CS570: Introduction to Data Mining

Basic Clustering

Reading: Chapter 10 Han, Chapter 8 Tan

Anca Doloc-Mihu, Ph.D.

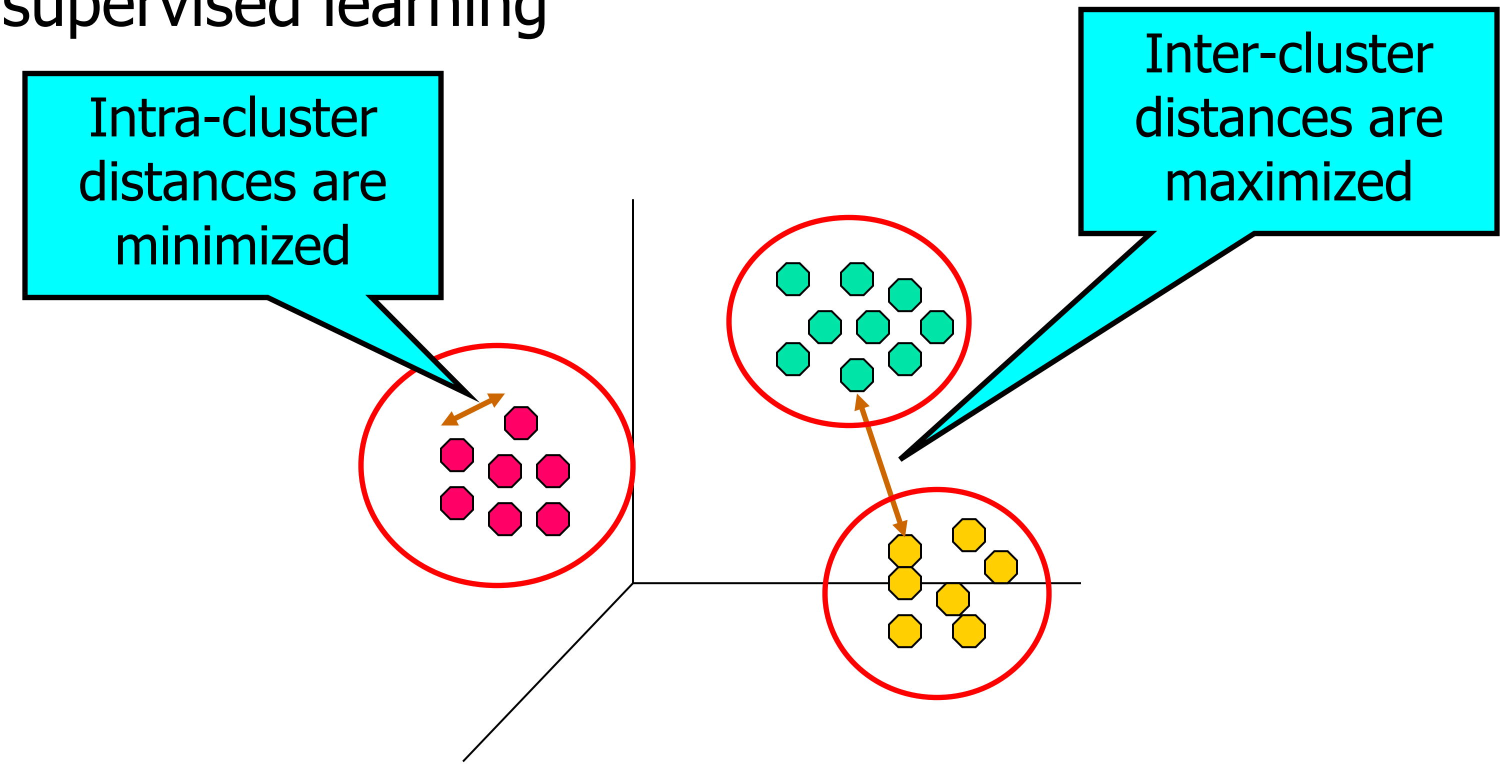
Slides courtesy of Li Xiong, Ph.D.,

Cluster Analysis

- Overview
- Partitioning methods
- Hierarchical methods
- Density-based methods
- Other Methods
- Outlier analysis
- Summary

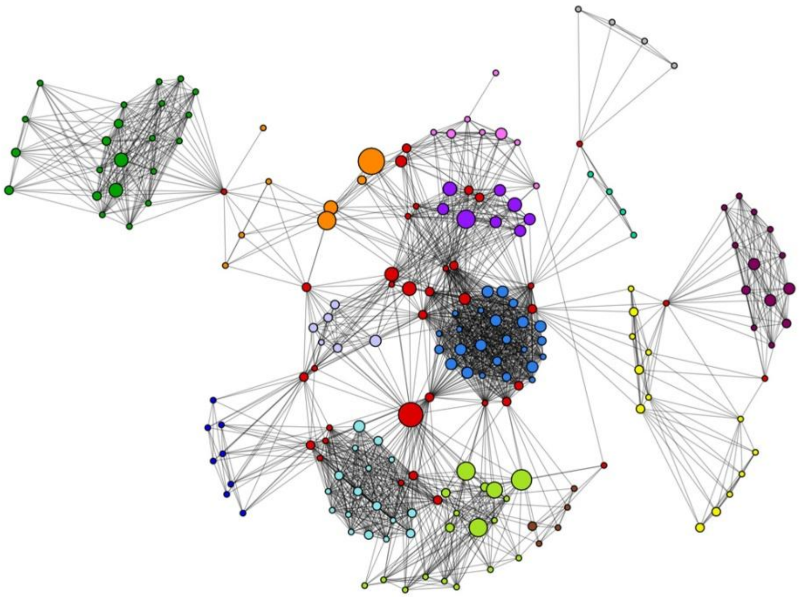
What is Cluster Analysis?

- Finding groups of objects (clusters)
  - Objects similar to one another in the same group
  - Objects different from the objects in other groups
- Unsupervised learning



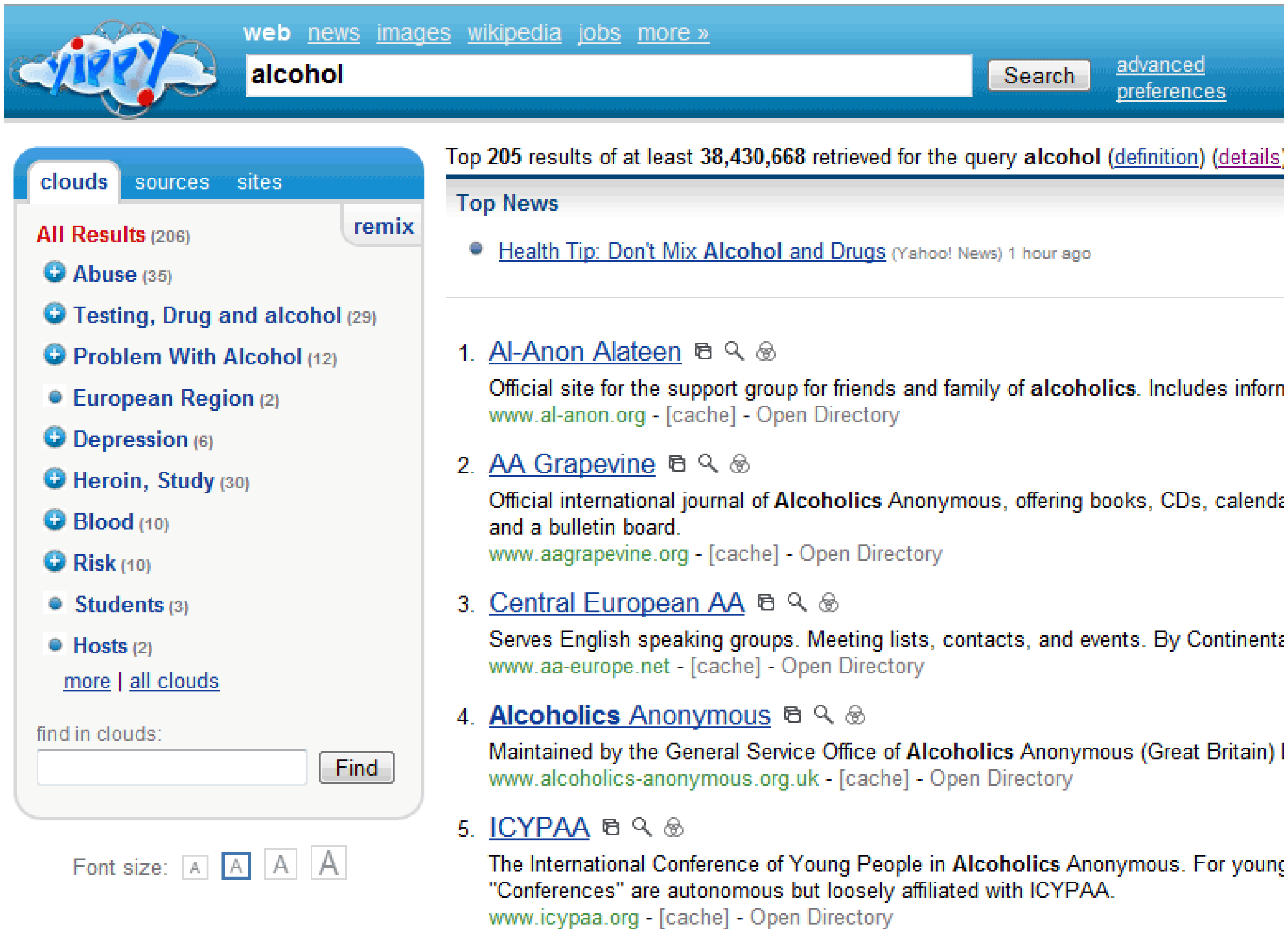
Clustering Applications

- Marketing research
- Social network analysis



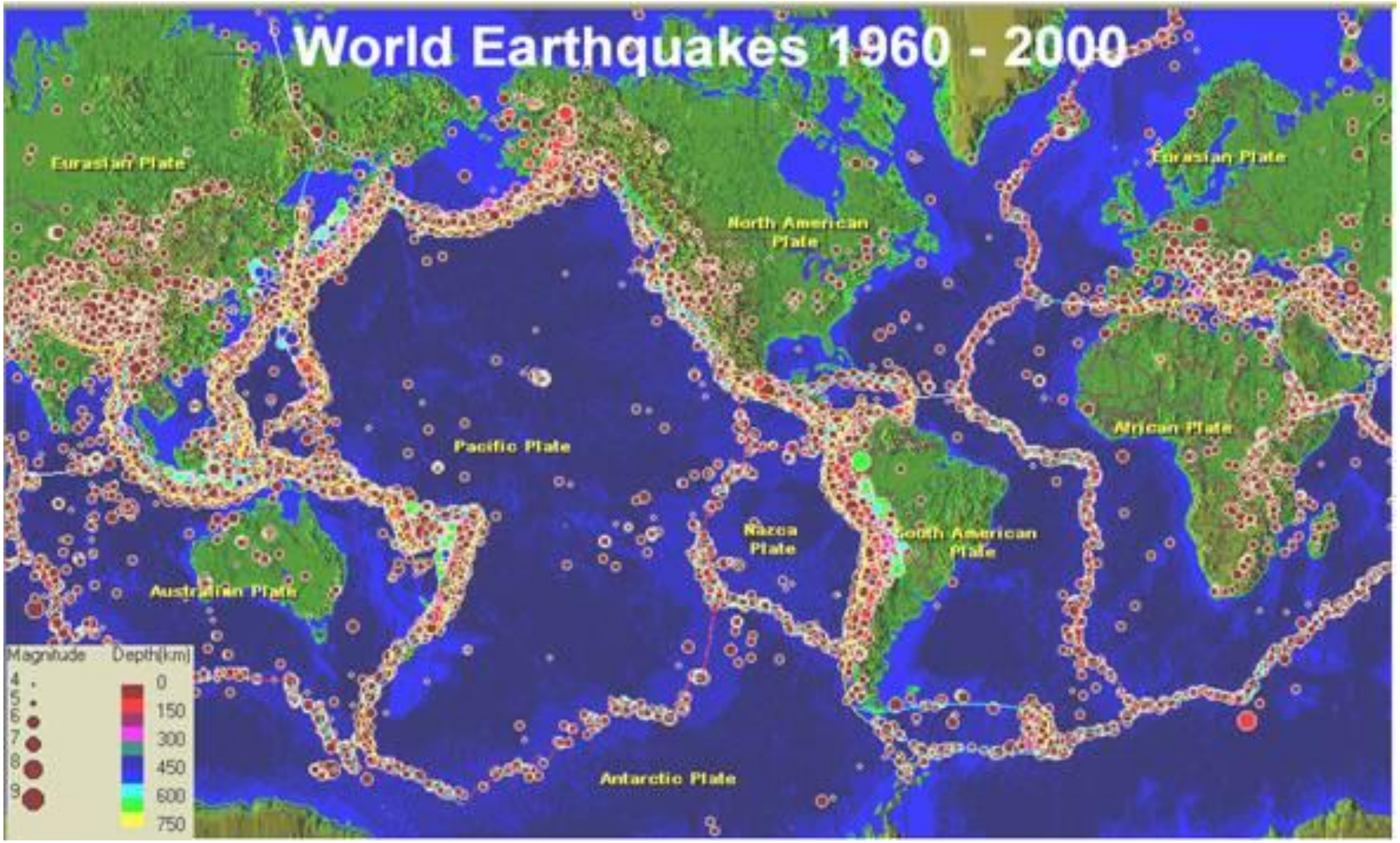
Clustering Applications

- WWW: Documents and search results clustering



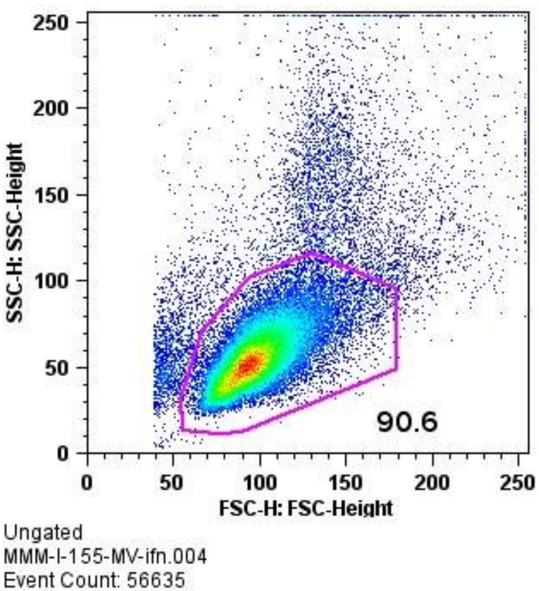
Clustering Applications

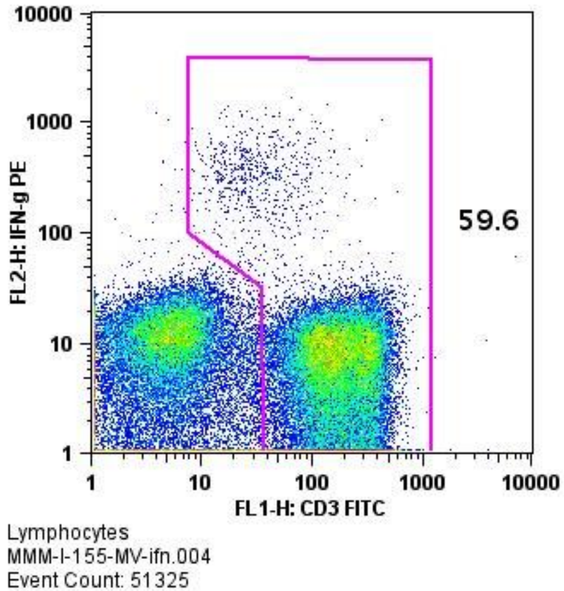
- Earthquake studies

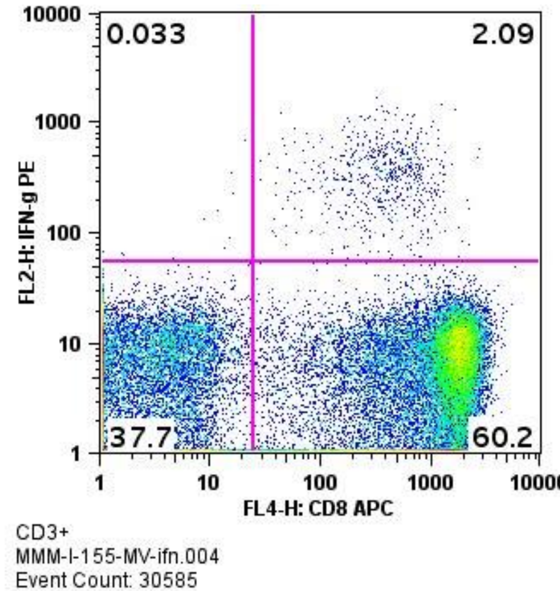


Clustering Applications

- Biology: plants and animals
- Bioinformatics: microarray data, flow cytometry data, genes and sequences





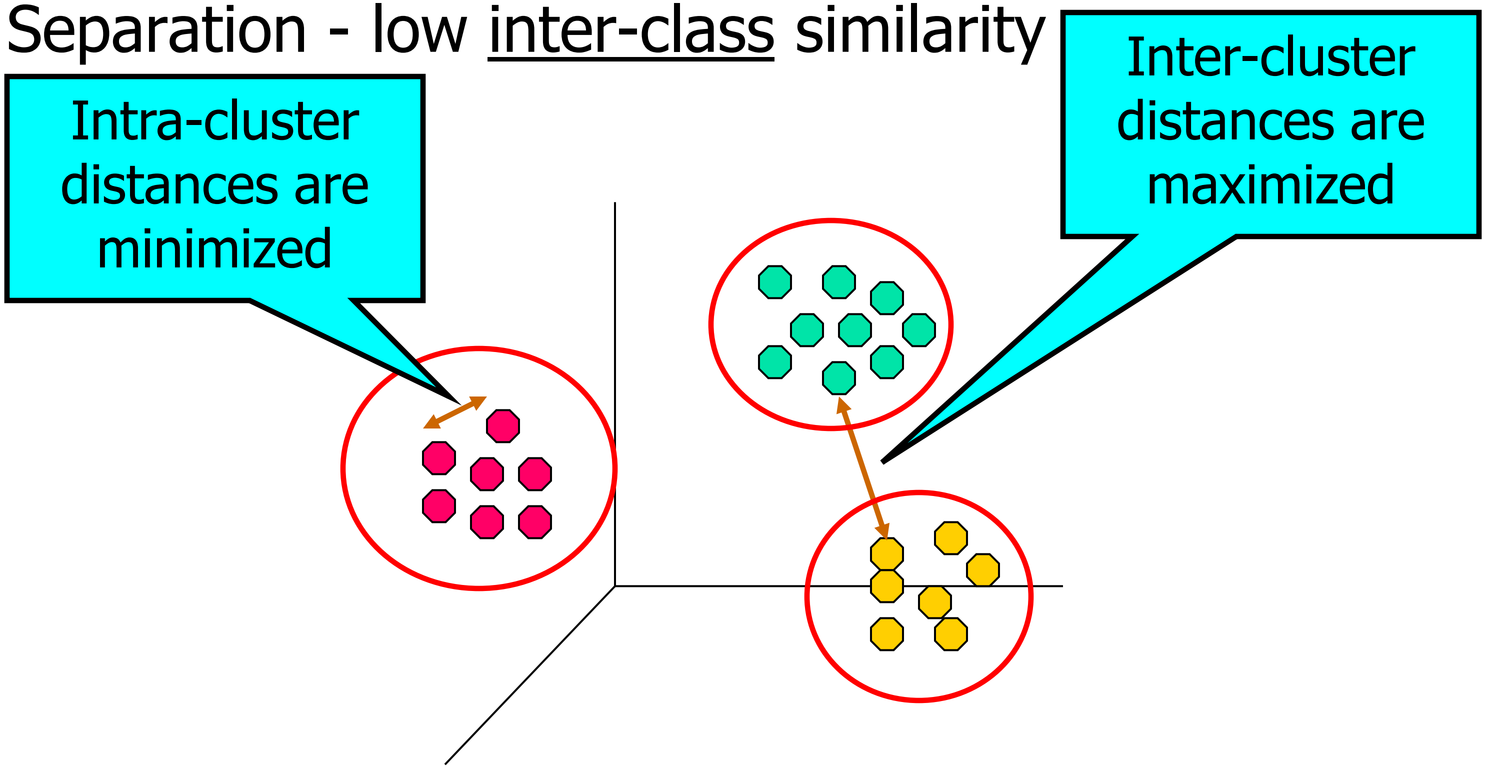


Requirements of Clustering

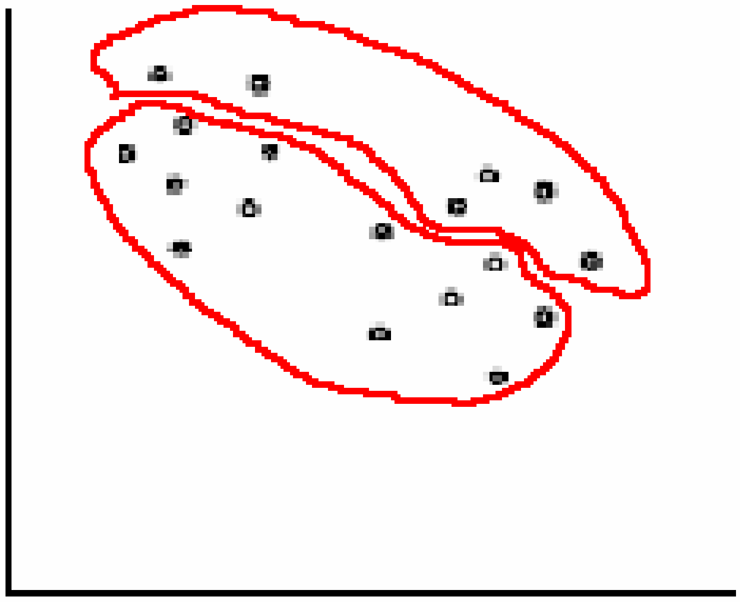
- Quality
- Scalability
- Ability to deal with different types of attributes
- Ability to handle dynamic data
- Ability to deal with noise and outliers
- Ability to deal with high dimensionality
- Minimal requirements for domain knowledge to determine input parameters
- Incorporation of user-specified constraints
- Interpretability and usability

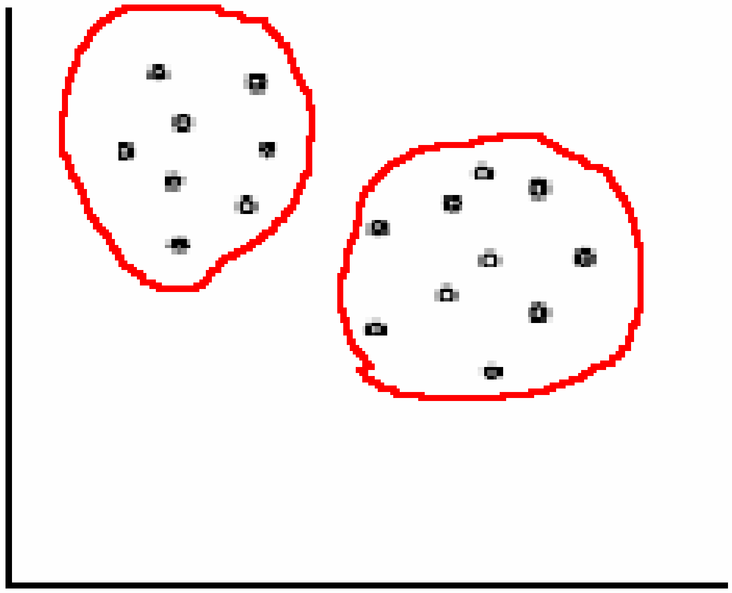
Quality: What Is Good Clustering?

- Agreement with "ground truth"
- A good clustering will produce high quality clusters with
  - Homogeneity - high intra-class similarity
  - Separation - low inter-class similarity



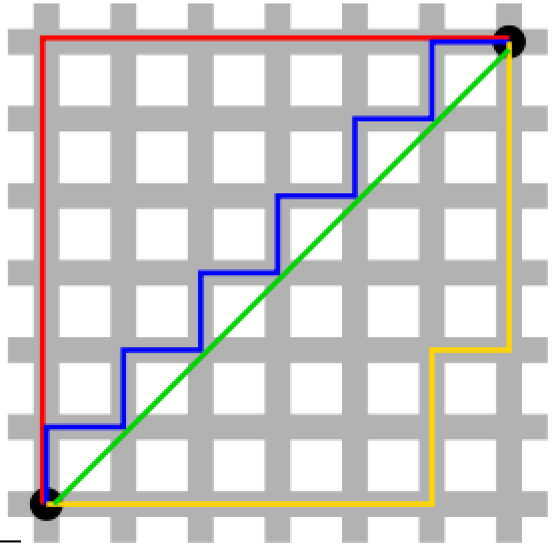
Bad Clustering vs. Good Clustering





Similarity or Dissimilarity between Data Objects

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix}$$



- Euclidean distance

$$d(i,j)=\sqrt{(|x_{i_1}-x_{j_1}|^2+|x_{i_2}-x_{j_2}|^2+\ldots+|x_{i_p}-x_{j_p}|^2)}$$

- Manhattan distance

$$d(i,j)=|x_{i_1}-x_{j_1}|+|x_{i_2}-x_{j_2}|+\ldots+|x_{i_p}-x_{j_p}|$$

- *Minkowski distance*

$$d(i,j)=\sqrt[q]{(|x_{i_1}-x_{j_1}|^q+|x_{i_2}-x_{j_2}|^q+\ldots+|x_{i_p}-x_{j_p}|^q)}$$

- Chebyshev distance

$$\lim_{p\to\infty}\left(\sum_{i=1}^{n}|x_i-y_i|^p\right)^{\frac{1}{p}}=\max_{i=1}^{n}|x_i-y_i|.$$

Other Similarity or Dissimilarity Metrics

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix}$$

- Pearson correlation $r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}.$
- Cosine measure $\frac{X_i \bullet X_j}{\|X_i\| \cdot \|X_j\|}$
- Jaccard coefficient $J(A,B) = \frac{|A \cap B|}{|A \cup B|}.$
- KL divergence, Bregman divergence, …

Different Attribute Types

- To compute $|x_{if} - x_{jf}|$
  - f is numeric (interval or ratio scale)
    - Normalization if necessary
  - f is ordinal
    - Mapping by rank

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

  - f is nominal
    - Mapping function

$|x_{if} - x_{jf}|$ = 0 if $x_{if} = x_{jf}$, or 1 otherwise

    - Hamming distance (edit distance) for strings

Clustering Approaches

- Partitioning approach:
  - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square errors
  - Typical methods: k-means, k-medoids, CLARANS
- Hierarchical approach:
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion
  - Typical methods: Diana, Agnes, BIRCH, ROCK, CAMELEON
- Density-based approach:
  - Based on connectivity and density functions
  - Typical methods: DBSACN, OPTICS, DenClue
- Others

Cluster Analysis

- Overview
- Partitioning methods
- Hierarchical methods
- Density-based methods
- Other Methods
- Outlier analysis
- Summary

Partitioning Algorithms: Basic Concept

- Partitioning method: Construct a partition of a database ***D*** of ***n*** objects into a set of ***k*** clusters, s.t., the sum of squared distance is minimized

$$\Sigma_{i=1}^{k}\Sigma_{p\in C_i}(p-m_i)^2$$

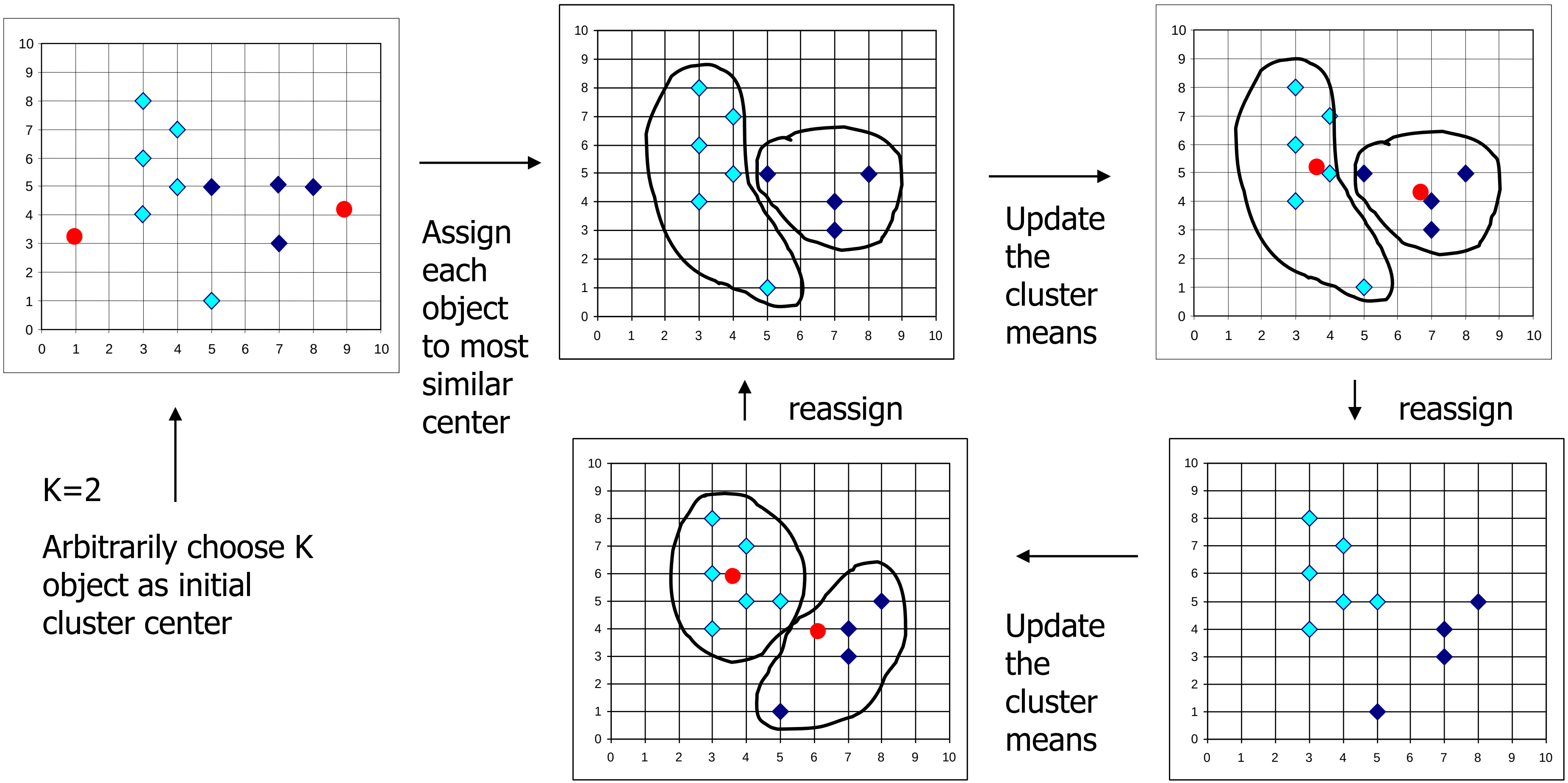
- Given a *k*, find a partition of *k clusters* that optimizes the chosen partitioning criterion
  - Global optimal: exhaustively enumerate all partitions
  - Heuristic methods: *k-means* and *k-medoids* algorithms
  - *k-means* (MacQueen'67): Each cluster is represented by the center of the cluster
  - *k-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Each cluster is represented by one of the objects in the cluster

K-Means Clustering: Lloyd Algorithm

- Given *k*, randomly choose *k* initial cluster centers
- Partition objects into *k* nonempty subsets by assigning each object to the cluster with the nearest centroid
- Update centroid, i.e. *mean point* of the cluster
- Go back to Step 2, stop when no more new assignment

The *K-Means* Clustering Method

- Example



K-means Clustering – Details

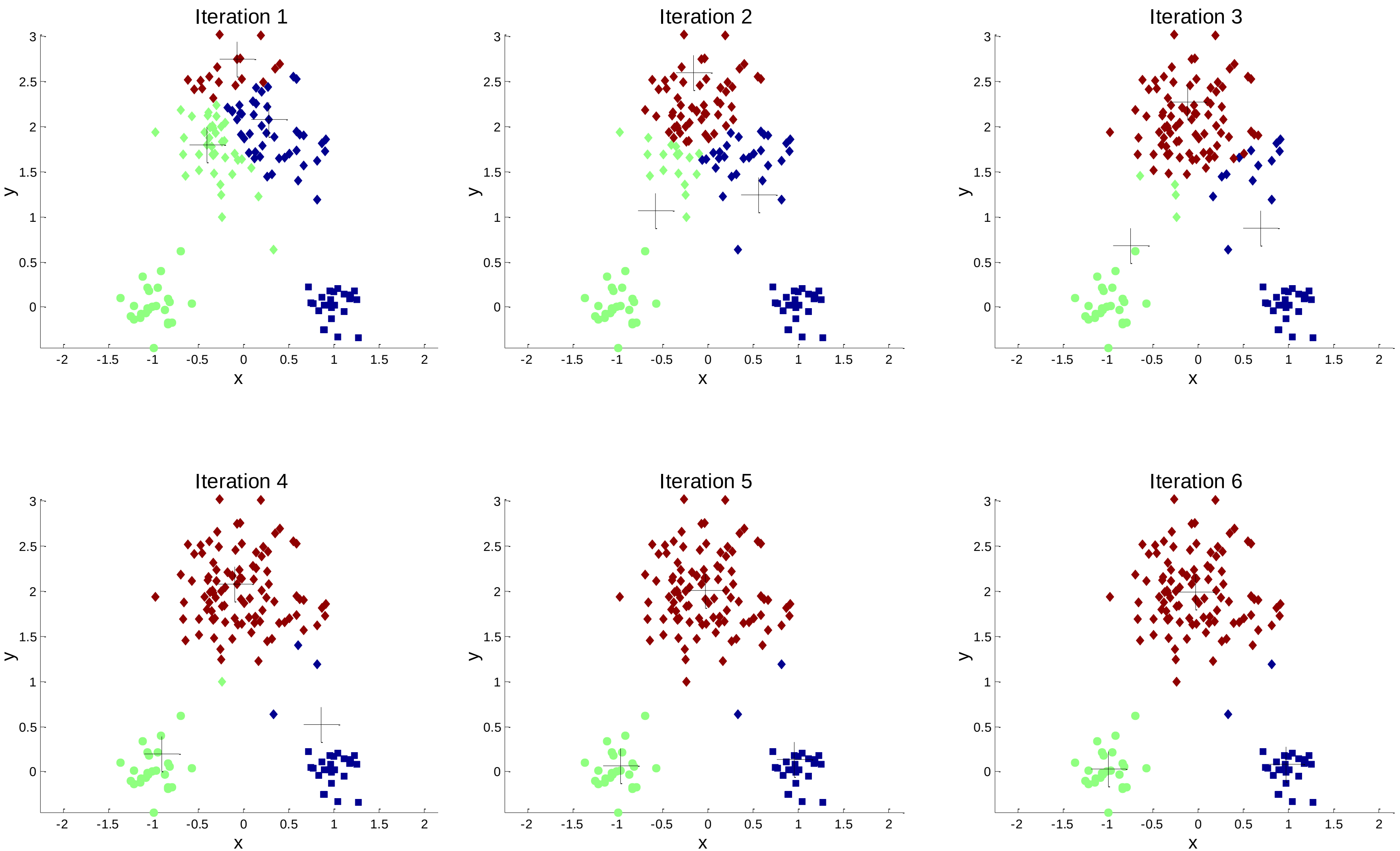
- Initial centroids are often chosen randomly.
- The centroid is (typically) the mean of the points in the cluster.
- 'Closeness' is measured by Euclidean distance, cosine similarity, correlation, etc.
- Most of the convergence happens in the first few iterations.
  - Often the stopping condition is changed to 'Until relatively few points change clusters'
- Complexity is $O(tkn)$

n is # objects, k is # clusters, and t is # iterations.

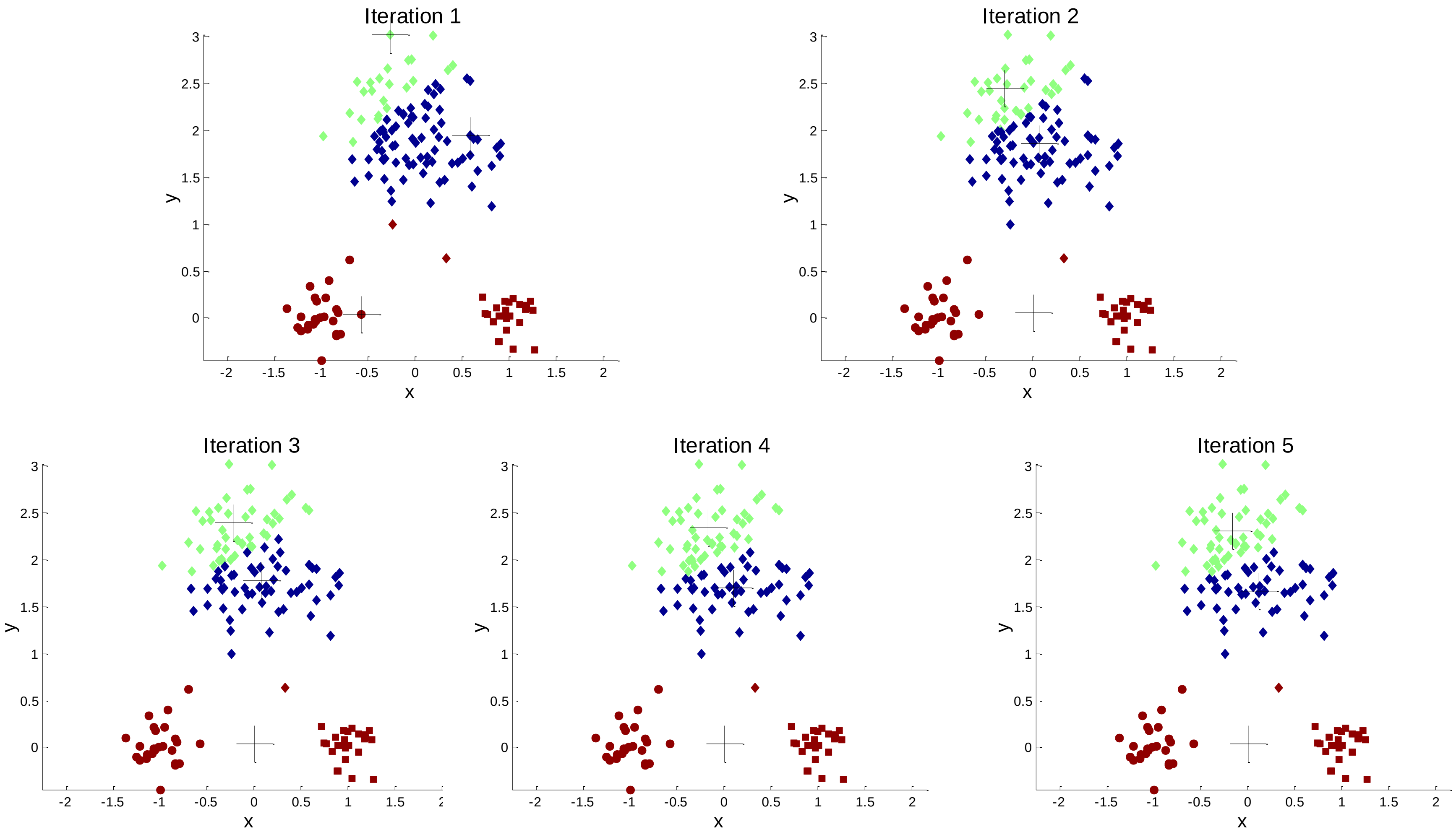
Comments on the *K-Means* Method

- Strength
  - Simple and works well for "regular" (spherical shape) disjoint clusters
  - Relatively efficient and scalable (normally, k, $t << n$)
  - Effective for small to medium size data sets
- Weakness
  - Need to specify *k,* the *number* of clusters, in advance
  - Depending on initial centroids, may terminate at a *local optimum*
    - Potential solutions
  - Unable to handle noisy data and *outliers*
  - Not suitable for clusters of
    - Different sizes
    - Non-convex shapes

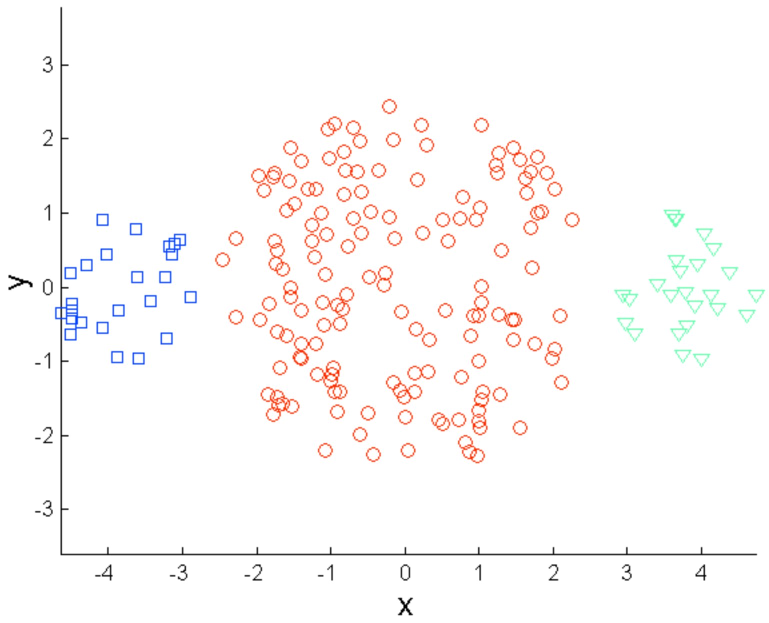
Importance of Choosing Initial Centroids – Case 1



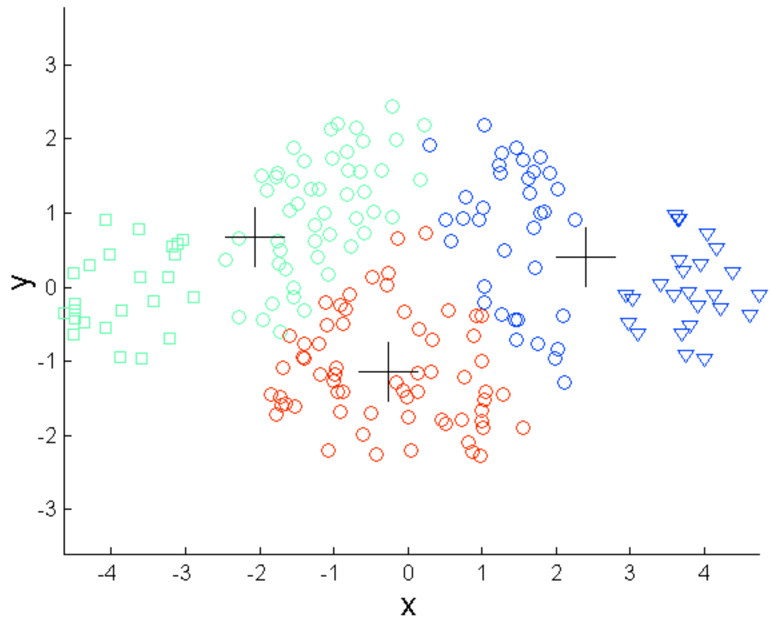
Importance of Choosing Initial Centroids – Case 2



Limitations of K-means: Differing Sizes

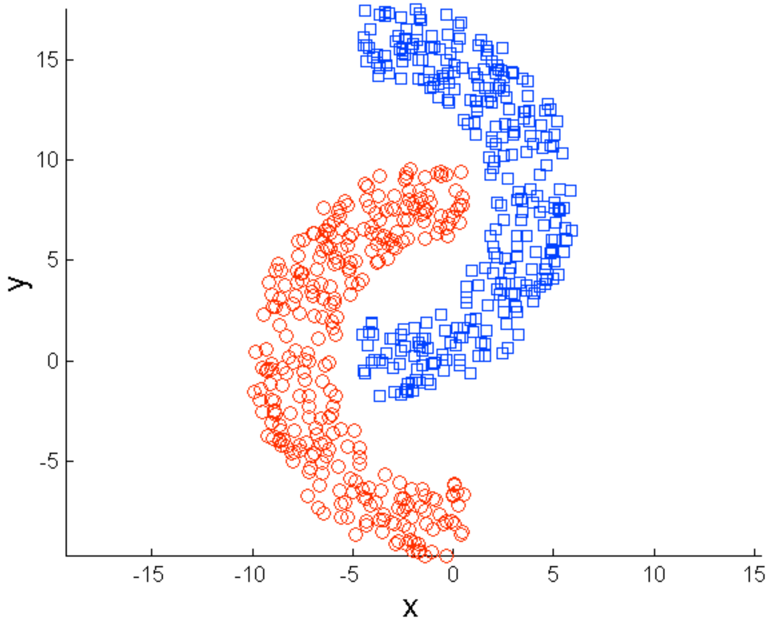


Original Points

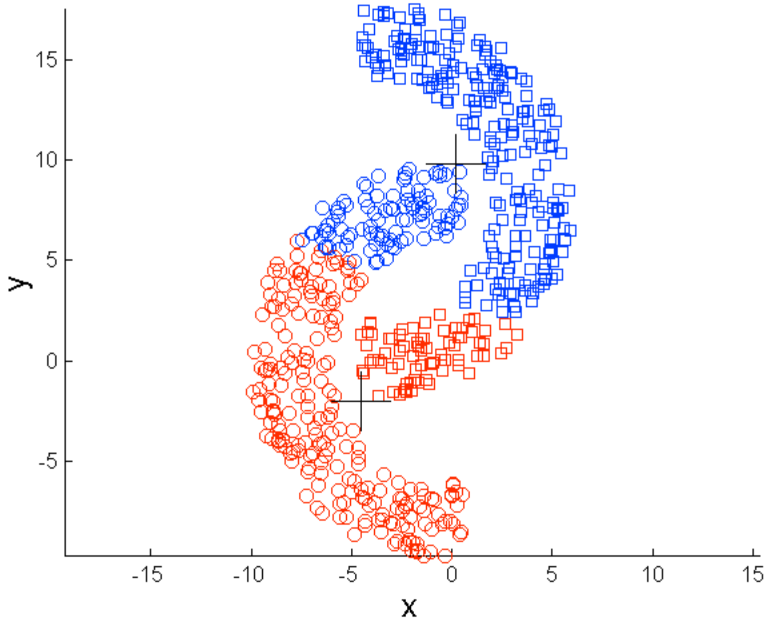


K-means (3 Clusters)

Limitations of K-means: Non-convex Shapes

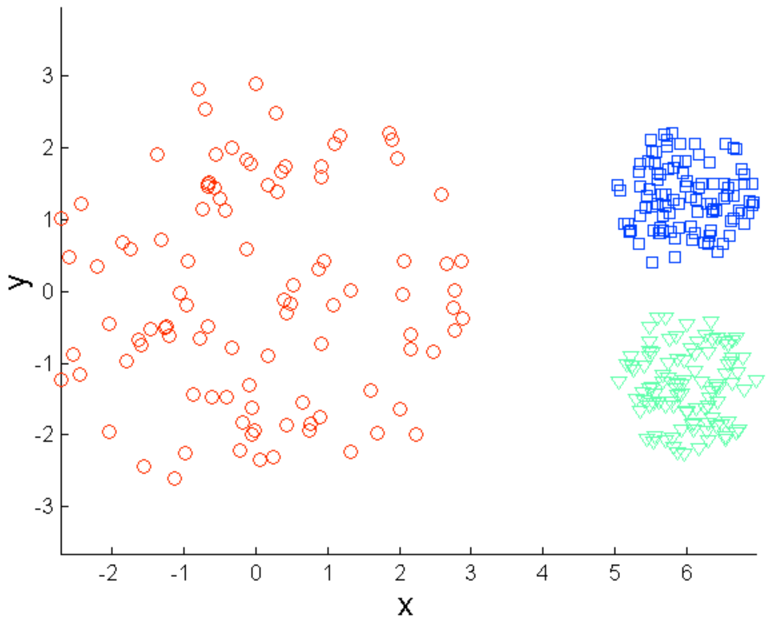


Original Points

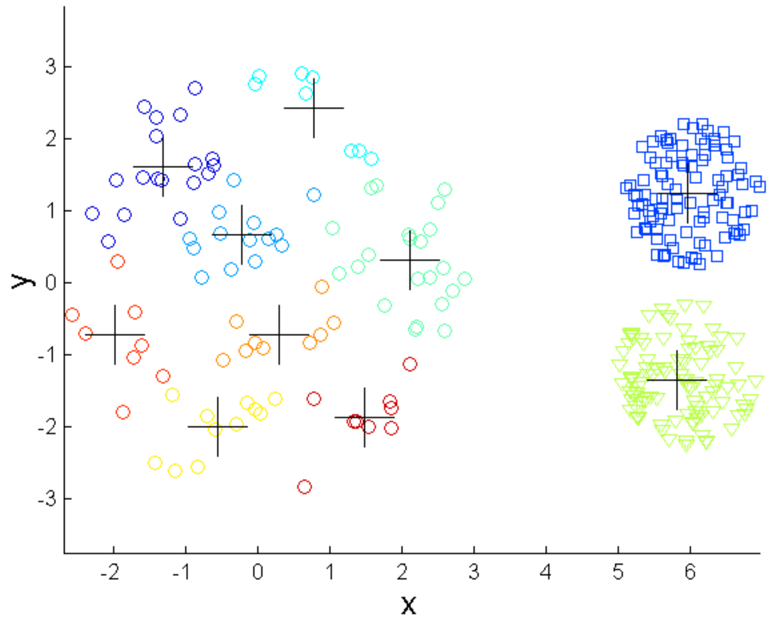


K-means (2 Clusters)

Overcoming K-means Limitations

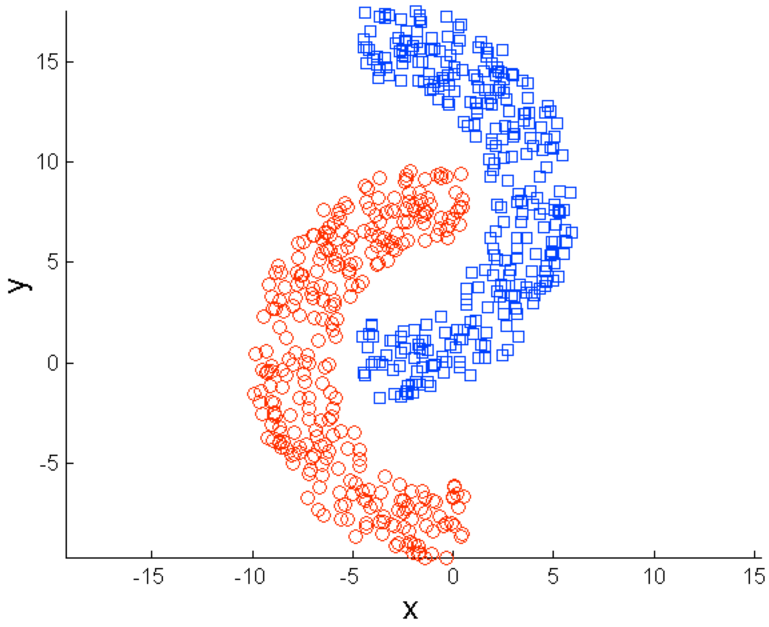


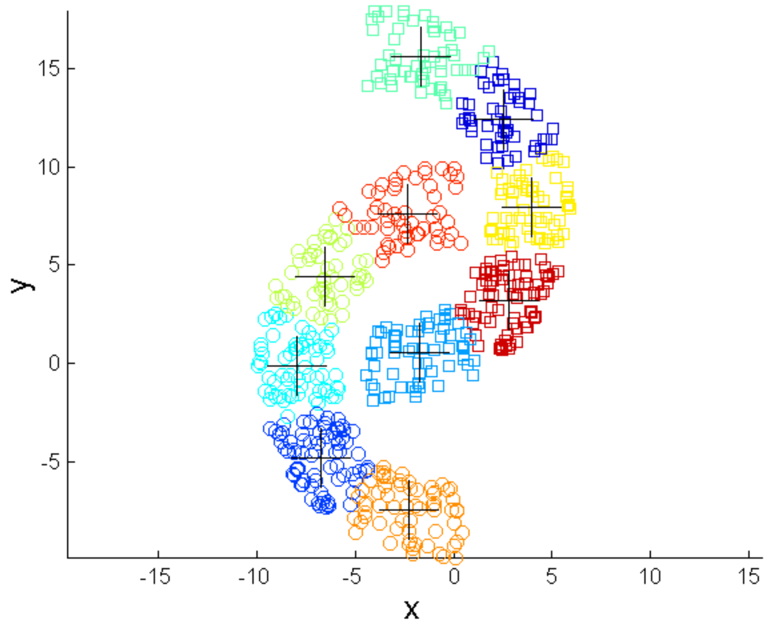
Original Points



K-means Clusters

Overcoming K-means Limitations





Original Points

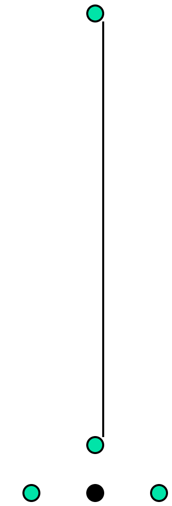
K-means Clusters

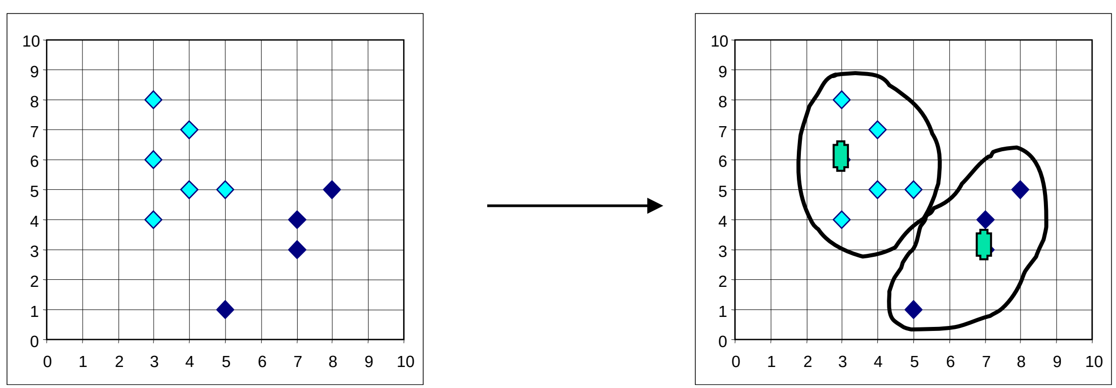
Variations of the *K-Means* Method

- A few variants of the *k-means* which differ in
  - Selection of the initial *k* means
  - Dissimilarity calculations
  - Strategies to calculate cluster means
- Handling categorical data: *k-modes* (Huang'98)
  - Replacing means of clusters with <u>modes</u>
  - Using new dissimilarity measures to deal with categorical objects
  - Using a <u>frequency</u>-based method to update modes of clusters
  - A mixture of categorical and numerical data: *k-prototype* method

K-Medoids Method?

- The k-means algorithm is sensitive to outliers !
  - Since an object with an extremely large value may substantially distort the distribution of the data.
- K-Medoids: Instead of taking the **mean** value of the object in a cluster as a reference point, **medoids** can be used, which is the **most centrally located** object in a cluster.





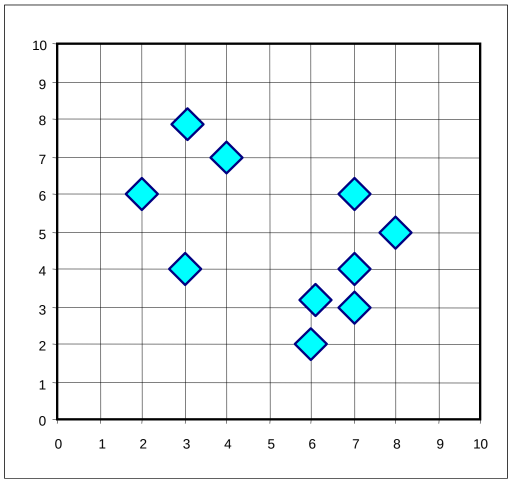
The *K-Medoids* Clustering Method

PAM (Kaufman and Rousseeuw, 1987)

- Arbitrarily select k objects as medoid
- Assign each data object in the given data set to most similar medoid.
- Randomly select nonmedoid object O'
- Compute total cost, S, of swapping a medoid object to O' (cost as total sum of absolute error)
- If S<0, then swap initial medoid with the new one
- Repeat until there is no change in the medoid.

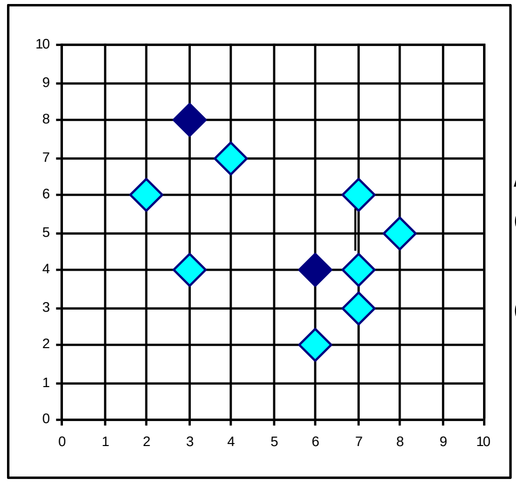
k-medoids and (n-k) instances pair-wise comparison

A Typical K-Medoids Algorithm (PAM)



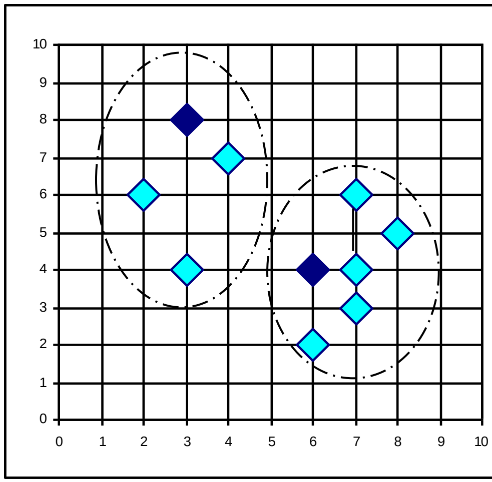
K=2

Arbitrary choose k objects as initial medoids

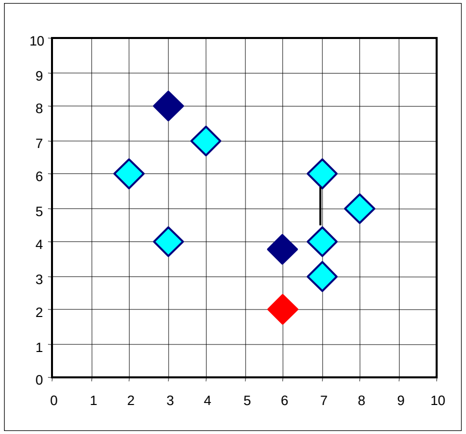


Assign each remaining object to nearest medoids

Total Cost = 20

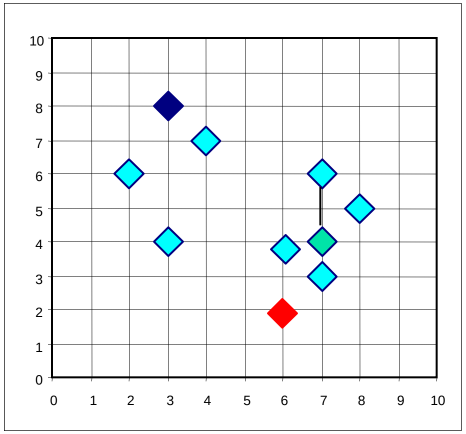


Randomly select a nonmedoid object,O_{random}



Compute total cost of swapping

Total Cost = 26



Swapping O and O_{ramdom}

If quality is improved.

Do loop

Until no change

What Is the Problem with PAM?

- Pam is more robust than k-means in the presence of noise and outliers
- Pam works efficiently for small data sets but does not **scale well** for large data sets.
  - Complexity? O(k(n-k)t)

    n is # of data,k is # of clusters, t is # of iterations

➔ Sampling based method,

CLARA(Clustering LARge Applications)

CLARA (Clustering Large Applications) (1990)

- *CLARA* (Kaufmann and Rousseeuw in 1990)
- It draws *multiple samples* of the data set, applies *PAM* on each sample, and gives the best clustering as the output
- Strength: deals with larger data sets than *PAM*
- Weakness:
  - Efficiency depends on the sample size
  - A good clustering based on samples will not necessarily represent a good clustering of the whole data set if the sample is biased

CLARANS ("Randomized" CLARA) *(1994)*

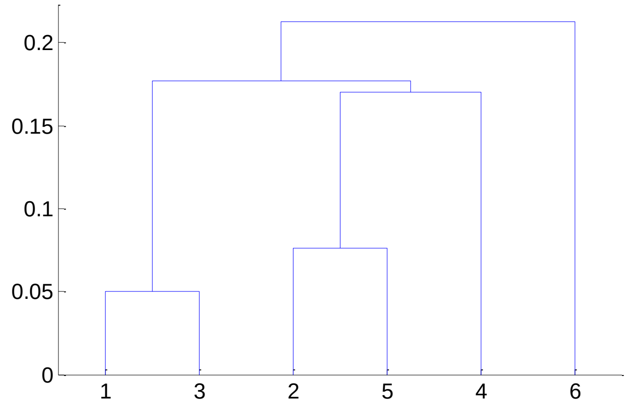
- *CLARANS* (A Clustering Algorithm based on Randomized Search) (Ng and Han'94)
- The clustering process can be presented as searching a graph where every node is a potential solution, that is, a set of *k* medoids
  - PAM examines neighbors for local minimum
  - CLARA works on subgraphs of samples
  - CLARANS examines neighbors dynamically
    - If local optimum is found, starts with new randomly selected node in search for a new local optimum

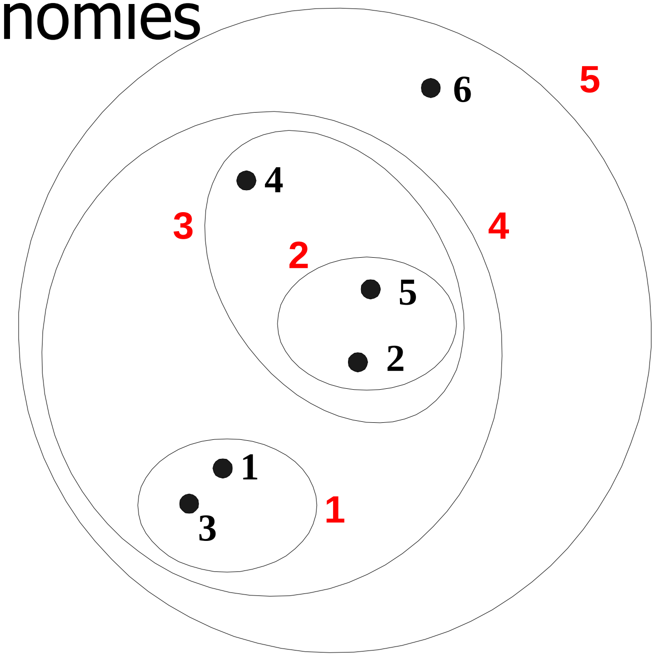
Cluster Analysis

- Overview
- Partitioning methods
- Hierarchical methods and graph-based methods
- Density-based methods
- Other Methods
- Outlier analysis
- Summary

Hierarchical Clustering

- Produces a set of nested clusters organized as a hierarchical tree
- Can be visualized as a dendrogram, a tree like diagram
  - Clustering obtained by cutting at desired level
- Do not have to assume any particular number of clusters
- May correspond to meaningful taxonomies





Hierarchical Clustering

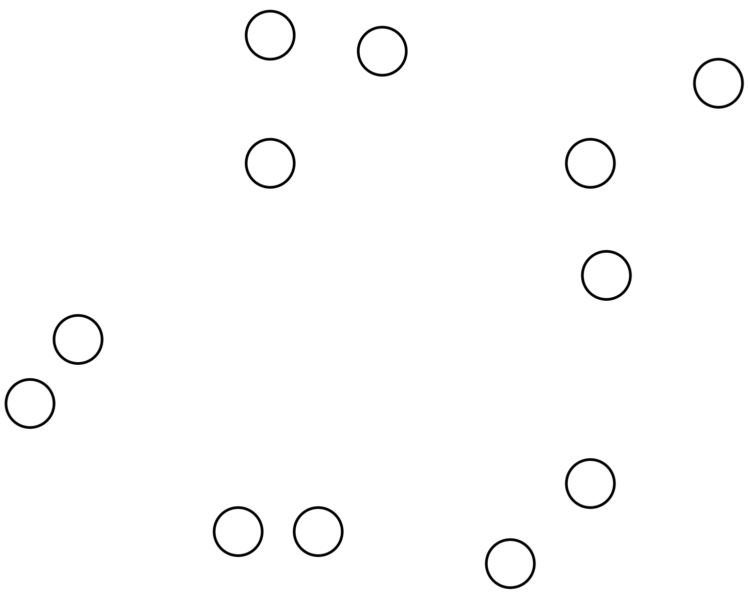
- Two main types of hierarchical clustering
  - Agglomerative:
    - Start with the points as individual clusters
    - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
  - Divisive:
    - Start with one, all-inclusive cluster
    - At each step, split a cluster until each cluster contains a point (or there are k clusters)

Agglomerative Clustering Algorithm

1. Compute the proximity matrix
2. Let each data point be a cluster
3. **Repeat**
4. Merge the two closest clusters
5. Update the proximity matrix
6. **Until** only a single cluster remains

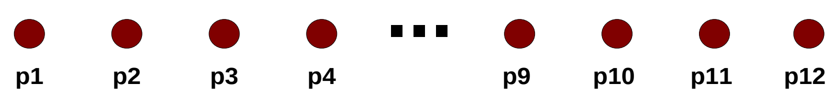
Starting Situation

- Start with clusters of individual points and a proximity matrix

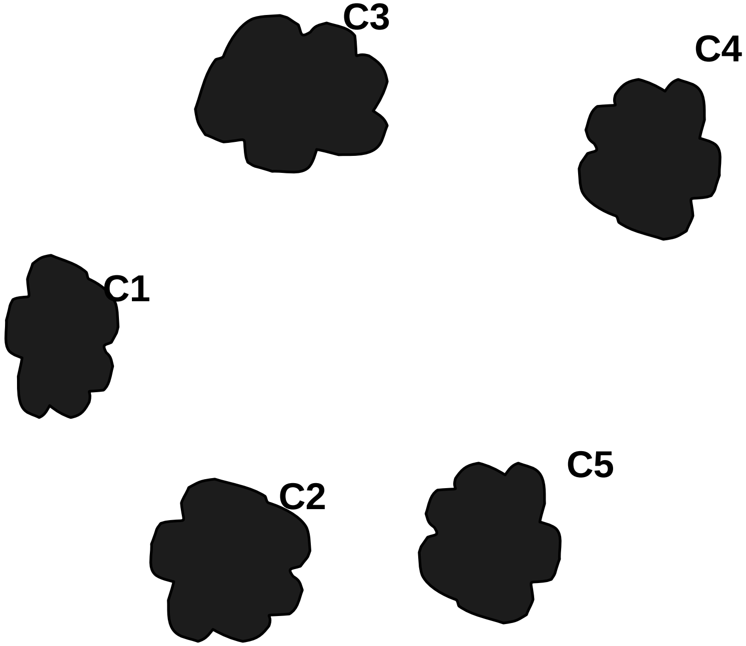


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

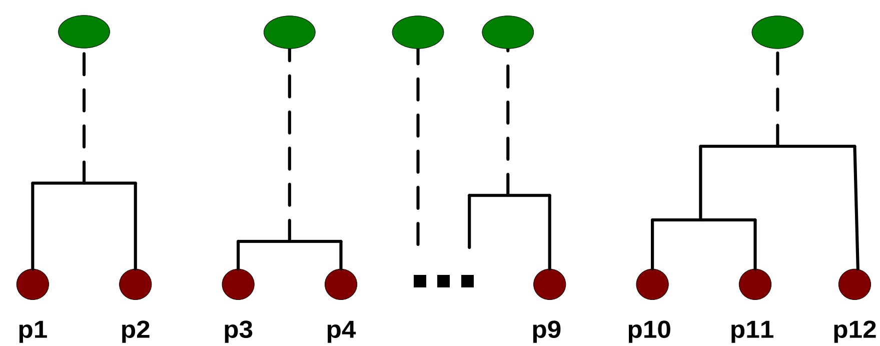


Intermediate Situation

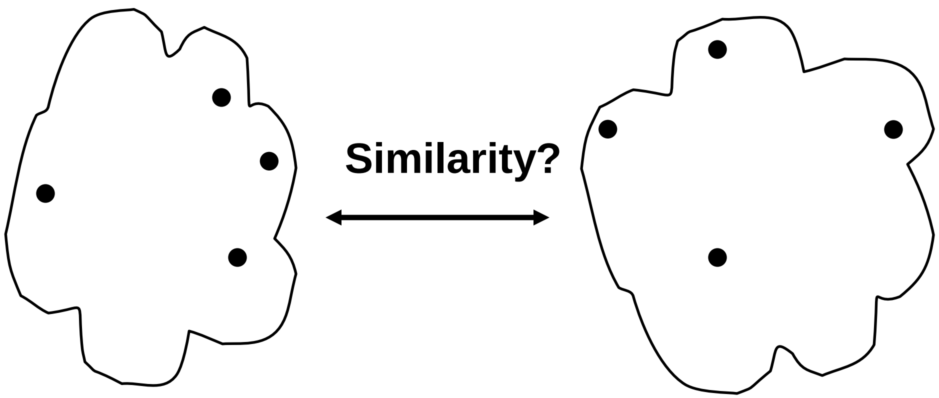


	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

Proximity Matrix



How to Define Inter-Cluster Similarity

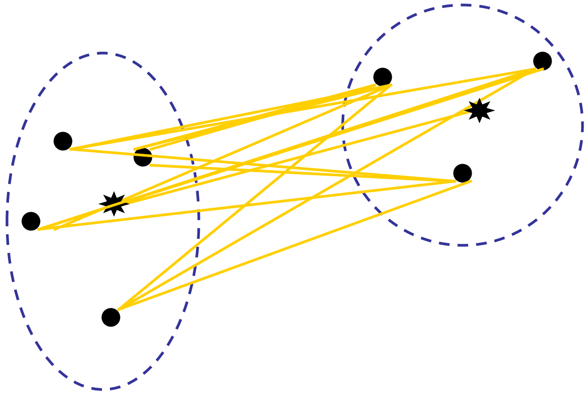


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

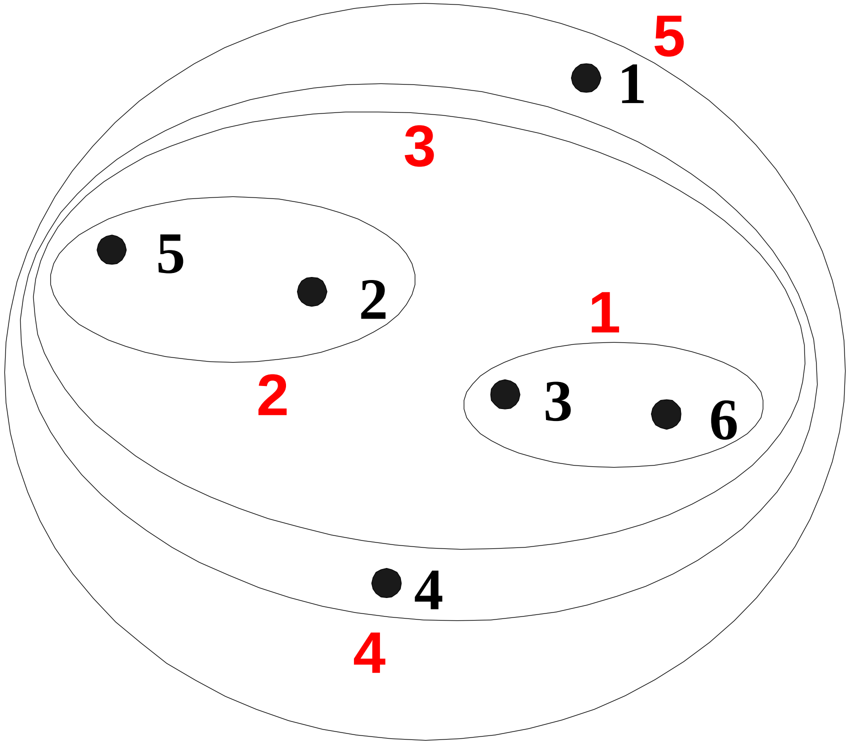
Proximity Matrix

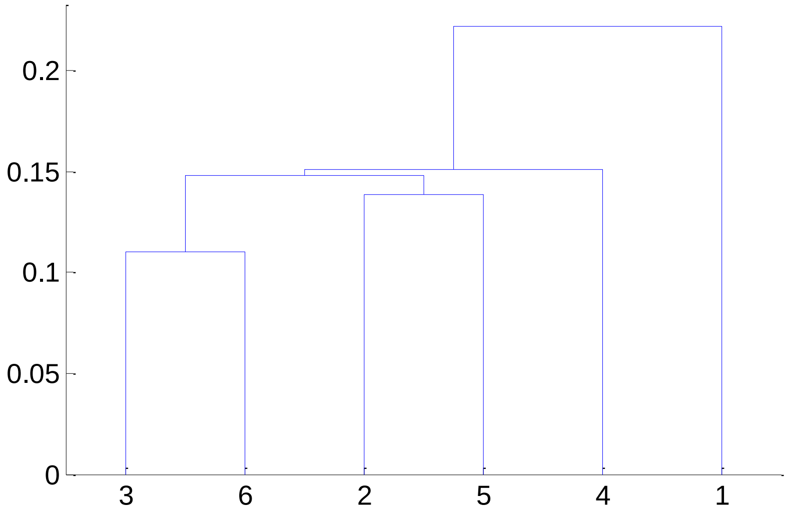
Distance Between Clusters

- ***Single Link***: smallest distance between points
- ***Complete Link:*** largest distance between points
- ***Average Link:*** average distance between points
- ***Centroid:*** distance between centroids



Hierarchical Clustering: MIN



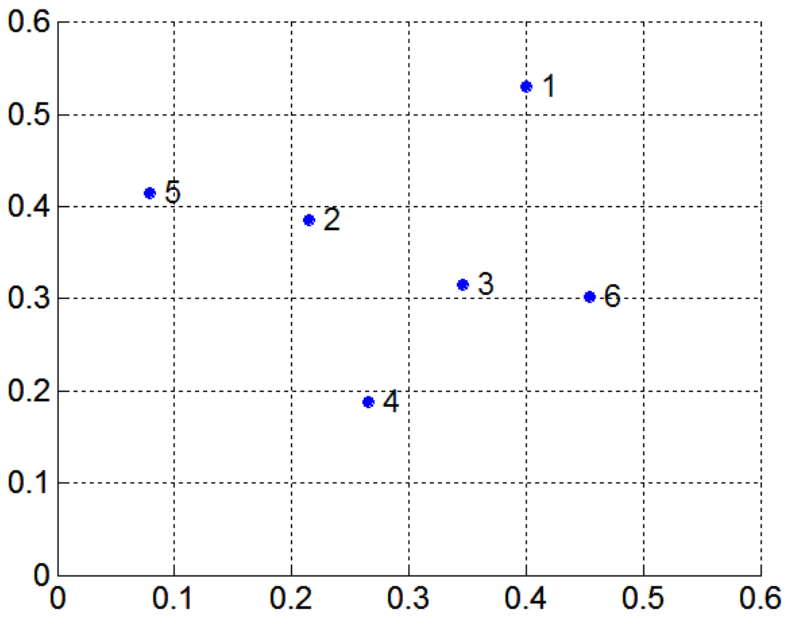


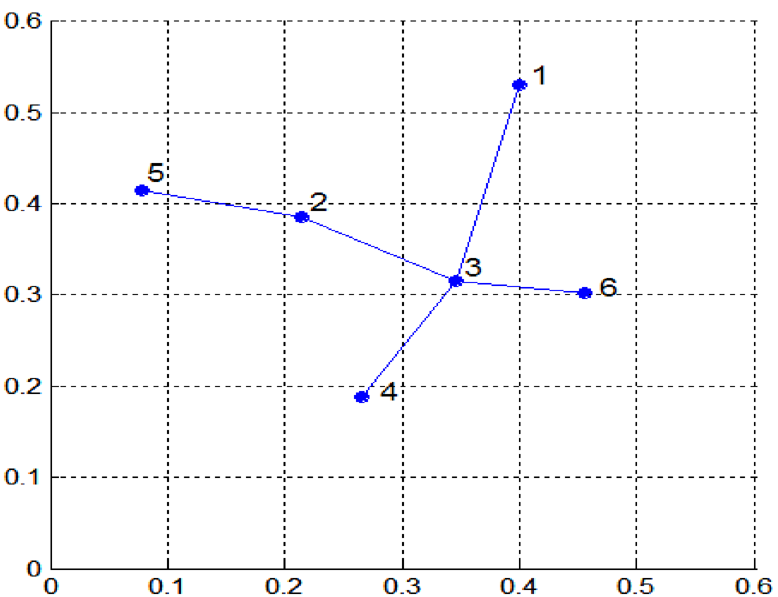
Nested Clusters

Dendrogram

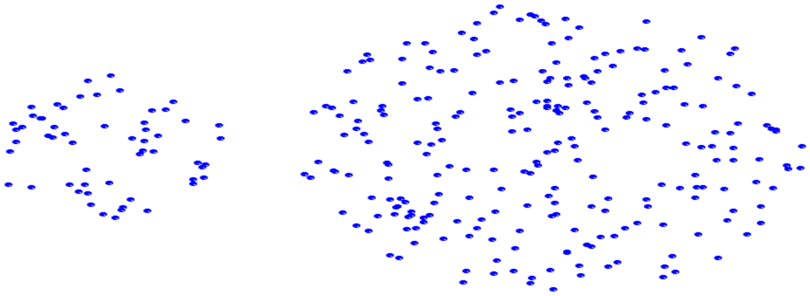
MST (Minimum Spanning Tree)

- An aggolomerative algorithm using minimum distance can be also called a minimal spanning tree (MST) algorithm
- Start with a tree that consists of any point
- In successive steps, look for the closest pair of points (p, q) such that one point (p) is in the current tree but the other (q) is not; add q to the tree and put an edge between p and q

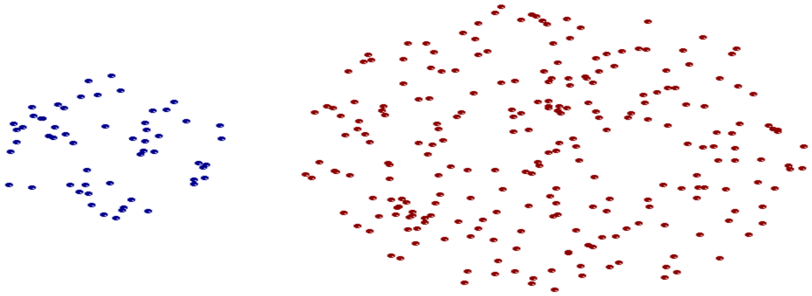




Strength of MIN



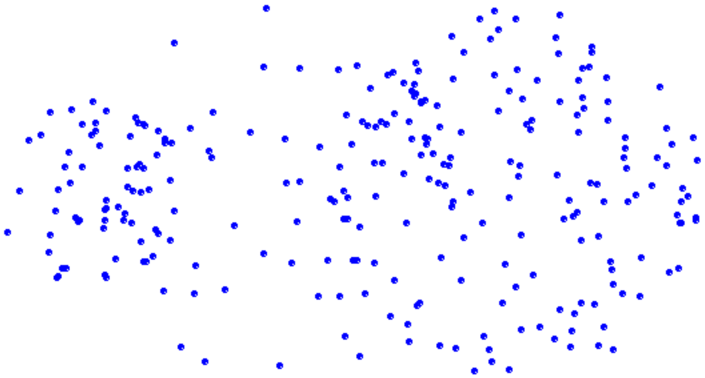
Original Points



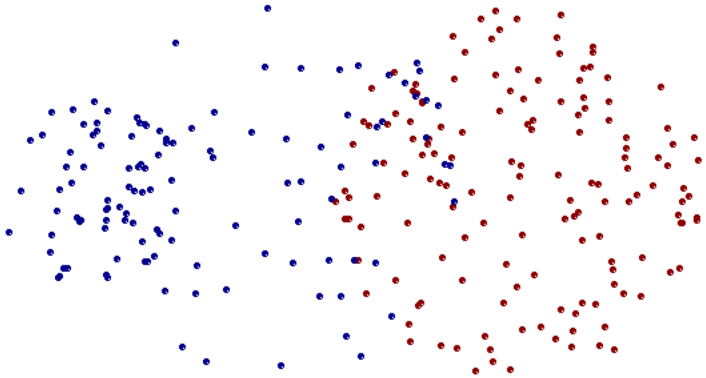
Two Clusters

- **Can handle non-elliptical shapes**

Limitations of MIN



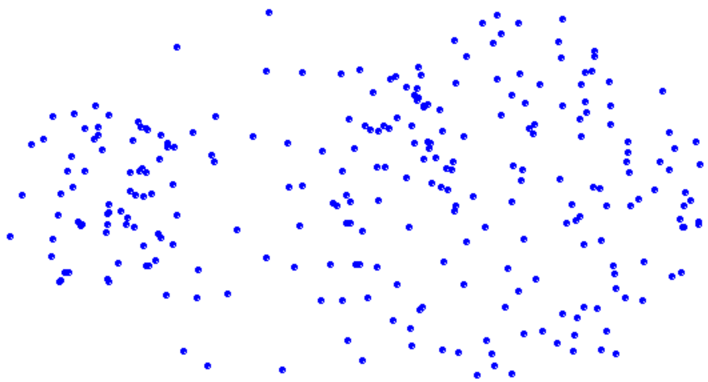
Original Points



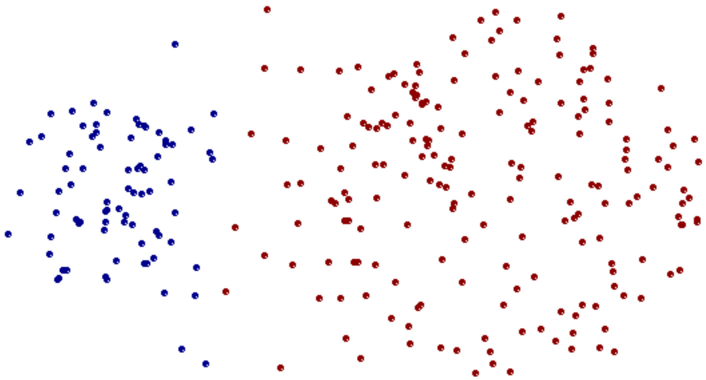
Two Clusters

- **Sensitive to noise and outliers**

Strength of MAX



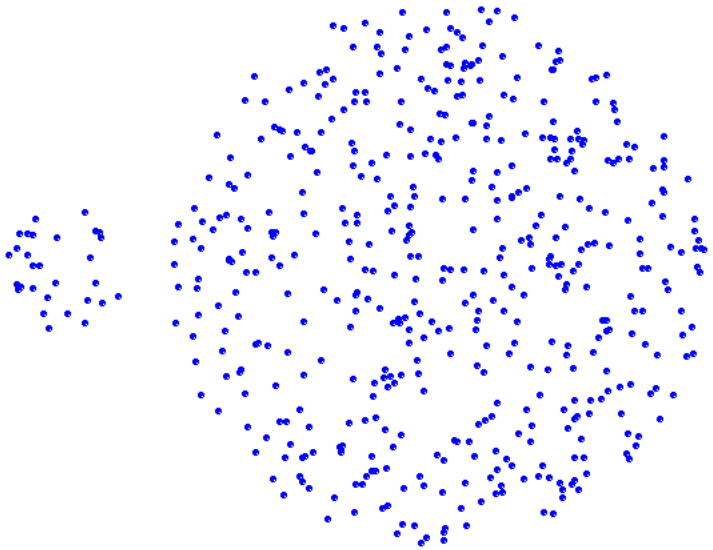
Original Points



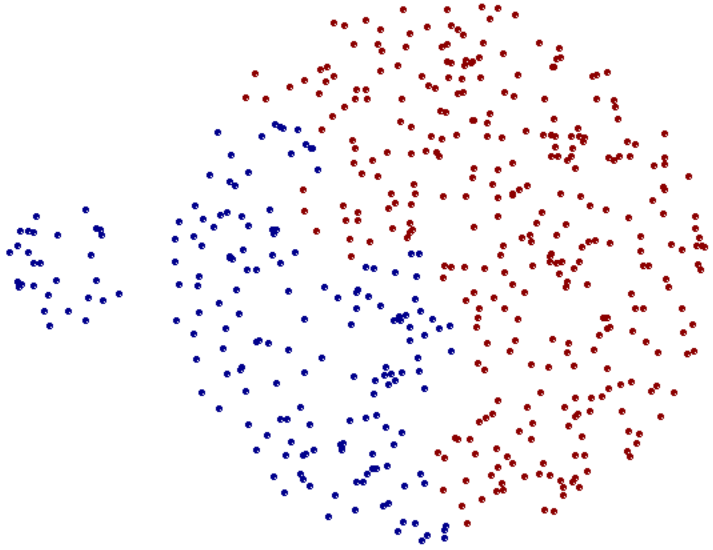
Two Clusters

- **Less susceptible to noise and outliers**

Limitations of MAX



Original Points



Two Clusters

- **Tends to break large clusters**
- **Biased towards globular clusters**

Hierarchical Clustering: Group Average

- Compromise between Single and Complete Link
- Strengths
  - Less susceptible to noise and outliers
  - Can handle categorical and numerical data
- Limitations
  - Biased towards globular clusters

Hierarchical Clustering: Major Weaknesses

- Do not scale well (N: number of points)
  - Space complexity: $O(N^2)$
  - Time complexity: $O(N^3)$

    $O(N^2 \log(N))$ for some cases/approaches

- Cannot undo what was done previously
- Quality varies in terms of distance measures
  - MIN (single link): susceptible to noise/outliers
  - MAX/GROUP AVERAGE: may not work well with non-globular clusters

Recent Hierarchical Clustering Methods

- BIRCH (1996): uses CF-tree and incrementally adjusts the quality of sub-clusters
- CURE(1998): uses representative points for inter-cluster distance
- ROCK (1999): clustering categorical data by neighbor and link analysis
- CHAMELEON (1999): hierarchical clustering using dynamic modeling

Cluster Analysis

- Overview
- Partitioning methods
- Hierarchical methods and graph-based methods
  - Classical methods
  - Recent methods
- Density-based methods
- Other Methods
- Outlier analysis
- Summary

References (1)

- R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. SIGMOD'98
- M. R. Anderberg. Cluster Analysis for Applications. Academic Press, 1973.
- M. Ankerst, M. Breunig, H.-P. Kriegel, and J. Sander. Optics: Ordering points to identify the clustering structure, SIGMOD'99.
- P. Arabie, L. J. Hubert, and G. De Soete. Clustering and Classification. World Scientific, 1996
- Beil F., Ester M., Xu X.: "Frequent Term-Based Text Clustering", KDD'02
- M. M. Breunig, H.-P. Kriegel, R. Ng, J. Sander. LOF: Identifying Density-Based Local Outliers. SIGMOD 2000.
- M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases. KDD'96.
- M. Ester, H.-P. Kriegel, and X. Xu. Knowledge discovery in large spatial databases: Focusing techniques for efficient class identification. SSD'95.
- D. Fisher. Knowledge acquisition via incremental conceptual clustering. Machine Learning, 2:139-172, 1987.
- D. Gibson, J. Kleinberg, and P. Raghavan. Clustering categorical data: An approach based on dynamic systems. VLDB'98.

References (2)

- V. Ganti, J. Gehrke, R. Ramakrishan. CACTUS Clustering Categorical Data Using Summaries. *KDD'99*.
- D. Gibson, J. Kleinberg, and P. Raghavan. Clustering categorical data: An approach based on dynamic systems. In Proc. VLDB'98.
- S. Guha, R. Rastogi, and K. Shim. Cure: An efficient clustering algorithm for large databases. SIGMOD'98.
- S. Guha, R. Rastogi, and K. Shim. ROCK: A robust clustering algorithm for categorical attributes. In *ICDE'99*, pp. 512-521, Sydney, Australia, March 1999.
- A. Hinneburg, D.l A. Keim: An Efficient Approach to Clustering in Large Multimedia Databases with Noise. KDD'98.
- A. K. Jain and R. C. Dubes. Algorithms for Clustering Data. Printice Hall, 1988.
- G. Karypis, E.-H. Han, and V. Kumar. CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling. *COMPUTER*, 32(8): 68-75, 1999.
- L. Kaufman and P. J. Rousseeuw. Finding Groups in Data: an Introduction to Cluster Analysis. John Wiley & Sons, 1990.
- E. Knorr and R. Ng. Algorithms for mining distance-based outliers in large datasets. VLDB'98.
- G. J. McLachlan and K.E. Bkasford. Mixture Models: Inference and Applications to Clustering. John Wiley and Sons, 1988.
- P. Michaud. Clustering techniques. Future Generation Computer systems, 13, 1997.
- R. Ng and J. Han. Efficient and effective clustering method for spatial data mining. VLDB'94.

References (3)

- *L. Parsons, E. Haque and H. Liu*, Subspace Clustering for High Dimensional Data: A Review , SIGKDD Explorations, 6(1), June 2004
- E. Schikuta. Grid clustering: An efficient hierarchical clustering method for very large data sets. Proc. 1996 Int. Conf. on Pattern Recognition,.
- G. Sheikholeslami, S. Chatterjee, and A. Zhang. WaveCluster: A multi-resolution clustering approach for very large spatial databases. VLDB'98.
- A. K. H. Tung, J. Han, L. V. S. Lakshmanan, and R. T. Ng. *Constraint-Based Clustering in Large Databases*, *ICDT'01.*
- A. K. H. Tung, J. Hou, and J. Han. *Spatial Clustering in the Presence of Obstacles* , *ICDE'01*
- H. Wang, W. Wang, J. Yang, and P.S. Yu. Clustering by pattern similarity in large data sets, *SIGMOD'*02.
- W. Wang, Yang, R. Muntz, STING: A Statistical Information grid Approach to Spatial Data Mining, VLDB'97.
- T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH : an efficient data clustering method for very large databases. SIGMOD'96.

Typical Alternatives to Calculate the Distance between Clusters

- Single link: smallest distance between an element in one cluster and an element in the other, i.e., $dis(K_i, K_j) = min(t_{ip}, t_{jq})$
- Complete link: largest distance between an element in one cluster and an element in the other, i.e., $dis(K_i, K_j) = max(t_{ip}, t_{jq})$
- Average: avg distance between an element in one cluster and an element in the other, i.e., $dis(K_i, K_j) = avg(t_{ip}, t_{jq})$
- Centroid: distance between the centroids of two clusters, i.e., $dis(K_i, K_j) = dis(C_i, C_j)$
- Medoid: distance between the medoids of two clusters, i.e., $dis(K_i, K_j) = dis(M_i, M_j)$
  - Medoid: one chosen, centrally located object in the cluster