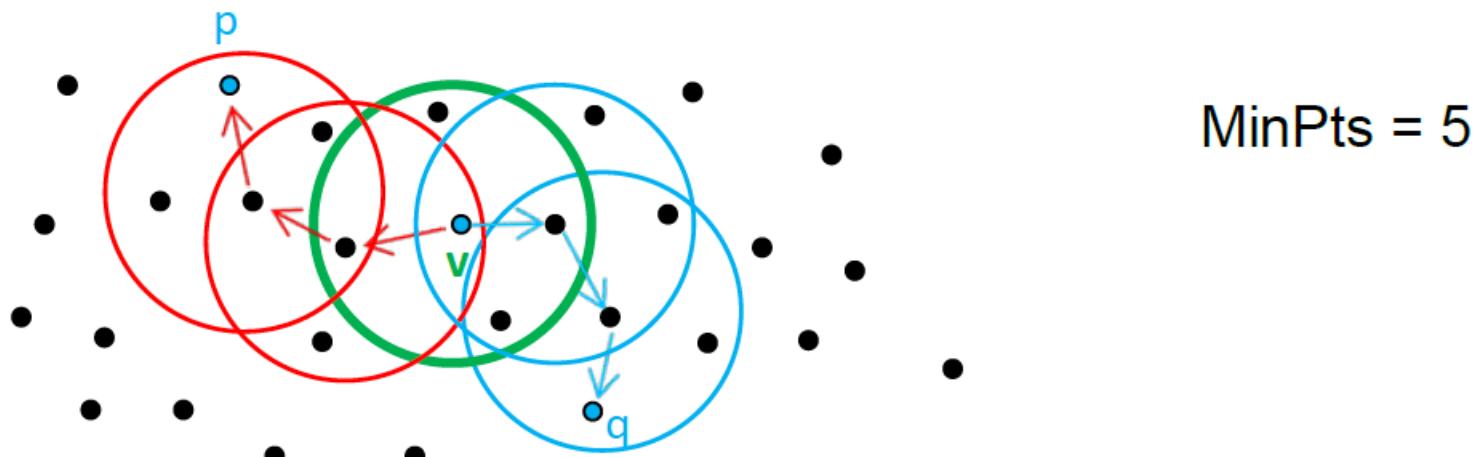
$$|N_{\epsilon ps}(p_1)| = 6 \geq 5 = \text{MinPts} \quad (\text{core point condition})$$

Density-based Clustering

- DBSCAN

- ✓ **Definition 4: density-connected**

- A point p is density-connected to a point q with regard to the parameters ε and MinPts , if there is a point v such that both p and q are density-reachable from v

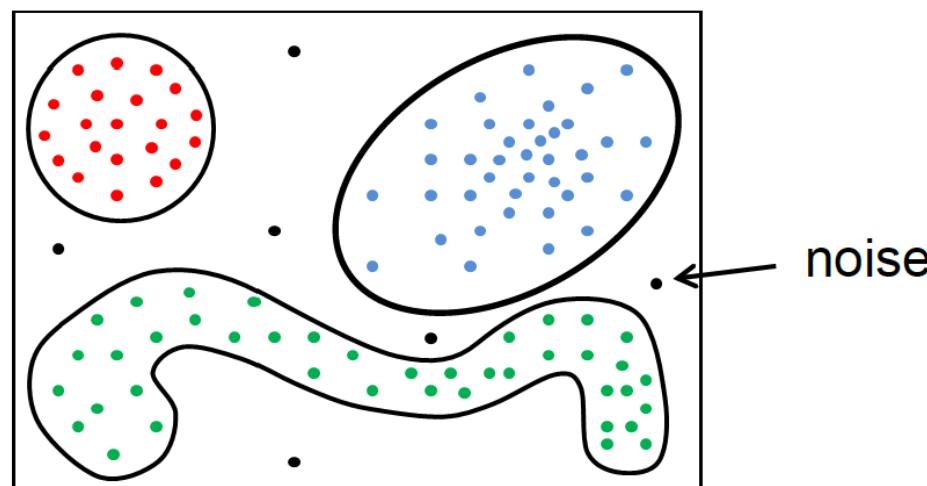


Density-based Clustering

- DBSCAN

- ✓ **Definition 5: Cluster**

- A cluster with regard to the parameters ε and MinPts is a non-empty subset C of the database D with
 - (1) For all $p, q \in D$: If $p \in C$ and q is density-reachable from p with regard to the parameters ε and MinPts, then $q \in C$ (**Maximality**)
 - (2) For all $p, q \in C$: The point p is density-connected to q with regard to the parameters ε and MinPts (**Connectivity**)

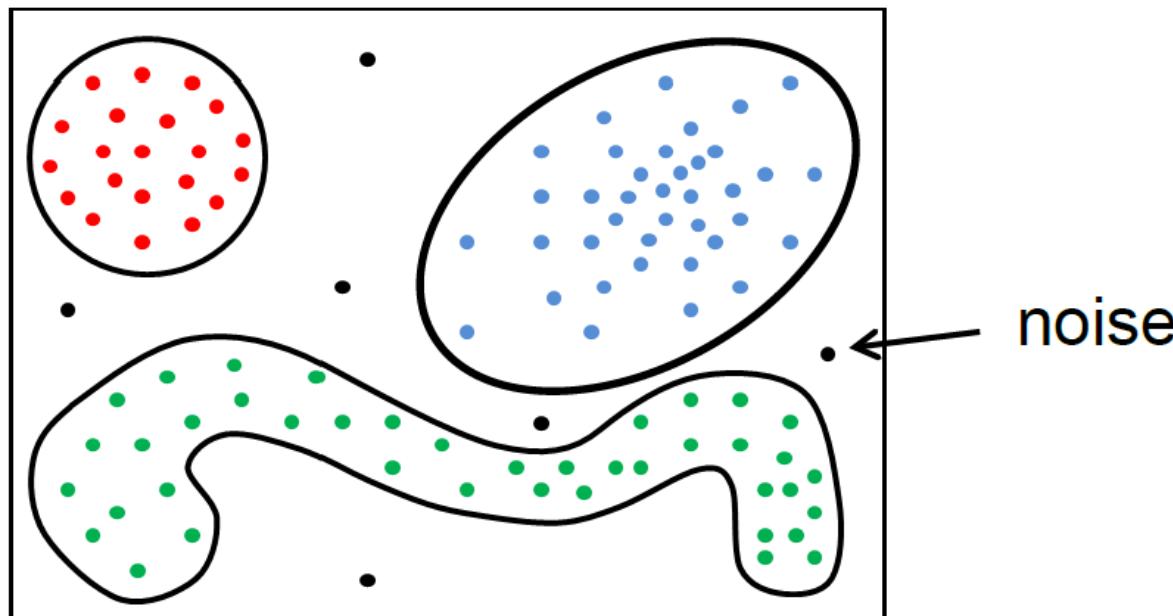


Density-based Clustering

- DBSCAN

- ✓ **Definition 6: Noise**

- Let C_1, \dots, C_k be the clusters of the database D with regard to the parameters ε and MinPts
 - The set of points in the database D not belonging to any cluster C_1, \dots, C_k is called noise

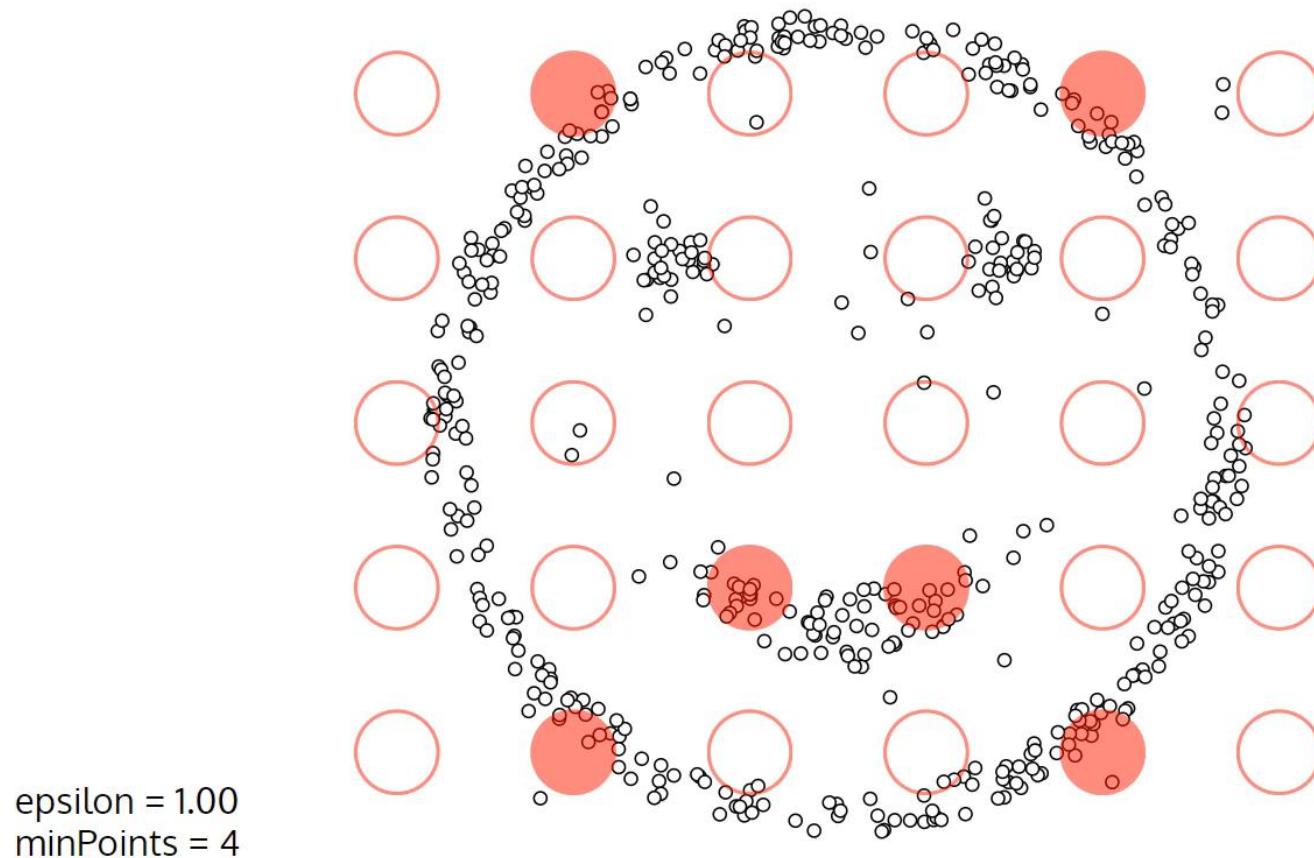


Density-based Clustering

- DBSCAN:Algorithm
 - ✓ Input: N objects to be clustered and global parameter, ε and MinPts
 - ✓ Output: Cluster of objects
- Algorithm
 - ✓ Arbitrary select a point p
 - ✓ Retrieve all points density-reachable from p w.r.t. ε and MinPts
 - ✓ If p is a core points, a cluster is formed
 - ✓ If p is a border point, no points are density reachable from p and DBSCAN visits the next point of the database
 - ✓ Continue the process until all of the points have been processed

Density-based Clustering

- DBSCAN example



Restart

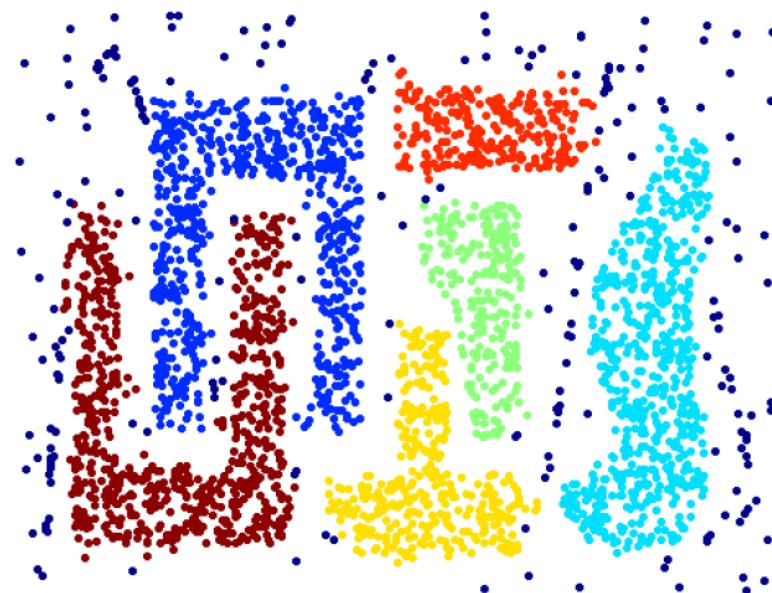
GO!

Density-based Clustering

- DBSCAN example



Original Points



Clusters

AGENDA

01 Document Similarity

02 Clustering Overview

03 Partition-based Clustering

04 Hierarchical Clustering

05 Density-based Clustering

06 Self-Organizing Map

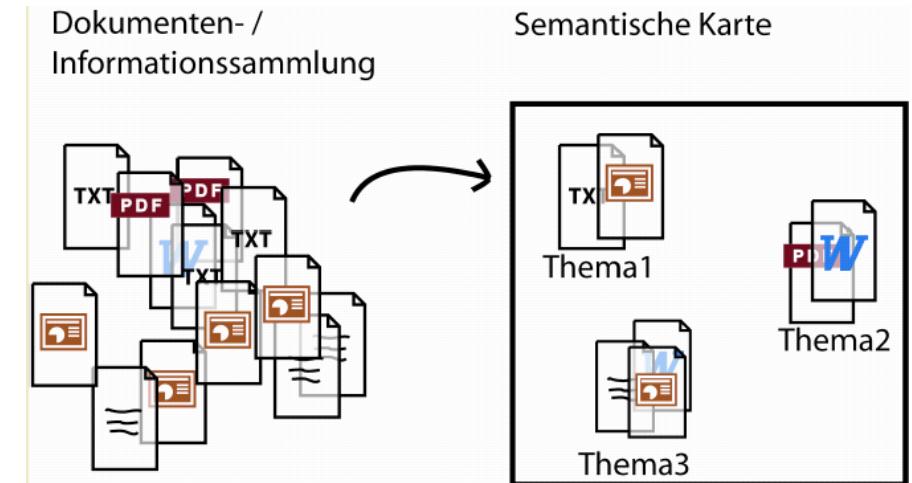
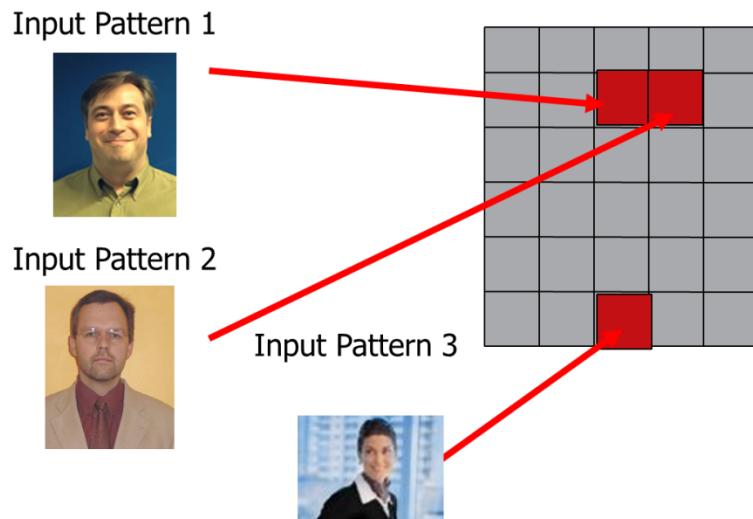
Self-Organizing Map (SOM)

- Self-Organizing Map (SOM)

- ✓ A way to represent higher dimensional data in an 2-D or 3-D grid such that similar records are grouped together

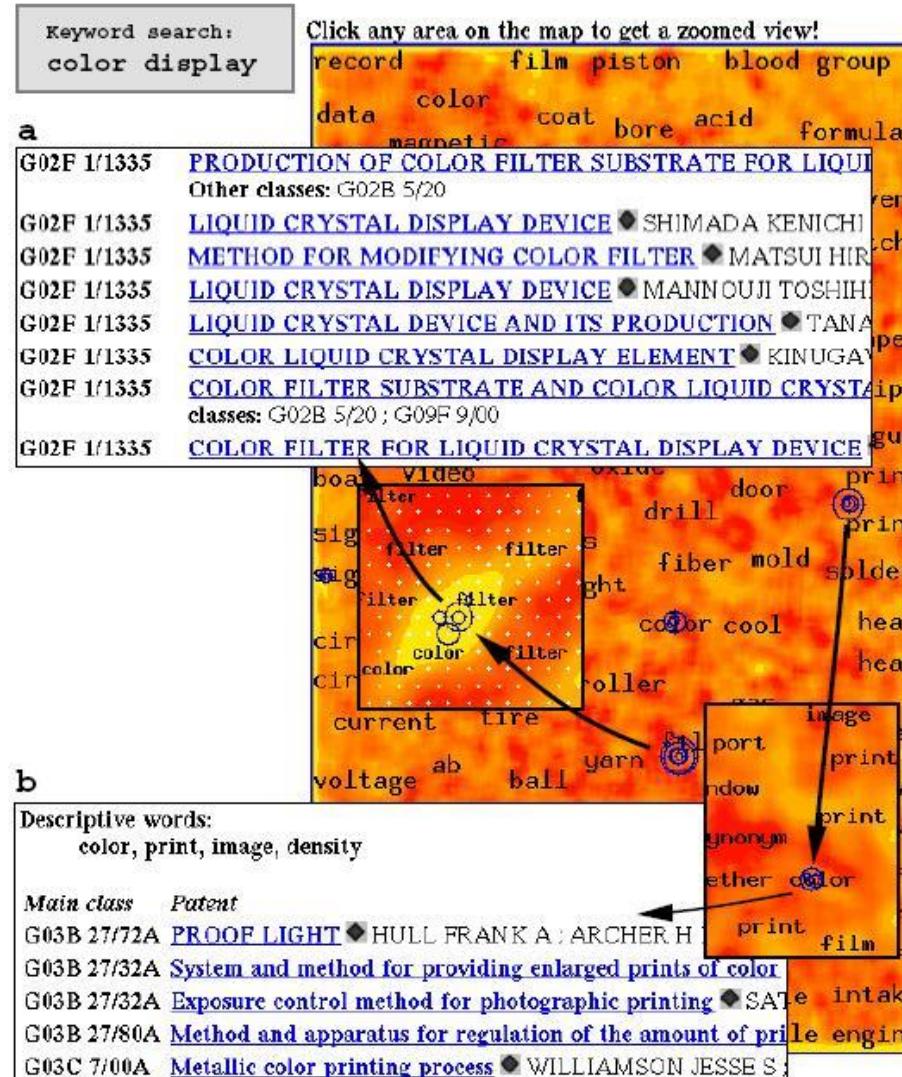
- Idea first introduced by C. von der Malsburg (1973), developed and refined by T. Kohonen (1982)
- Neural network algorithm using unsupervised competitive learning

- ✓ Geometric relationships between image points indicate similarity



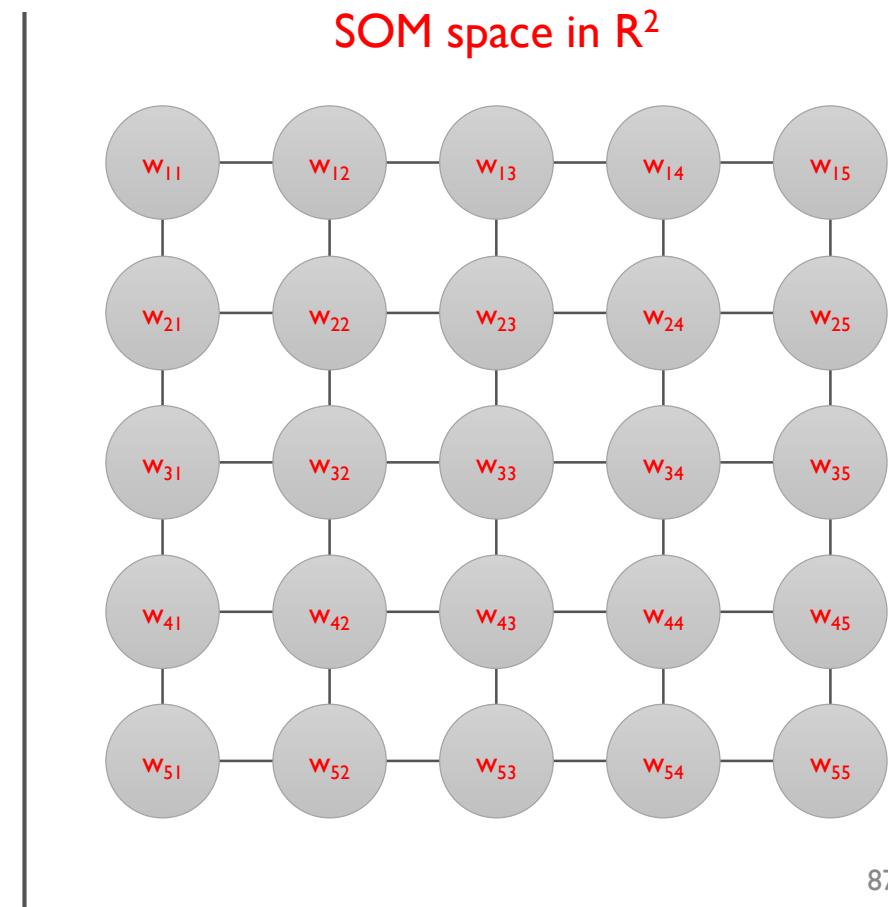
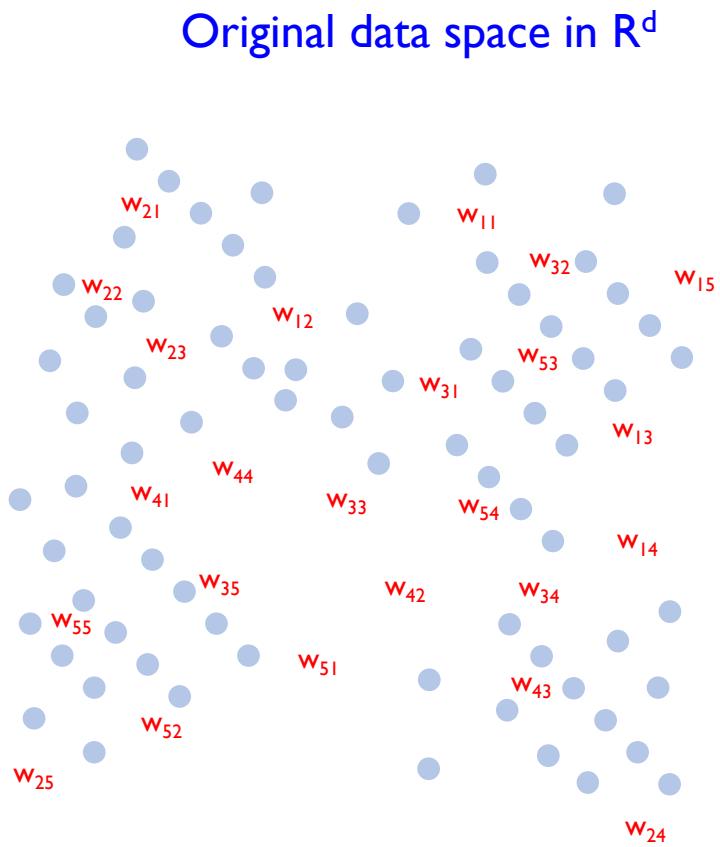
Self-Organizing Map (SOM)

- Application: Document Organization



Self-Organizing Map (SOM)

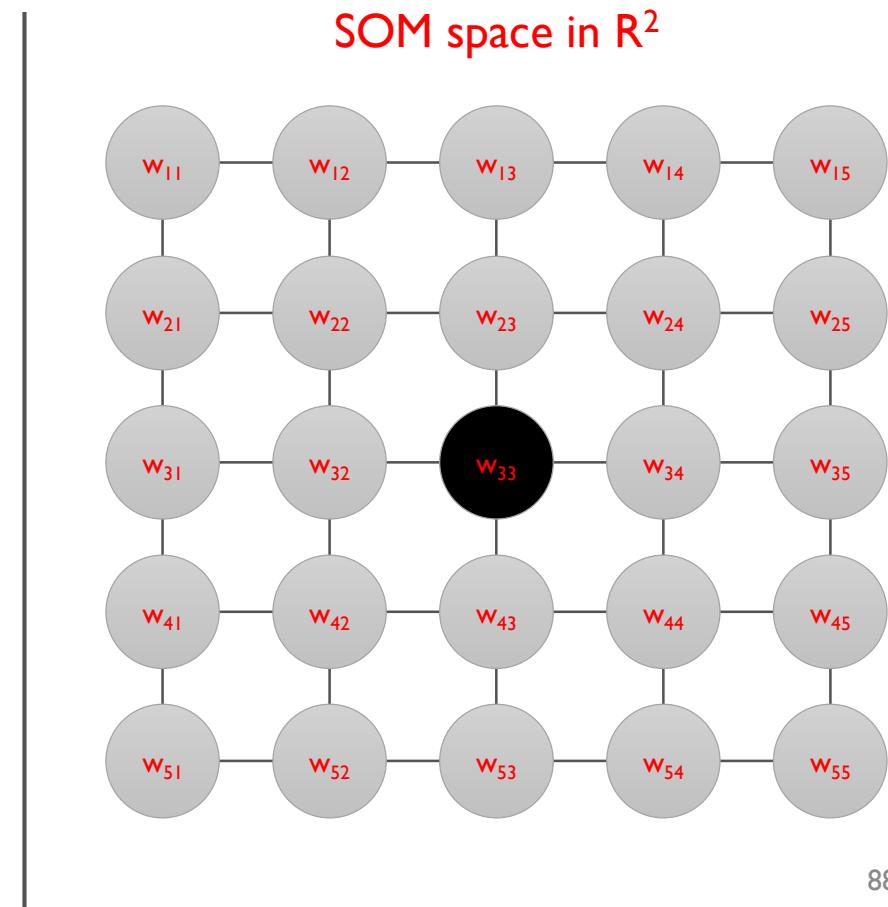
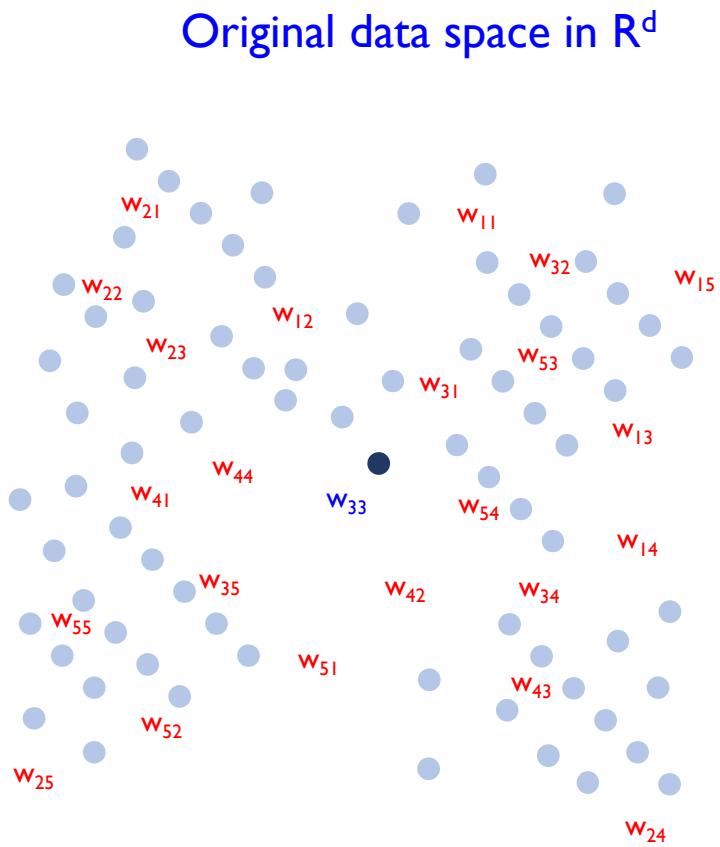
- SOM: Architecture
 - ✓ Every node (lattice of neurons) is connected to a single point in the original data space.



Self-Organizing Map (SOM)

- SOM: Architecture

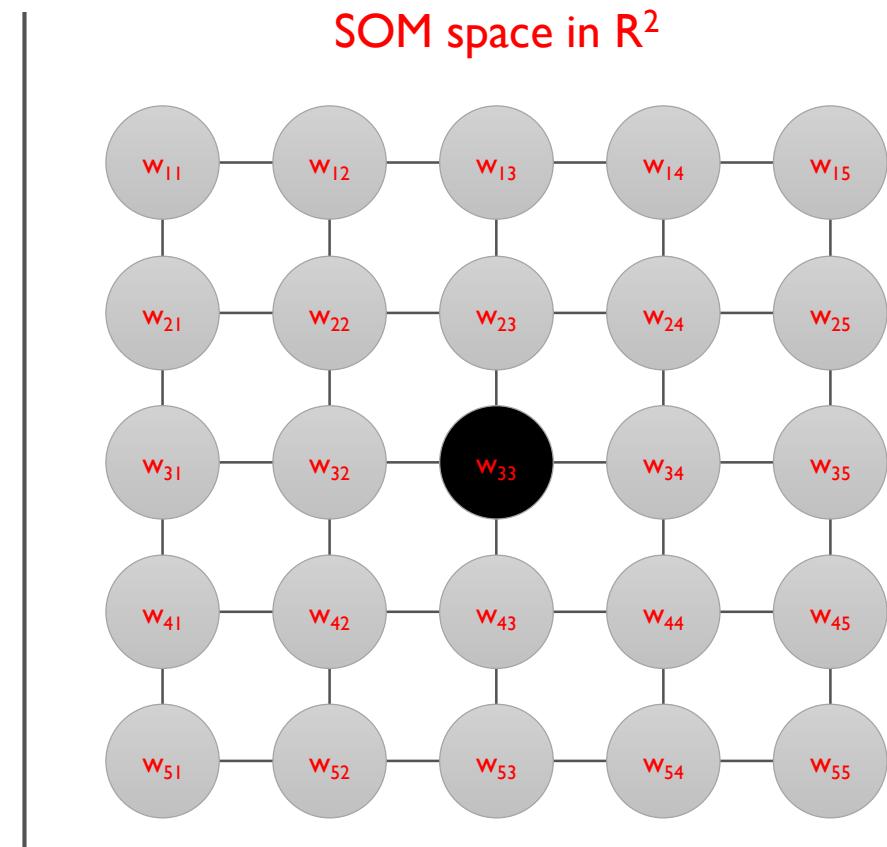
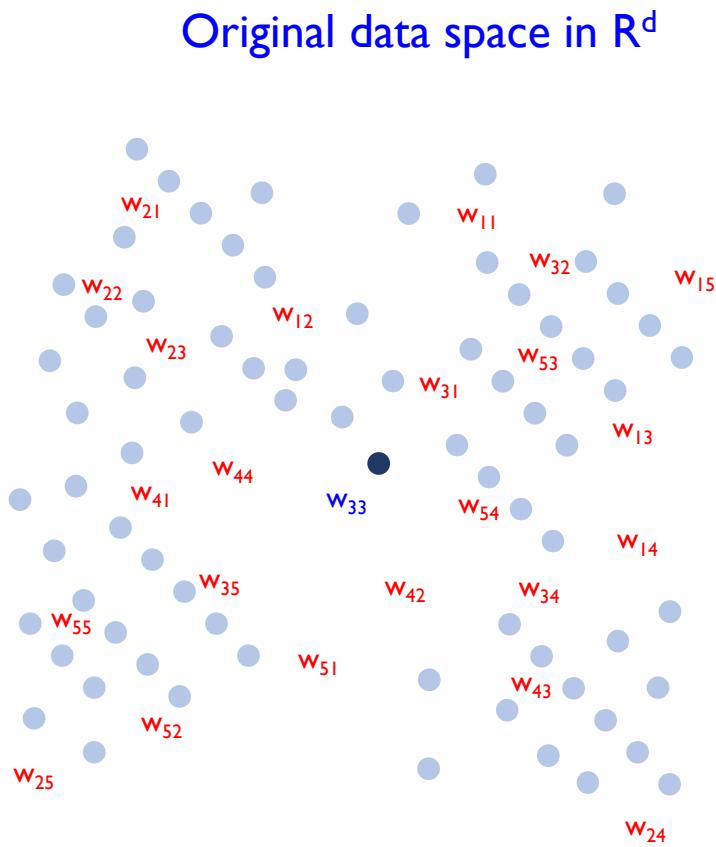
- ✓ Responses from the input is compared and ‘winning’ node is selected



Self-Organizing Map (SOM)

- SOM: Architecture

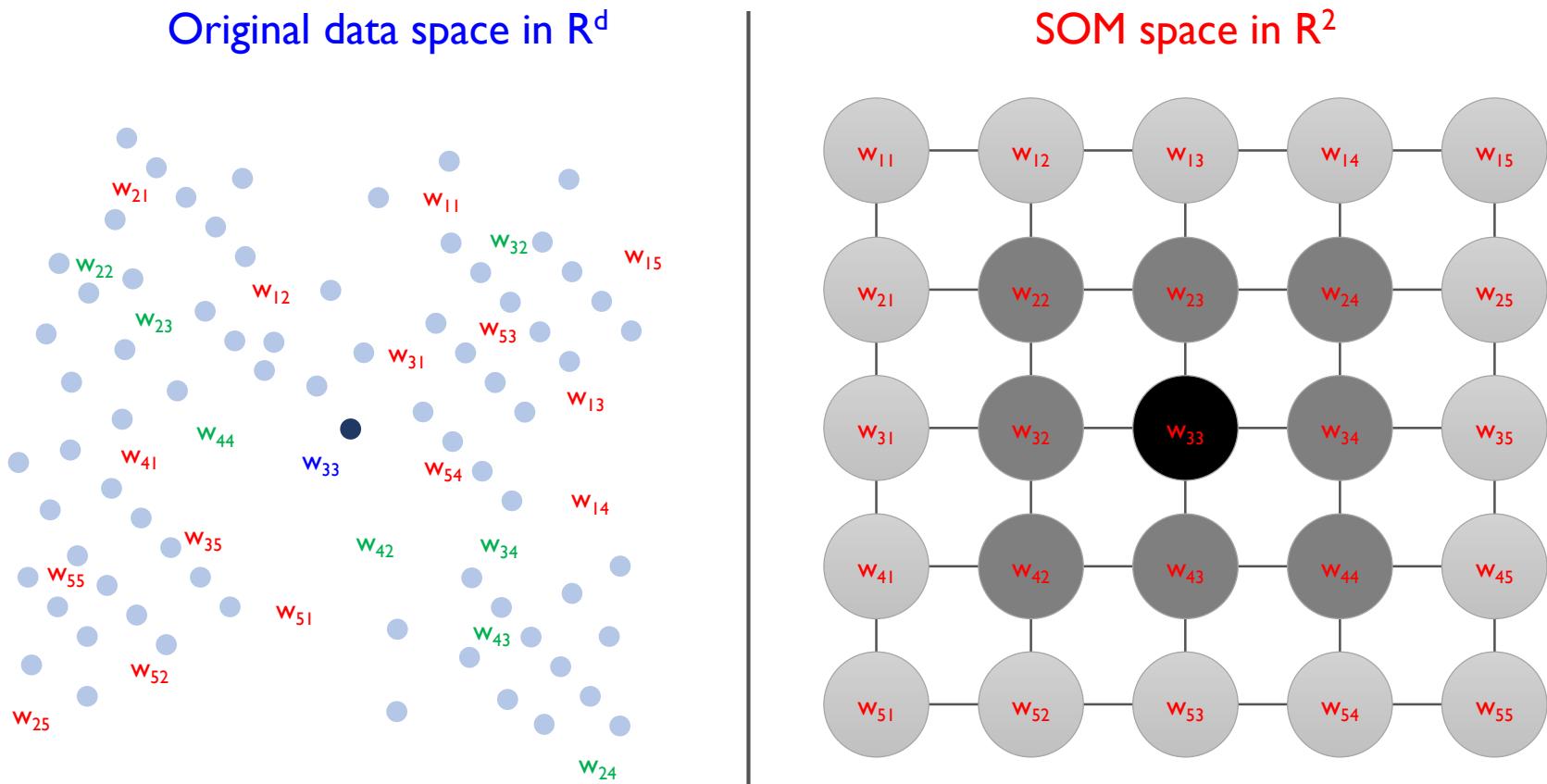
- ✓ Responses from the input is compared and ‘winning’ node is selected
- ✓ Adaptive process changes weights to more closely resemble inputs



Self-Organizing Map (SOM)

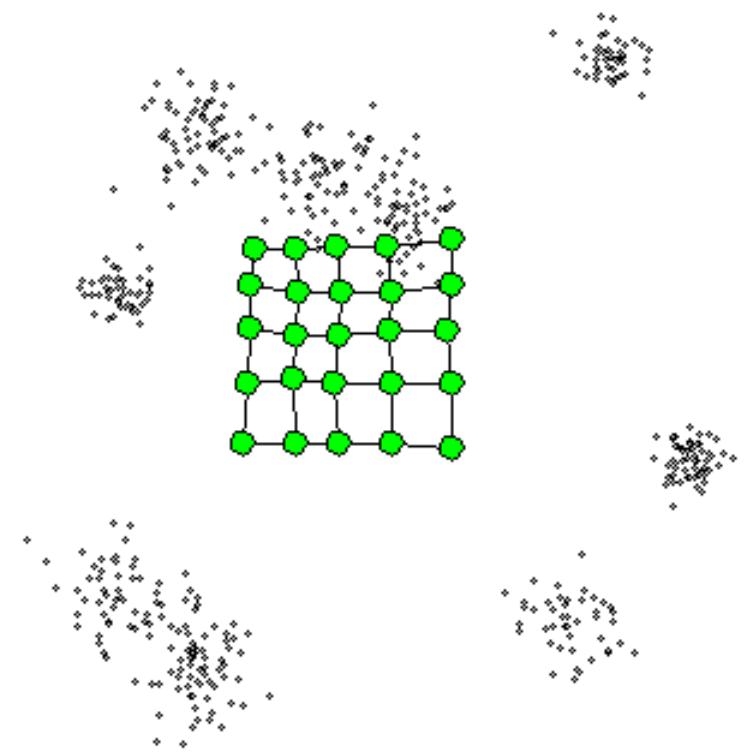
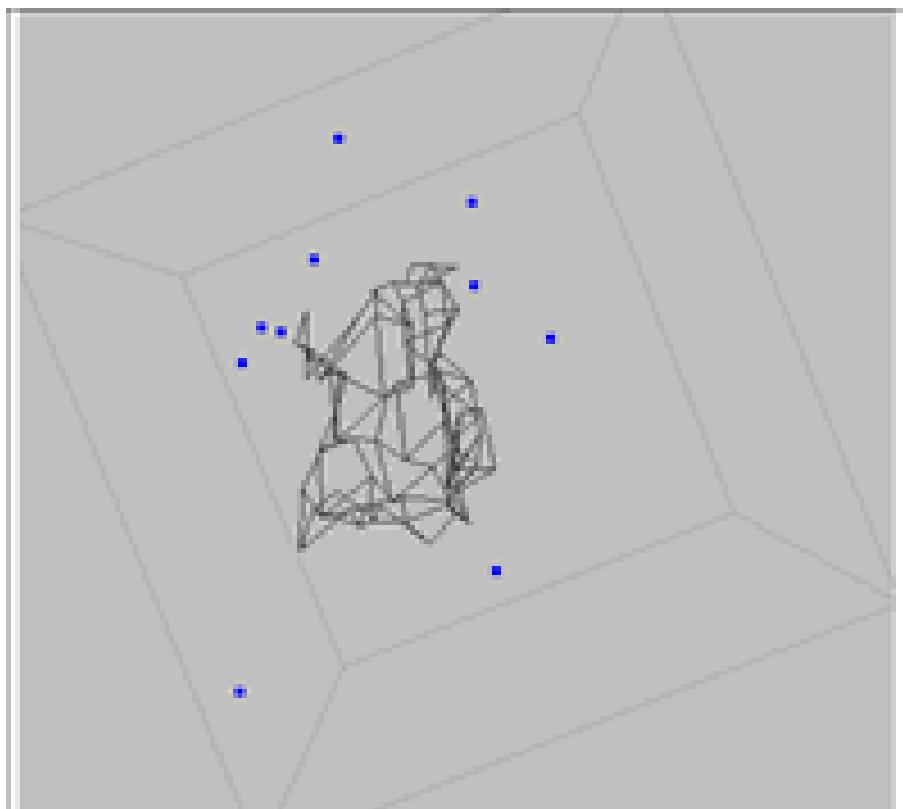
- SOM: Architecture

- ✓ Selected node is activated together with ‘neighborhood’ node
- ✓ Adaptive process changes weights to more closely resemble inputs



Self-Organizing Map (SOM)

- SOM: Architecture



Self-Organizing Map (SOM)

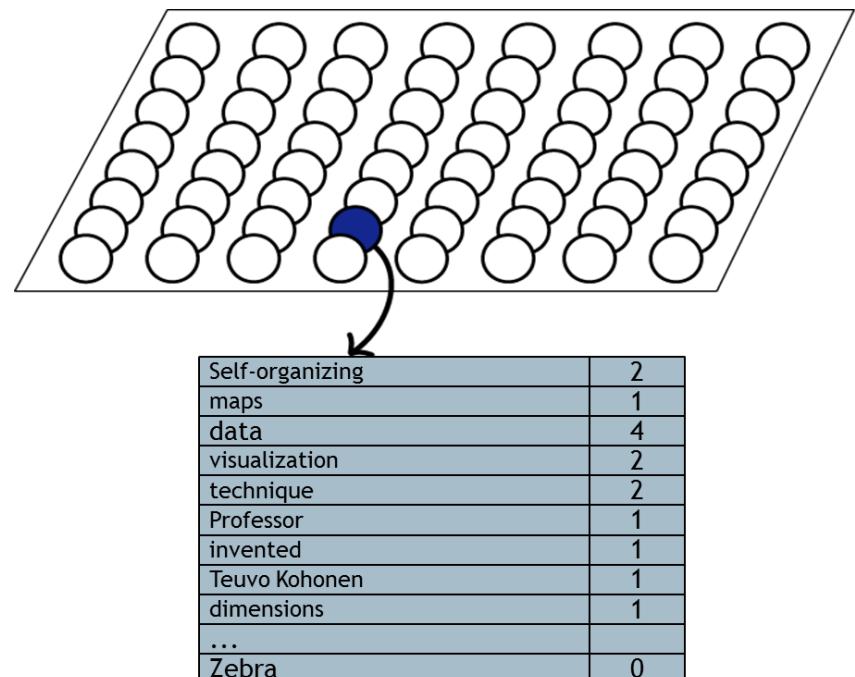
- SOM: Learning
 1. Initialize each node's weights (usually at random)
 2. Choose a random vector from training data and present it to the SOM
 3. Every node is examined to find the **Best Machine Unit (BMU)**
 4. The radius of the neighborhood around the BMU is calculated, the size of the neighborhood decreases with each iteration
 5. Each node in the BMU's neighborhood has its weights adjusted to become more like the BMU. Nodes closest to the BMU are altered more than the nodes furthest away in the neighborhood
 6. Repeat from step 2 for enough iterations for convergence

Self-Organizing Map (SOM)

- SOM: Learning
 2. Choose a random vector from training data and present it to the SOM
 3. Every node is examined to find the **Best Machine Unit (BMU)**

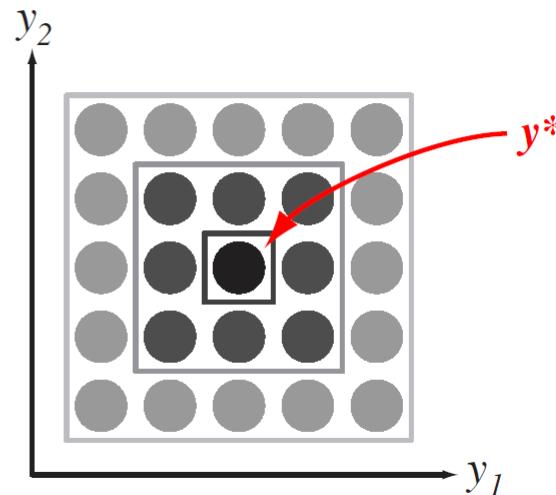
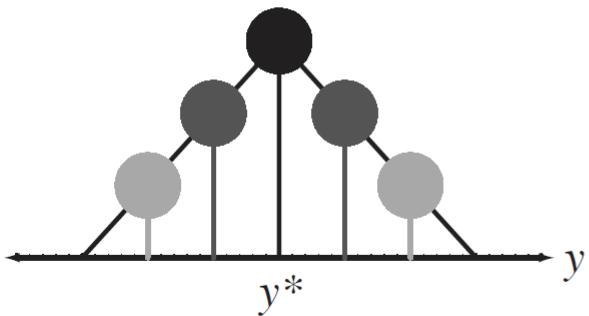
A Text Example:

Self-organizing maps (SOMs) are a **data** visualization **technique** invented by Professor Teuvo Kohonen which reduce the dimensions of **data** through the use of **self-organizing** neural networks. The problem that data visualization attempts to solve is that humans simply cannot visualize high dimensional **data** as is so **technique** are created to help us understand this high dimensional **data**.



Self-Organizing Map (SOM)

- SOM: Learning
4. The radius of the neighborhood around the BMU is calculated, the size of the neighborhood decreases with each iteration



Self-Organizing Map (SOM)

- SOM: Learning
5. Each node in the BMU's neighborhood has its weights adjusted to become more like the BMU. Nodes closest to the BMU are altered more than the nodes furthest away in the neighborhood

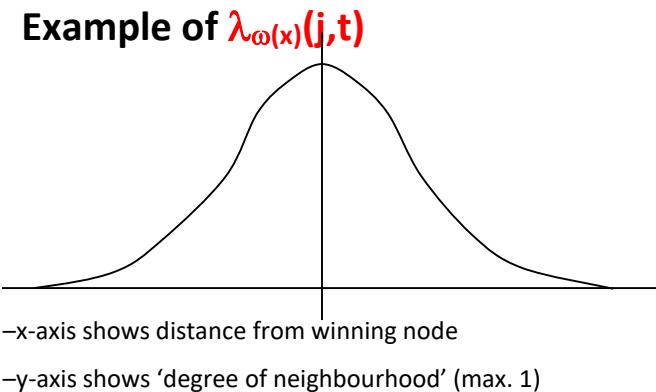
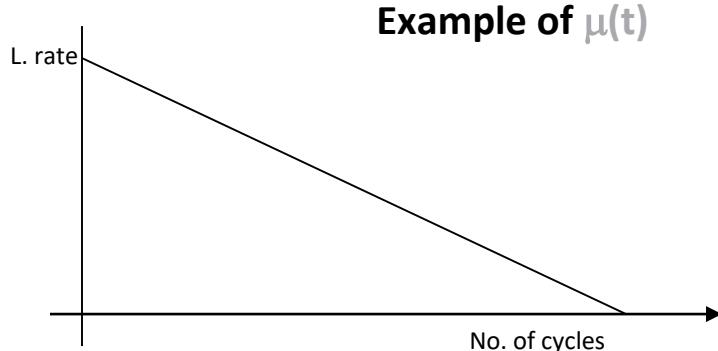
<SOM Weight Update Equation>

$$w_j(t+1) = w_j(t) + \mu(t) \lambda_{\omega(x)}(j,t) [x - w_j(t)]$$

“The weights of every node are updated at each cycle by adding

Current learning rate \times Degree of neighbourhood with respect to winner \times Difference between current weights and input vector

to the current weights”



Self-Organizing Map (SOM)

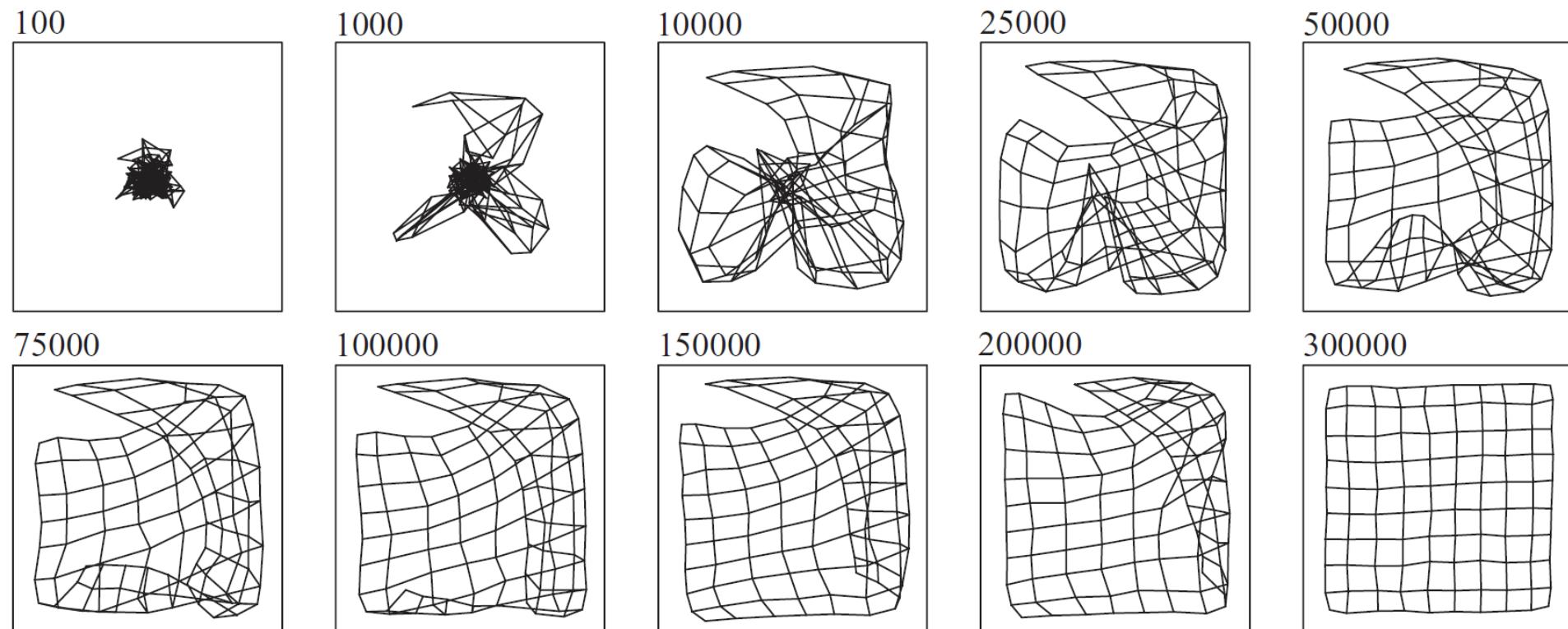
Components of Self-Organization

1. Initialization
2. Competition
3. Cooperation
4. Adaptation

Self-Organizing Map (SOM)

- SOM: Mapping through iterations

✓ Happily converges to...

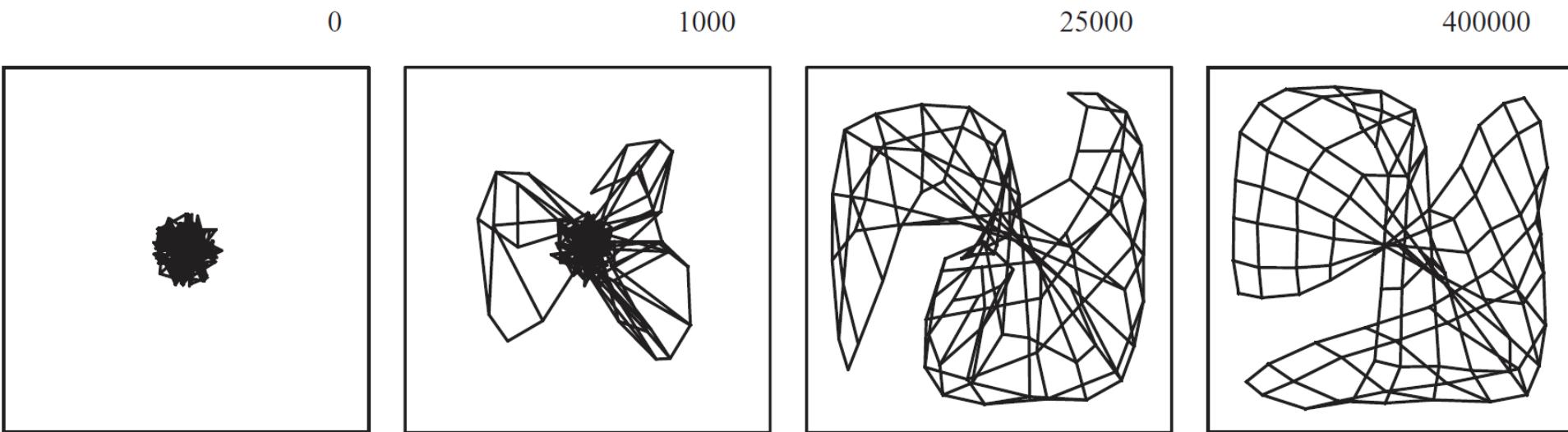


Self-Organizing Map (SOM)

- Training SOM: Example

Self-Organizing Map (SOM)

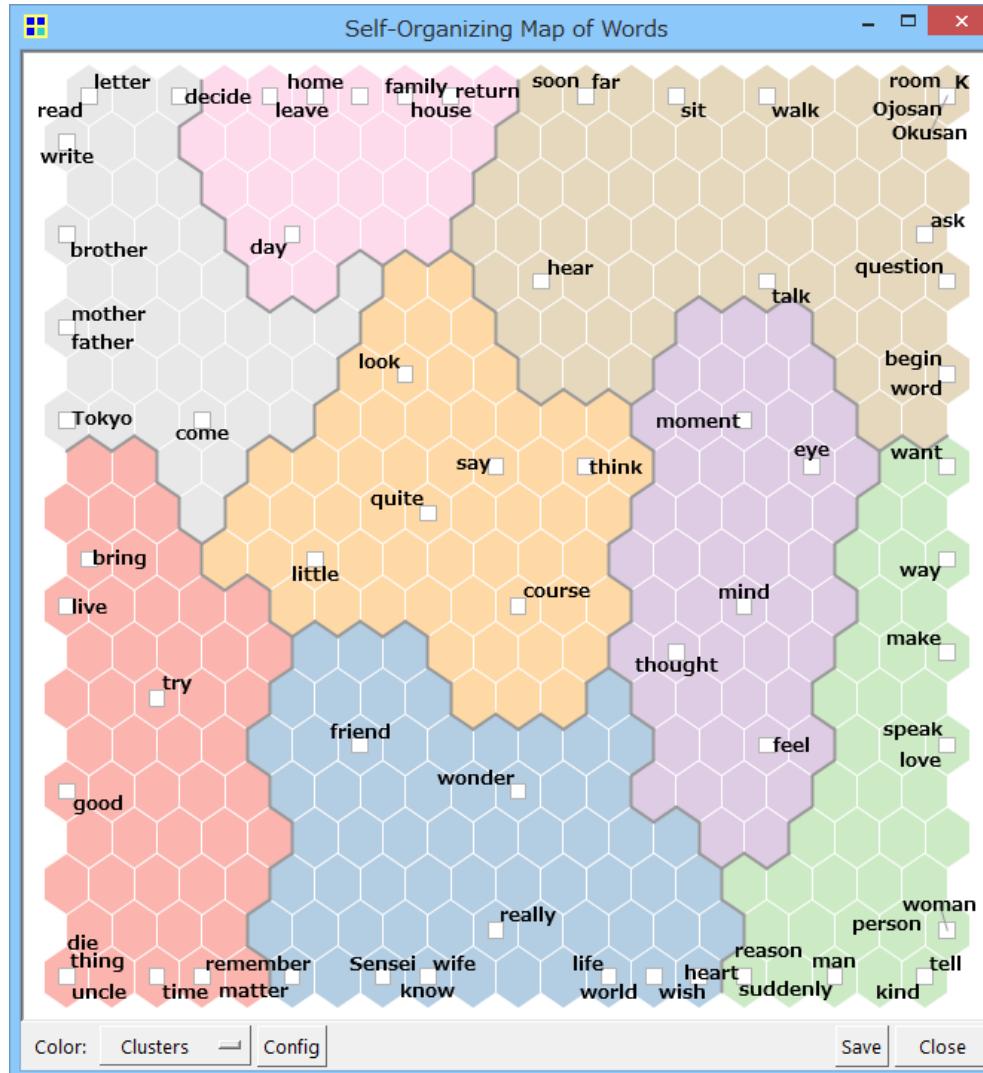
- SOM: Mapping through iterations
 - ✓ Unfortunately it happens sometimes...



✓ Initial weights should be regenerated!

Case Studies

- Self-organizing map of words





ANY
questions?

Appendix A: Similarity/Distance Metrics for Binary Data

Choi et al. (2010)

$S_{JACCARD} = \frac{a}{a+b+c}$	(1)	$D_{VARI} = \frac{(b+c)}{4(a+b+c+d)}$	(23)	$S_{FAGERM\text{-}GOWAN} = \frac{a}{\sqrt{(a+b)(a+c)}} - \frac{\max(a+b, a+c)}{2}$	(47)	$S_{ANDERBERG} = \frac{\sigma - \sigma'}{2n}$	(70)
$S_{DICE} = \frac{2a}{2a+b+c}$	(2)	$D_{SIZEDIFFERENCE} = \frac{(b+c)^2}{(a+b+c+d)^2}$	(24)	$S_{FORBES-II} = \frac{na - (a+b)(a+c)}{n \min(a+b, a+c) - (a+b)(a+c)}$	(48)	$S_{BARONI\text{-}URBANI \& BUSER-I} = \frac{\sqrt{ad} + a}{\sqrt{ad} + a + b + c}$	(71)
$S_{CZEKANOWSKI} = \frac{2a}{2a+b+c}$	(3)	$D_{SHAPE DIFFERENCE} = \frac{n(b+c) - (b-c)^2}{(a+b+c+d)^2}$	(25)	$S_{SOKAL\&SNEATH-IV} = \frac{\frac{a}{(a+b)} + \frac{a}{(a+c)} + \frac{d}{(b+d)} + \frac{d}{(c+d)}}{4}$	(49)	$S_{BARONI\text{-}URBANI \& BUSER-II} = \frac{\sqrt{ad} + a - (b+c)}{\sqrt{ad} + a + b + c}$	(72)
$S_{3W\text{-}JACCARD} = \frac{3a}{3a+b+c}$	(4)	$D_{PATTERNDIFFERENCE} = \frac{4bc}{(a+b+c+d)^2}$	(26)	$S_{GOWER} = \frac{a+d}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$	(50)	$S_{PEIRCE} = \frac{ab+bc}{ab+2bc+cd}$	(73)
$S_{NEI\&LI} = \frac{2a}{(a+b)+(a+c)}$	(5)	$D_{LANCE\&WILLIAMS} = \frac{b+c}{(2a+b+c)}$	(27)	$S_{PEARSON-I} = \chi^2 \text{ where } \chi^2 = \frac{n(ad-bc)^2}{(a+b)(a+c)(c+d)(b+d)}$	(51)	$S_{EVRAUD} = \frac{n^2(na - (a+b)(a+c))}{(a+b)(a+c)(b+d)(c+d)}$	(74)
$S_{SOKAL\&SNEATH-I} = \frac{a}{a+2b+c}$	(6)	$D_{BRAY\&CURTIS} = \frac{b+c}{(2a+b+c)}$	(28)	$S_{PEARSON-II} = \left(\frac{\chi^2}{n+\chi^2}\right)^{1/2}$	(52)	$S_{TARANTULA} = \frac{a}{\frac{c}{(c+d)}} = \frac{a(c+d)}{c(a+b)}$	(75)
$S_{SOKAL\&SNEATH-II} = \frac{a+d}{a+b+c+d}$	(7)	$D_{HELLINGER} = 2\sqrt{1 - \frac{a}{\sqrt{(a+b)(a+c)}}}$	(29)	$S_{PEARSON-III} = \left(\frac{\rho}{n+\rho}\right)^{1/2} \text{ where } \rho = \frac{ad-bc}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$	(53)	$S_{AMPLE} = \left \frac{a}{\frac{c}{(c+d)}}\right = \left \frac{a(c+d)}{c(a+b)}\right $	(76)
$S_{SOKAL\&SNEATH-III} = \frac{2(a+d)}{2a+b+c+2d}$	(8)	$D_{CHORD} = \sqrt{2\left(1 - \frac{a}{\sqrt{(a+b)(a+c)}}\right)}$	(30)	$S_{PEARSON\&HERON-I} = \frac{ad-bc}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$	(54)		
$S_{ROGER\&TAHOMOTO} = \frac{a+d}{a+2(b+c)+d}$	(9)	$S_{COSINE} = \frac{a}{\sqrt{(a+b)(a+c)^2}}$	(31)	$S_{PEARSON\&HERON-II} = \text{Cos}\left(\frac{\pi\sqrt{bc}}{\sqrt{ad} + \sqrt{bc}}\right)$	(55)		
$S_{FAITH} = \frac{a+0.5d}{a+b+c+d}$	(10)	$S_{GILBERT\&WELLS} = \log a - \log n - \log\left(\frac{a+b}{n}\right) - \log\left(\frac{a+c}{n}\right)$	(32)	$S_{SOKAL\&SNEATH-III} = \frac{a+d}{b+c}$	(56)		
$S_{GOWER\&LEGENDRE} = \frac{a+d}{a+0.5(b+c)+d}$	(11)	$S_{OCHIAI-I} = \frac{a}{\sqrt{(a+b)(a+c)}}$	(33)	$S_{SOKAL\&SNEATH-V} = \frac{ad}{(a+b)(a+c)(b+d)(c+d)^{0.5}}$	(57)		
$S_{INTERSECTION} = a$	(12)	$S_{FORBES-I} = \frac{na}{(a+b)(a+c)}$	(34)	$S_{COLE} = \frac{\sqrt{2}(ad-bc)}{\sqrt{(ad-bc)^2 - (a+b)(a+c)(b+d)(c+d)}}$	(58)		
$S_{INNERPRODUCT} = a+d$	(13)	$S_{FOSSUM} = \frac{n(a-0.5)^2}{(a+b)(a+c)}$	(35)	$S_{STILES} = \log_{10} \frac{n(ad-bc - \frac{n}{2})^2}{(a+b)(a+c)(b+d)(c+d)}$	(59)		
$S_{RUSSELL\&RAO} = \frac{a}{a+b+c+d}$	(14)	$S_{SORGENFREY} = \frac{a^2}{(a+b)(a+c)}$	(36)	$S_{OCHIAI-II} = \frac{ad}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$	(60)		
$D_{HAMMING} = b+c$	(15)	$S_{AMOUNTFORD} = \frac{a}{0.5(ab+ac)+bc}$	(37)	$S_{YULEQ} = \frac{ad-bc}{ad+bc}$	(61)		
$D_{EUCLID} = \sqrt{b+c}$	(16)	$S_{OTSUKA} = \frac{a}{((a+b)(a+c))^{0.5}}$	(38)	$D_{YULEQ} = \frac{2bc}{ad+bc}$	(62)		
$D_{SQUARED-EUCLID} = \sqrt{(b+c)^2}$	(17)	$S_{MC CONNAGHET} = \frac{a^2-bc}{(a+b)(a+c)}$	(39)	$S_{YULEW} = \frac{\sqrt{ad} - \sqrt{bc}}{\sqrt{ad} + \sqrt{bc}}$	(63)		
$D_{CANBERRA} = (b+c)^{\frac{2}{2}}$	(18)	$S_{TARWD} = \frac{na - (a+b)(a+c)}{na + (a+b)(a+c)}$	(40)	$S_{KULCZINSKI-I} = \frac{a}{b+c}$	(64)		
$D_{MANHATTAN} = b+c$	(19)	$S_{KULCZINSKI-II} = \frac{\frac{a}{2}(2a+b+c)}{(a+b)(a+c)}$	(41)	$S_{TANIMOTO} = \frac{a}{(a+b)+(a+c)-a}$	(65)		
$D_{MEAN-MANHATTAN} = \frac{b+c}{a+b+c+d}$	(20)	$S_{DRIVER\&KROEBER} = \frac{a}{2}(\frac{1}{a+b} + \frac{1}{a+c})$	(42)	$S_{DISPERSION} = \frac{ad-bc}{(a+b+c+d)^2}$	(66)		
$D_{CITYBLOCK} = b+c$	(21)	$S_{JOHNSON} = \frac{a}{a+b} + \frac{a}{a+c}$	(43)	$S_{HAMANN} = \frac{(a+d)-(b+c)}{a+b+c+d}$	(67)		
$D_{MDIKOWSKI} = (b+c)^{\frac{1}{2}}$	(22)	$S_{DENNIS} = \frac{ad-bc}{\sqrt{n(a+b)(a+c)}}$	(44)	$S_{MICHAEL} = \frac{4(ad-bc)}{(a+d)^2 + (b+c)^2}$	(68)		
		$S_{SIMPSON} = \frac{a}{\min(a+b, a+c)}$	(45)	$S_{GOODMAN\&KRUSKAL} = \frac{\sigma - \sigma'}{2n - \sigma'}$ where $\sigma = \max(a, b) + \max(c, d) + \max(a, c) + \max(b, d),$ $\sigma' = \max(a+c, b+d) + \max(a+b, c+d)$	(69)		
		$S_{BRAUN\&BANQUET} = \frac{a}{\max(a+b, a+c)}$	(46)				

Appendix B: Clustering Validity Measures

Liu et al. (2010)

Measure	Notation	Definition	Optimal value
1 Root-mean-square std dev	$RMSSTD$	$\{\sum_i \sum_{x \in C_i} \ x - c_i\ ^2 / [P \sum_i (n_i - 1)]\}^{\frac{1}{2}}$	Elbow
2 R-squared	RS	$(\sum_{x \in D} \ x - c\ ^2 - \sum_i \sum_{x \in C_i} \ x - c_i\ ^2) / \sum_{x \in D} \ x - c\ ^2$	Elbow
3 Modified Hubert Γ statistic	Γ	$\frac{2}{n(n-1)} \sum_{x \in D} \sum_{y \in D} d(x, y) d_{x \in C_i, y \in C_j}(c_i, c_j)$	Elbow
4 Calinski-Harabasz index	CH	$\frac{\sum_i n_i d^2(c_i, c)}{\sum_i \sum_{x \in C_i} d^2(x, c_i) / (n - NC)}$	Max
5 I index	I	$(\frac{1}{NC} \cdot \frac{\sum_{x \in D} d(x, c)}{\sum_i \sum_{x \in C_i} d(x, c_i)} \cdot \max_{i,j} d(c_i, c_j))^p$	Max
6 Dunn's indices	D	$\min_i \{ \min_j (\frac{\min_{x \in C_i, y \in C_j} d(x, y)}{\max_k \{ \max_{x, y \in C_k} d(x, y) \}}) \}$	Max
7 Silhouette index	S	$\frac{1}{NC} \sum_i \{ \frac{1}{n_i} \sum_{x \in C_i} \frac{b(x) - a(x)}{\max[b(x), a(x)]} \}$ $a(x) = \frac{1}{n_i - 1} \sum_{y \in C_i, y \neq x} d(x, y)$, $b(x) = \min_{j, j \neq i} [\frac{1}{n_j} \sum_{y \in C_j} d(x, y)]$	Max
8 Davies-Bouldin index	DB	$\frac{1}{NC} \sum_i \max_{j, j \neq i} \{ [\frac{1}{n_i} \sum_{x \in C_i} d(x, c_i) + \frac{1}{n_j} \sum_{x \in C_j} d(x, c_j)] / d(c_i, c_j) \}$	Min
9 Xie-Beni index	XB	$[\sum_i \sum_{x \in C_i} d^2(x, c_i)] / [n \cdot \min_{i, j \neq i} d^2(c_i, c_j)]$	Min
10 SD validity index	SD	$Dis(NC_{max}) Scat(NC) + Dis(NC)$ $Scat(NC) = \frac{1}{NC} \sum_i \ \sigma(C_i)\ / \ \sigma(D)\ $, $Dis(NC) = \frac{\max_{i,j} d(c_i, c_j)}{\min_{i,j} d(c_i, c_j)} \sum_i (\sum_j d(c_i, c_j))^{-1}$	Min
11 S_Dbw validity index	S_Dbw	$Scat(NC) + Dens_bw(NC)$ $Dens_bw(NC) = \frac{1}{NC(NC-1)} \sum_i [\sum_{j, j \neq i} \frac{\sum_{x \in C_i \cup C_j} f(x, u_{ij})}{\max\{\sum_{x \in C_i} f(x, c_i), \sum_{x \in C_j} f(x, c_j)\}}]$	Min

D : data set; n : number of objects in D ; c : center of D ; P : attributes number of D ; NC : number of clusters; C_i : the i -th cluster; n_i : number of objects in C_i ; c_i : center of C_i ; $\sigma(C_i)$: variance vector of C_i ; $d(x, y)$: distance between x and y ; $\|X_i\| = (X_i^T \cdot X_i)^{\frac{1}{2}}$

References

Research Papers

- Andrews, N. O. and Fox, E. A. (2007). Recent Developments in Document Clustering.
- Choi, S.-S., Cha, S.-H., and Tappert, C. C. (2010). A survey of binary similarity and distance measures. *Journal of Systemics, Cybernetics and Informatics* 8(1): 43-48.
- Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd* (Vol. 96, No. 34, pp. 226-231).
- Fahad, A., Alshatri, N., Tari, Z., Alamri, A., Khalil, I., Zomaya, A. Y., Foufou, S., and Bouras, A. (2014). A survey of clustering algorithms for big data: Taxonomy and empirical analysis. *IEEE Transactions on Emerging Topics in Computing* 2(3): 267-279.
- Kang, P., & Cho, S. (2009, September). K-means clustering seeds initialization based on centrality, sparsity, and isotropy. In *International Conference on Intelligent Data Engineering and Automated Learning* (pp. 109-117). Springer Berlin Heidelberg.
- Kohonen, T., Kaski, S., Lagus, K., Salojarvi, J., Honkela, J., Paatero, V., Saarela, A. (2000). Self organization of a massive document collection, *IEEE Transactions on Neural Networks* 11(3): 574-585.
- Lee, M., Pincombe, B. and Welsh, M. (2005). An empirical evaluation of models of text document similarity. *Cognitive Science*: 1254-1259.
- Liu, Y., Li, Z., Xiong, H., Gao, X., and Wu, J. (2010). Understanding of internal clustering validation measures, In: proceedings of the 2010 IEEE International Conference on Data Mining (ICDM'10): 911-916.
- Zhao, Y. and Karypis, G. (2002). Comparison of agglomerative and partitional document clustering algorithms. No. TR-02-014. MINNESOTA UNIV MINNEAPOLIS DEPT OF COMPUTER SCIENCE.

References

Other Materials

- Figure in the first page: <https://www.cs.ubc.ca/labs/imager/tr/2014/Overview/figures/warlogs.jpg>
- Training SOM example: <https://www.youtube.com/watch?v=HKj2ASG0DKQ>
- SOM training procedure: <https://www.youtube.com/watch?v=GdZckTLNqsY>