Welcome to:

# Machine Learning

Welcome to:

# Unit – 5: Clustering Techniques

# Outline



- The Clustering Task and the Requirements for Cluster Analysis
- Overview of Some Basic Clustering Methods
- Hierarchical Methods:
  - Agglomerate versus Divisive Hierarchical Clustering,
  - Distance Measures,
  - Probabilistic Hierarchical Clustering,
  - Multiphase Hierarchical Clustering Using Clustering Feature Trees
- Partitioning Methods:
  - *k*-Means Clustering, *k*-Medoids Clustering
- Density-Based Clustering:
  - DBSCAN - Density-Based Clustering Based on Connected Regions with High Density
- Measuring Clustering Goodness

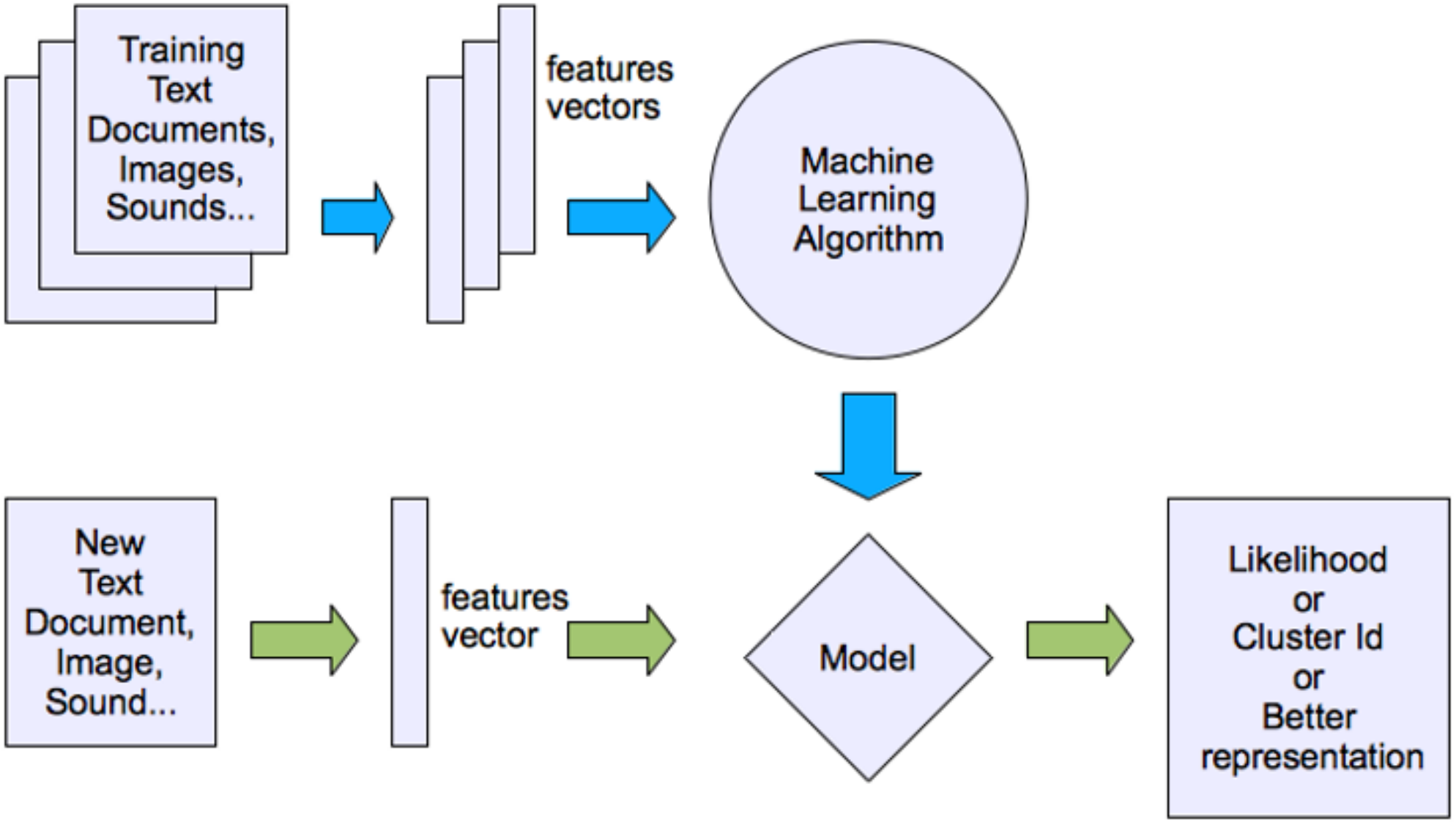
# Objectives



- In this unit, we planned to made the students to understand about:
  - Clustering and its importance in machine learning
  - Hierarchical Methods
  - Partitioning Methods
  - Density-Based Clustering Methods
  - Criteria to determine Clustering Goodness

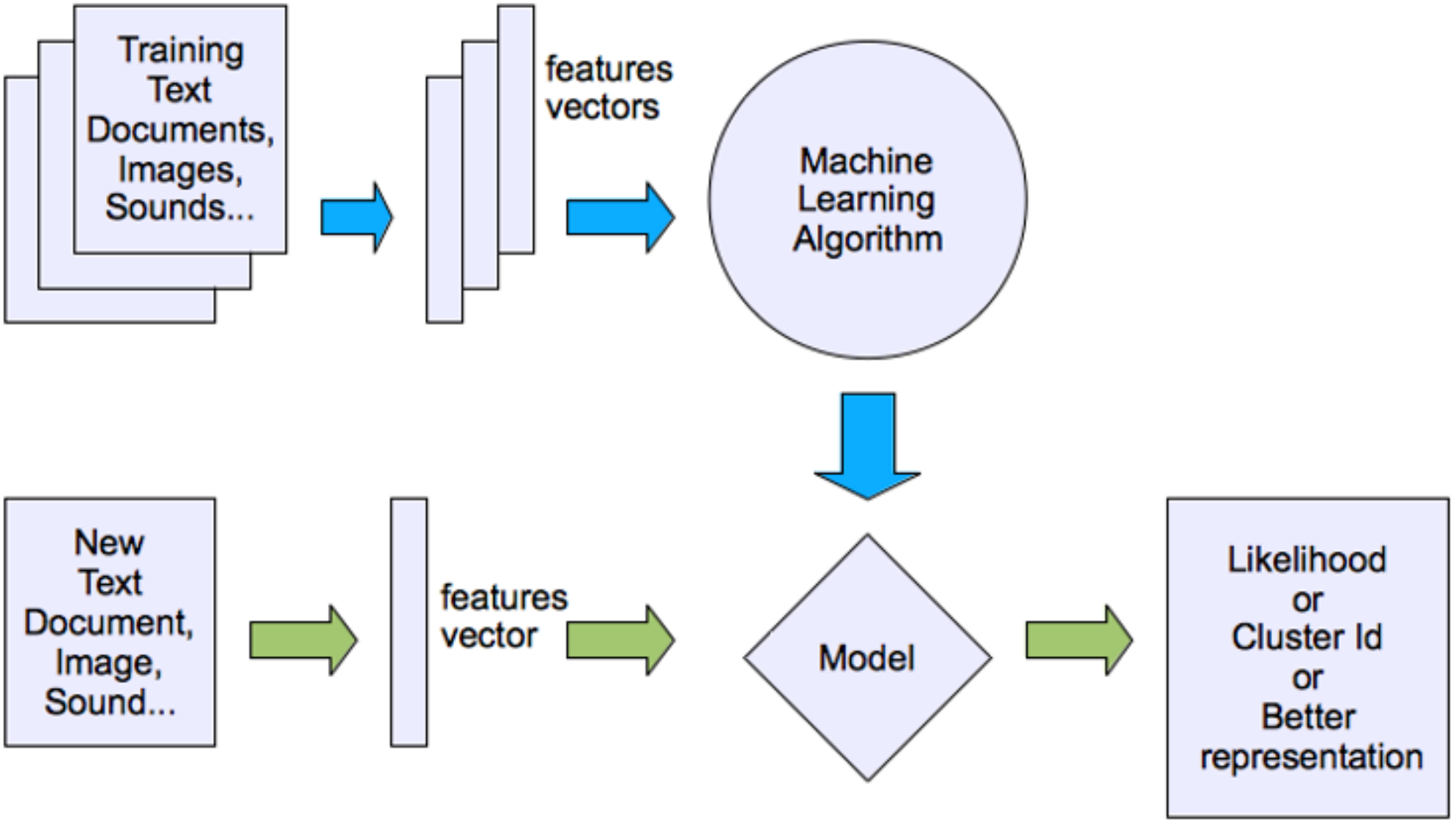
# Unsupervised Learning





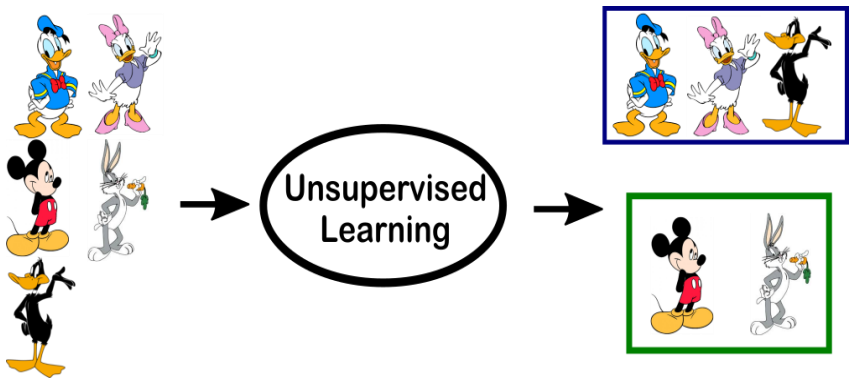
# Semi-Supervised Learning

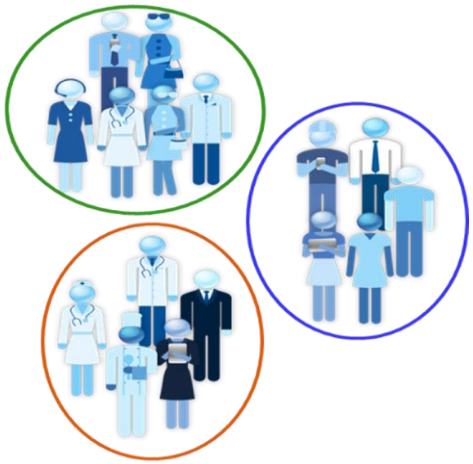


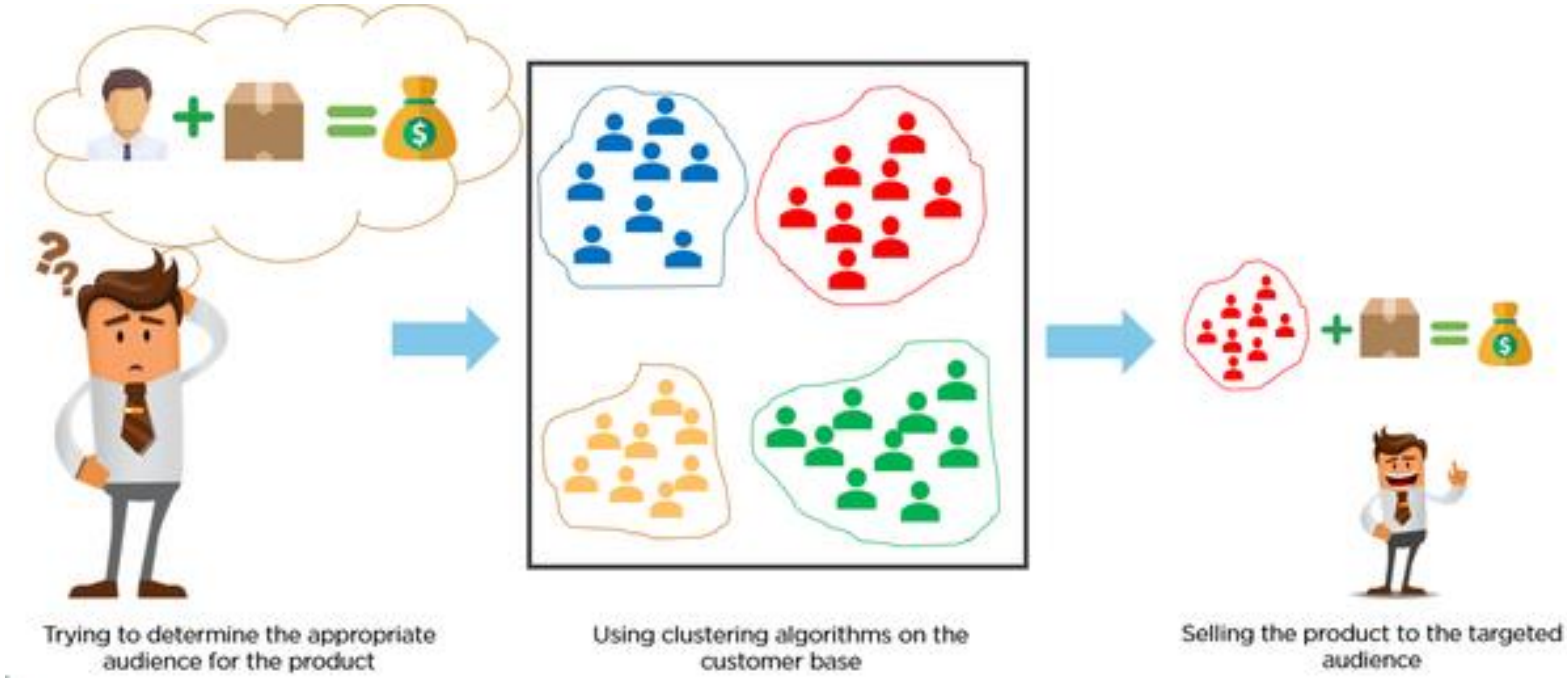


# Clustering





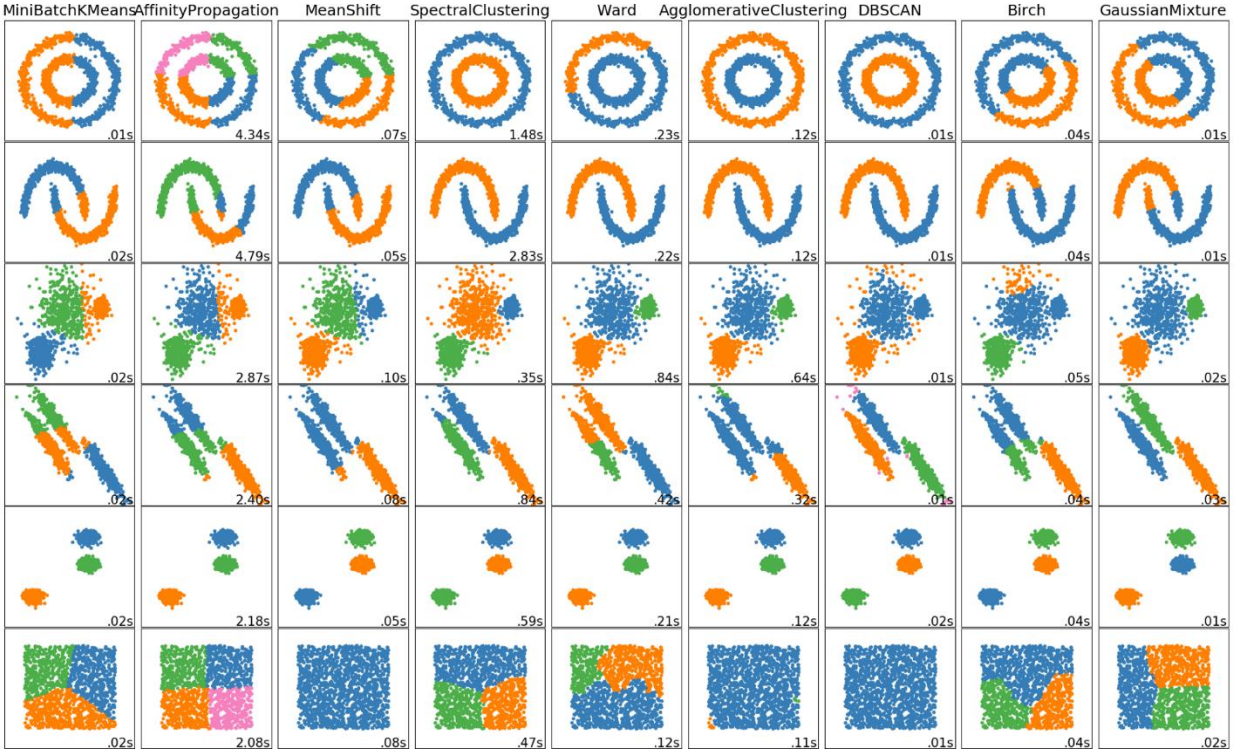





# Clustering Algorithms



- Clustering approaches group the data based on certain similarity of features.
- Clustering methods generally use centroid-based and hierarchal kind of modelling approaches.
- Most of the methods tend to use the inherent structures in the data to best organize the data into groups, considering the maximum similarity in the data.
- Well known clustering algorithms are,
  - k-Means
  - k-Medians
  - Expectation Maximisation (EM)
  - Hierarchical Clustering



# What is clustering?



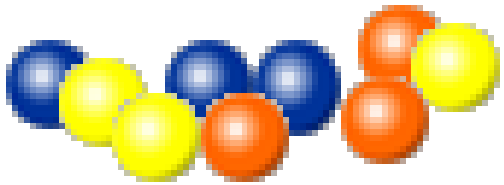
- Clustering: the process of grouping a set of objects into classes of similar objects
- A way of grouping together data samples that are ***similar*** in some way - according to some criteria that you pick
- A form of ***unsupervised learning*** – learning from raw data, as opposed to supervised data where a classification of examples is given
- A common and important task that finds many applications in IR and other places
  - Documents within a cluster should be similar.
  - Documents from different clusters should be dissimilar.
- So, it's a method of ***data exploration*** – a way of looking for patterns or structure in the data that are of interest

# What is clustering?

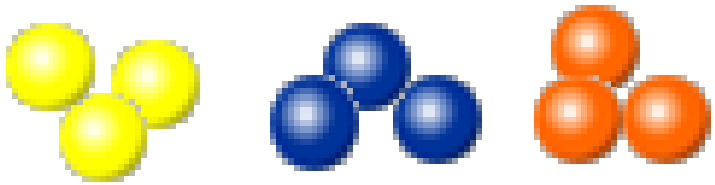


Clustering of data is a method by which large sets of data is grouped into clusters of smaller sets of similar data.

The example below demonstrates the clustering of balls of same colour. There are a total of 10 balls which are of three different colours. We are interested in clustering of balls of the three different colours into three different groups.



The balls of same colour are clustered into a group as shown below :



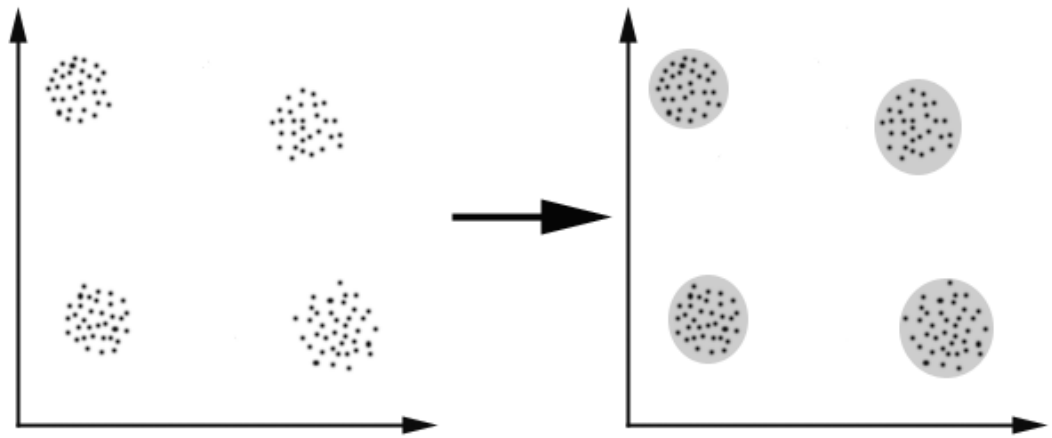
Thus, we see clustering means grouping of data or dividing a large data set into smaller data sets of some similarity.

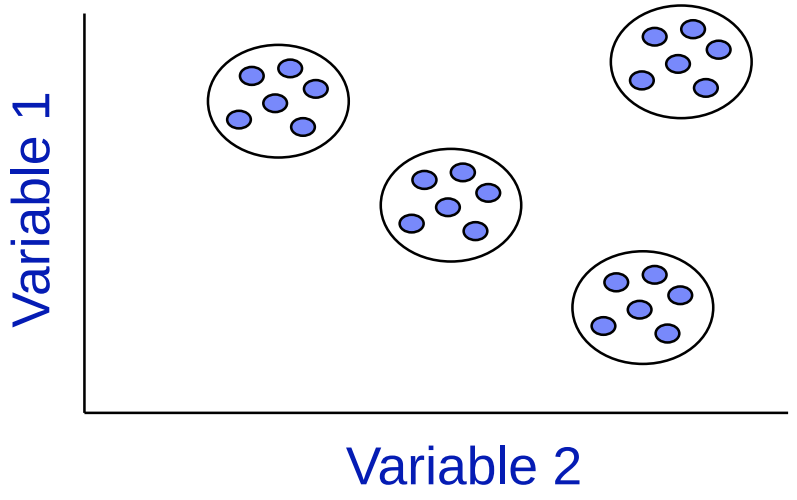
# A data set with clear cluster structure



Find K clusters (or a classification that consists of K clusters) so that the objects of one cluster are similar to each other whereas objects of different clusters are dissimilar. (Bacher 1996)

An Ideal Clustering Situation

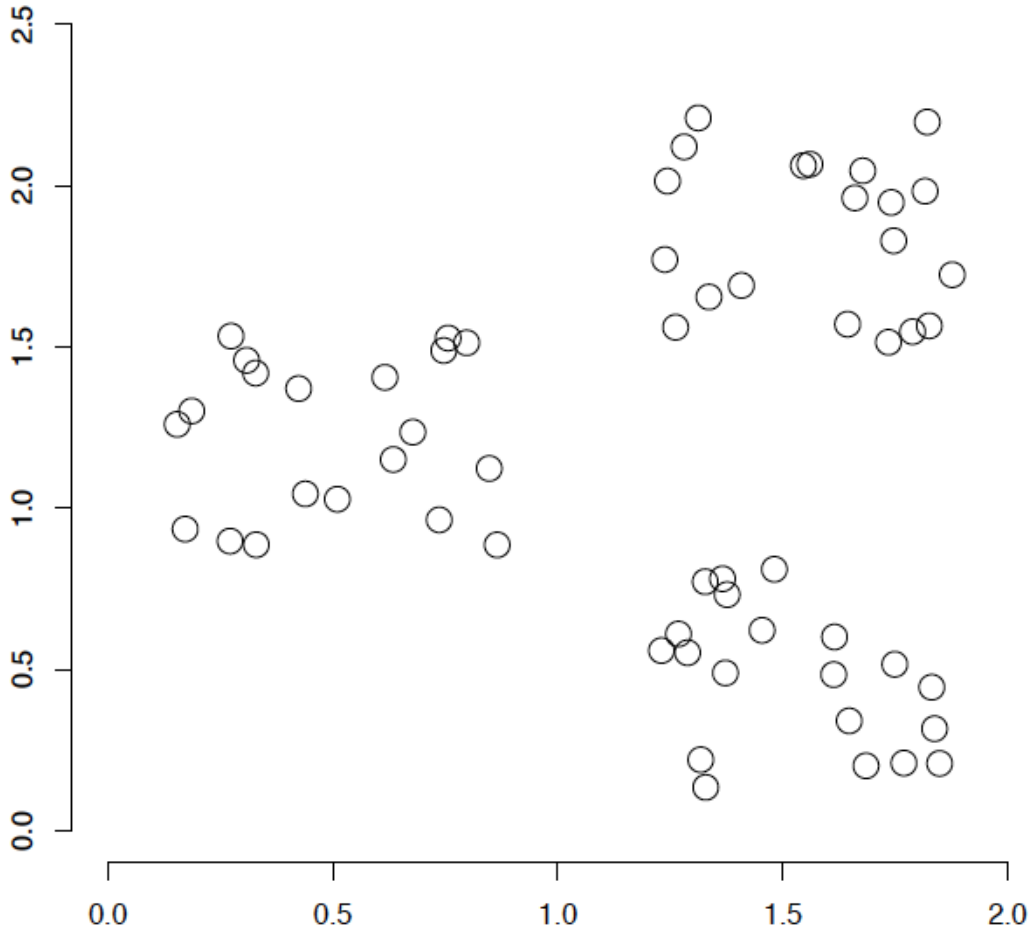




# A data set with clear cluster structure

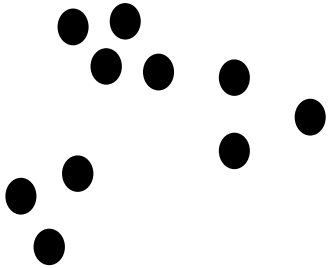


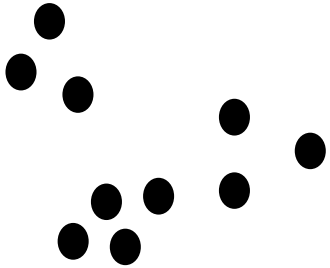
- How would you design an algorithm for finding the three clusters in this case?



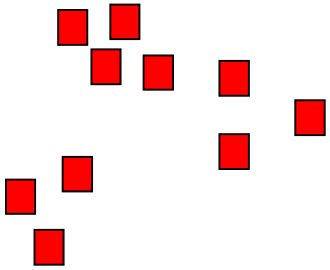
# Notion of a cluster can be ambiguous

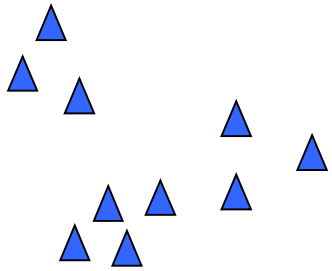




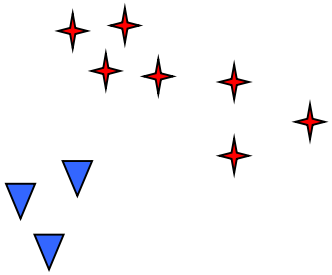


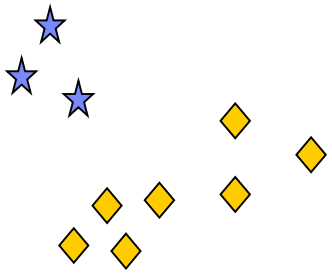
How many clusters?



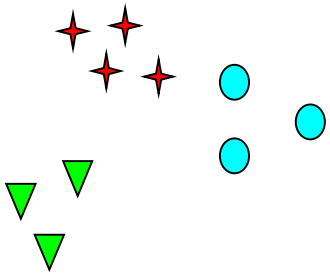


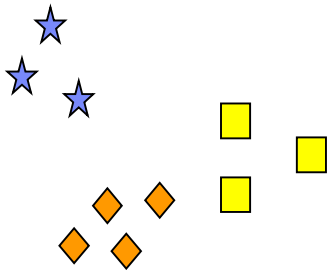
Two Clusters





Four Clusters



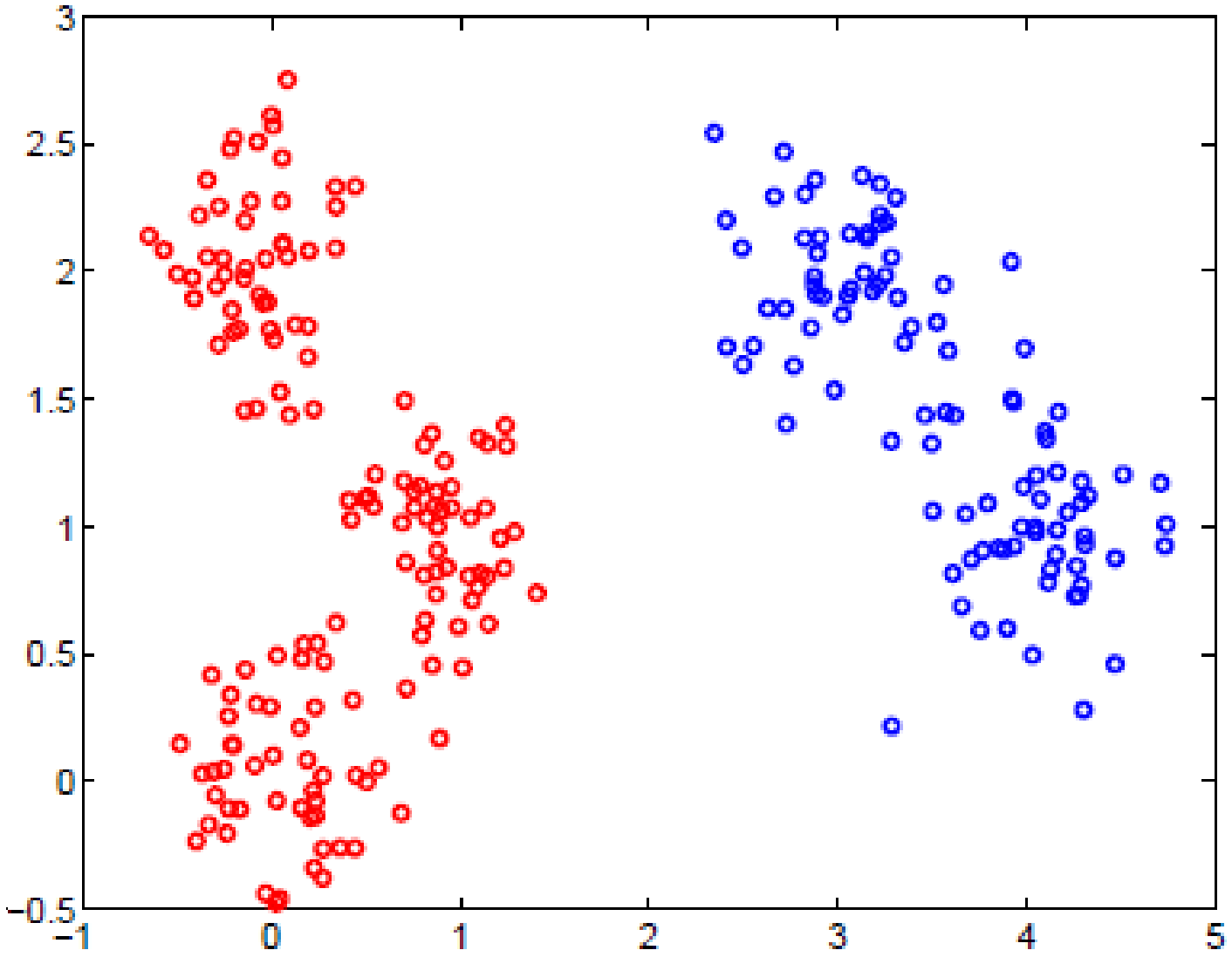


Six Clusters

# Illustration – 1



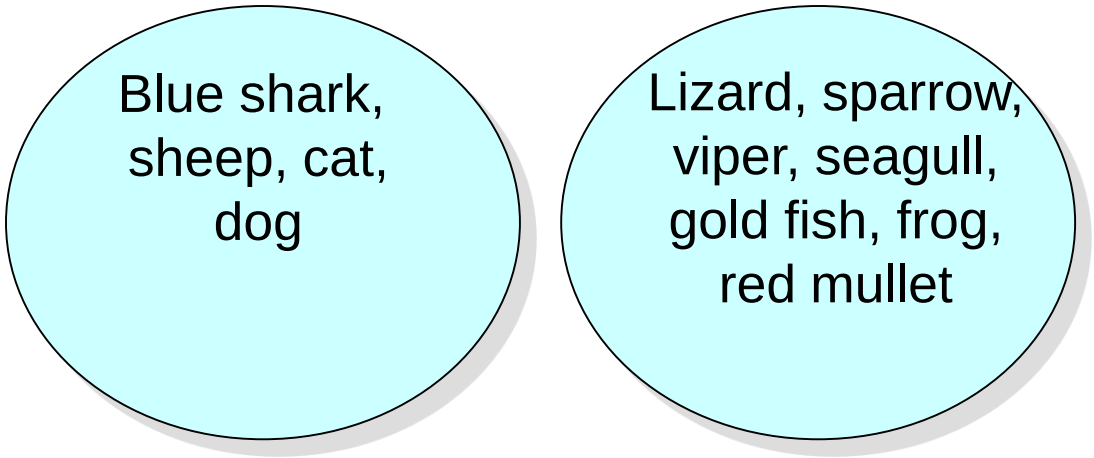
- Illustrative Example: How many clusters?



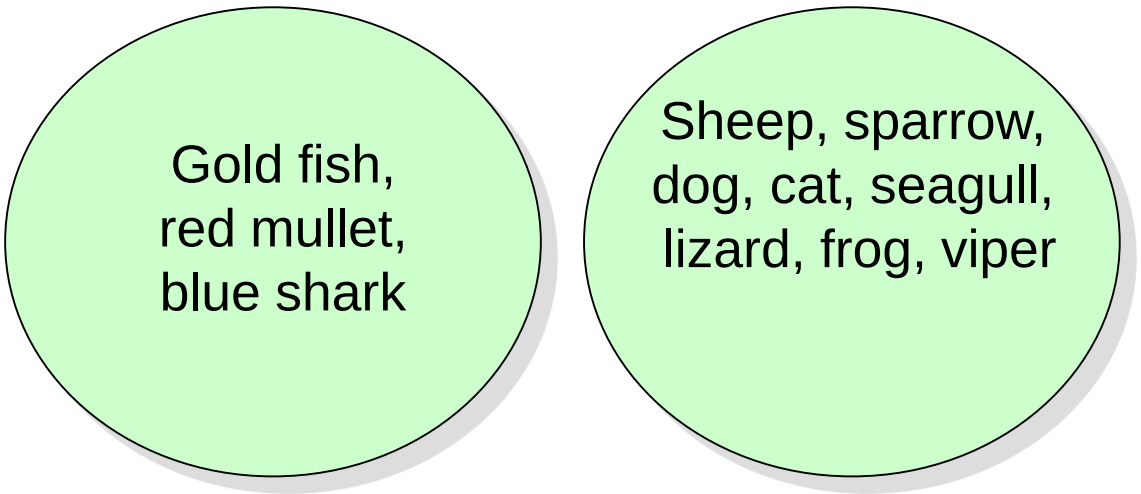
# Illustration – 2



- Illustrative Example 2: are they in the same cluster?



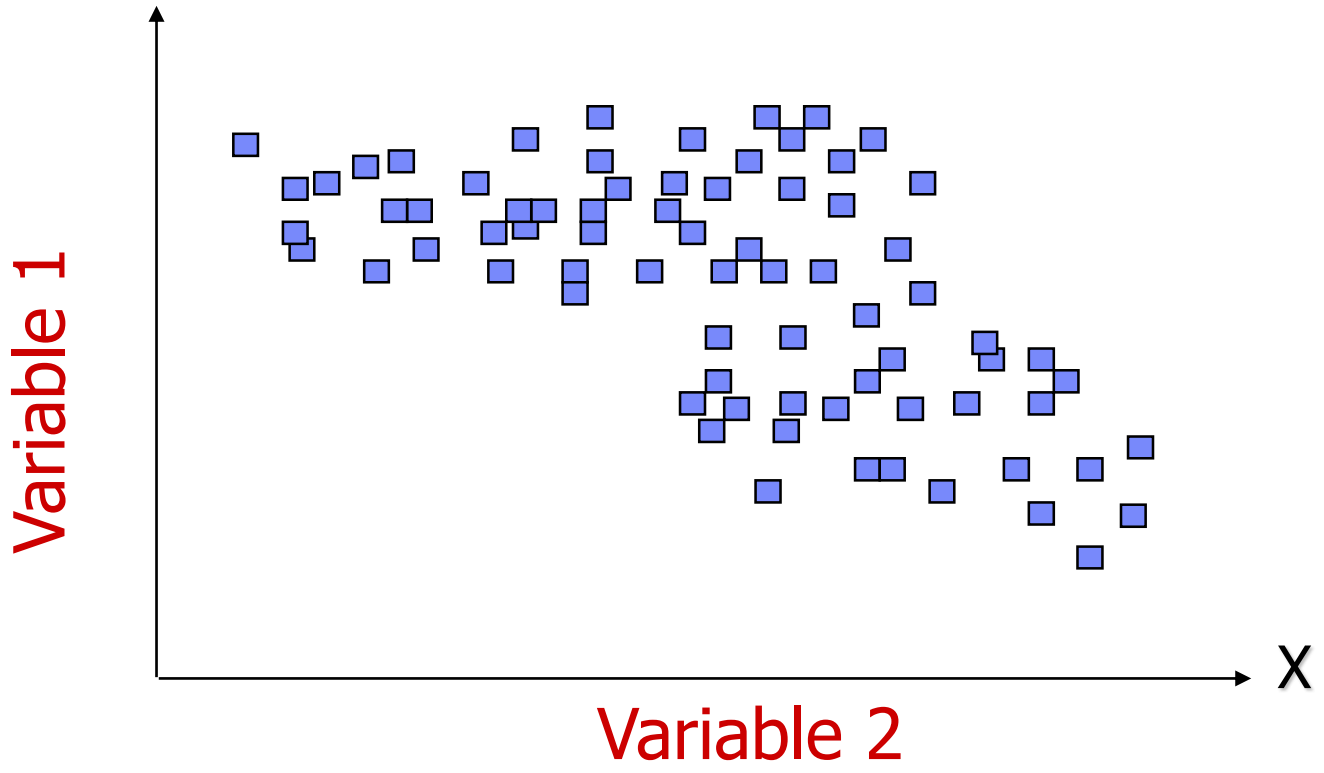

1. Two clusters
2. Clustering criterion: How animals bear their progeny




1. Two clusters
2. Clustering criterion: Existence of lungs

# More common clustering situation





# Good clustering and its goals



- **Internal criterion:** A good clustering will produce high quality clusters in which:
  - the intra-class (that is, intra-cluster) similarity is high
  - the inter-class similarity is low
  - The measured quality of a clustering depends on both the document representation and the similarity measure used
- **Goals of clustering**
  - Determine the intrinsic grouping in a set of unlabeled data.
  - All clustering algorithms will produce clusters, regardless of whether the data contains them
  - There is no golden standard, depends on goal:
    - data reduction; “natural clusters”; “useful” clusters; outlier detection

# Statistics Associated with Cluster Analysis



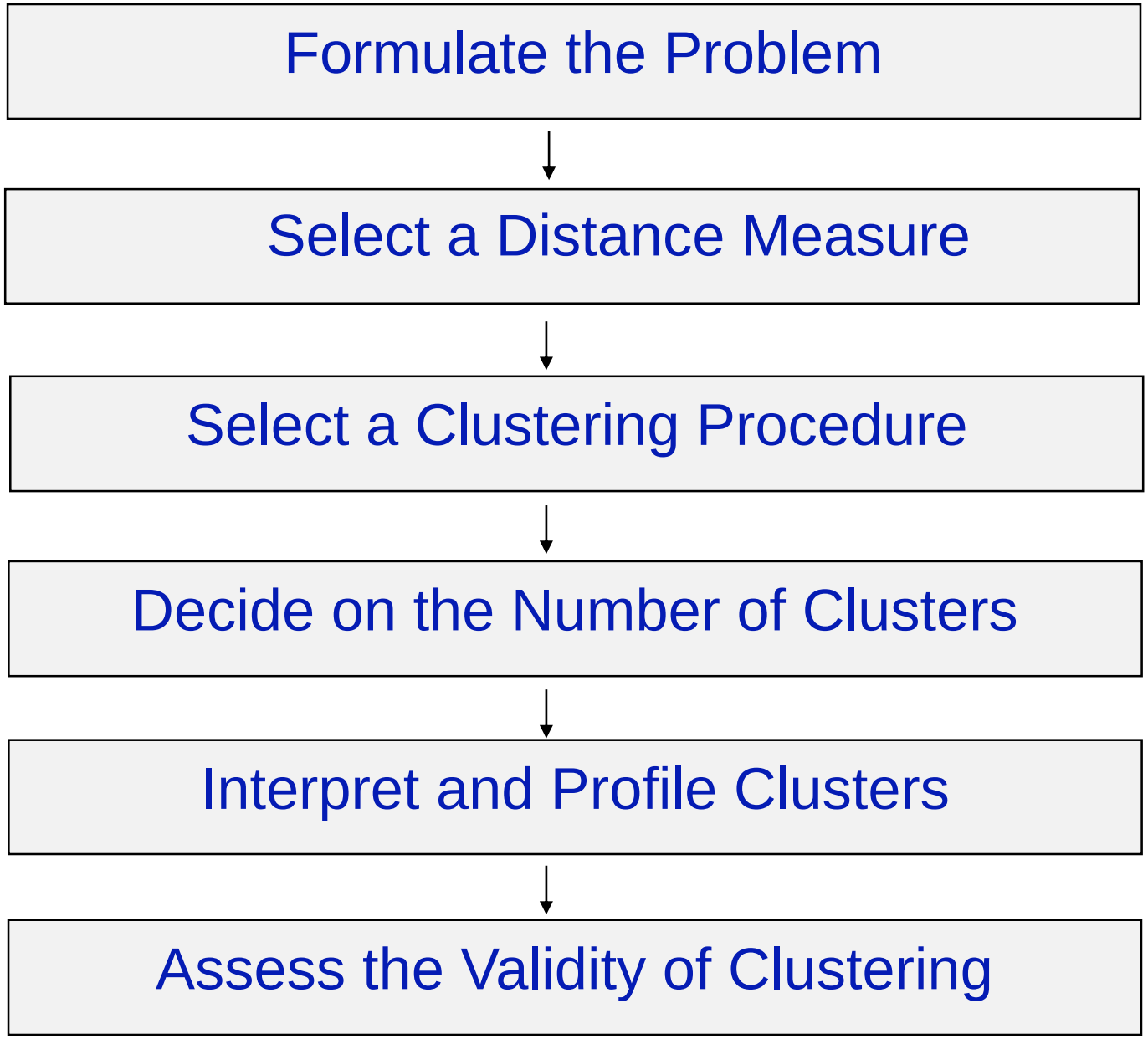
- **Agglomeration schedule**: Gives information on the objects or cases being combined at each stage of a hierarchical clustering process.
- **Cluster centroid**: Mean values of the variables for all the cases in a particular cluster.
- **Cluster centers**: Initial starting points in nonhierarchical clustering. Clusters are built around these centers, or *seeds*.
- **Cluster membership**: Indicates the cluster to which each object or case belongs.
- **Dendrogram (**A *tree graph):* A graphical device for displaying clustering results.

  -Vertical lines represent clusters that are joined together.

  -The position of the line on the scale indicates distances at which clusters were joined.
- **Distances between cluster centers:** These distances indicate how separated the individual pairs of clusters are. Clusters that are widely separated are distinct, and therefore desirable.
- **Icicle diagram:** Another type of graphical display of clustering results.

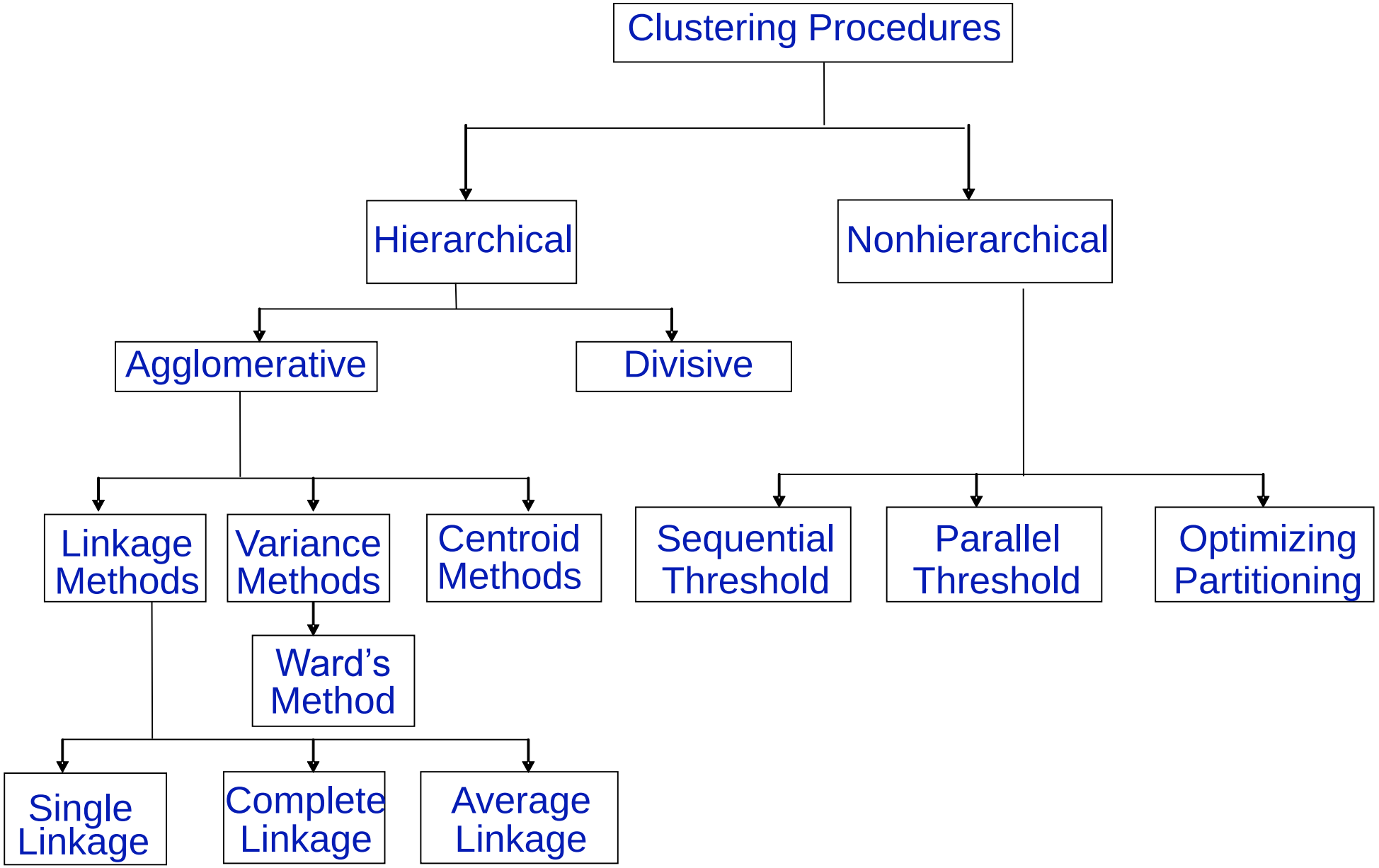
# Conducting Cluster Analysis





# Classification of Clustering Procedures





# General Applications of Clustering



- Pattern Recognition
- Spatial Data Analysis
  - create thematic maps in GIS by clustering feature spaces
  - detect spatial clusters and explain them in spatial data mining
- Image Processing
- Economic Science (especially market research)
- WWW
  - Document classification
  - Cluster Weblog data to discover groups of similar access patterns

# Applications



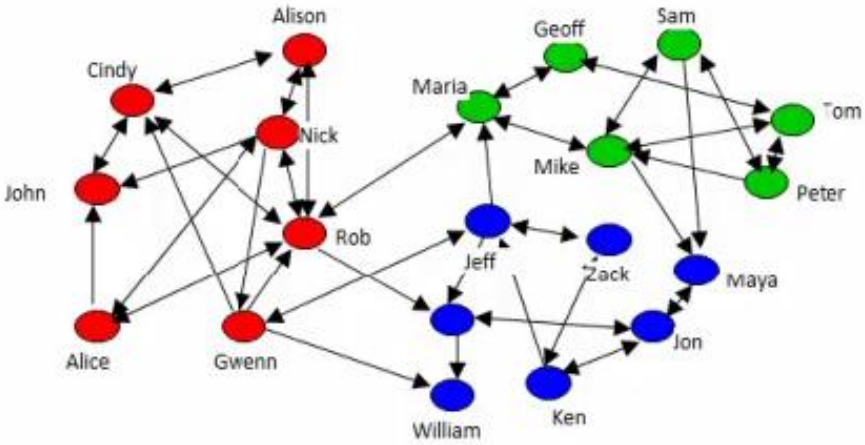
- A technique demanded by many real world tasks
  - Bank/Internet Security: fraud/spam pattern discovery
  - Biology: taxonomy of living things such as kingdom, phylum, class, order, family, genus and species
  - City-planning: Identifying groups of houses according to their house type, value, and geographical location
  - Climate change: understanding earth climate, find patterns of atmospheric and ocean
  - Earth-quake studies: Observed earth quake epicenters should be clustered along continent faults
  - Finance: stock clustering analysis to uncover correlation underlying shares
  - Image Compression/segmentation: coherent pixels grouped
  - Information retrieval/organization: Google search, topic-based news
  - Land use: Identification of areas of similar land use in earth observation database
  - Marketing: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
  - Social network mining: special interest group automatic discovery

# Applications

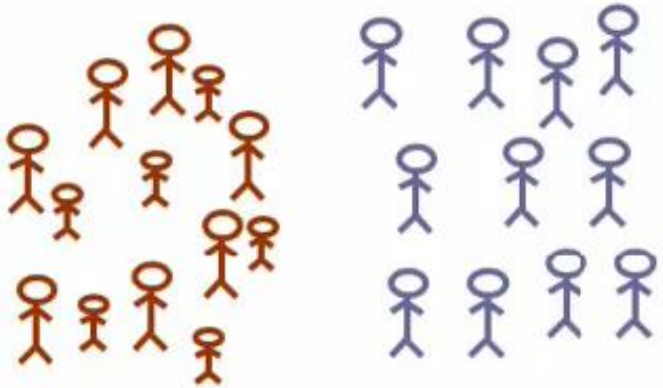


- Real Applications: Emerging Applications

Organize computing clusters




Social network analysis



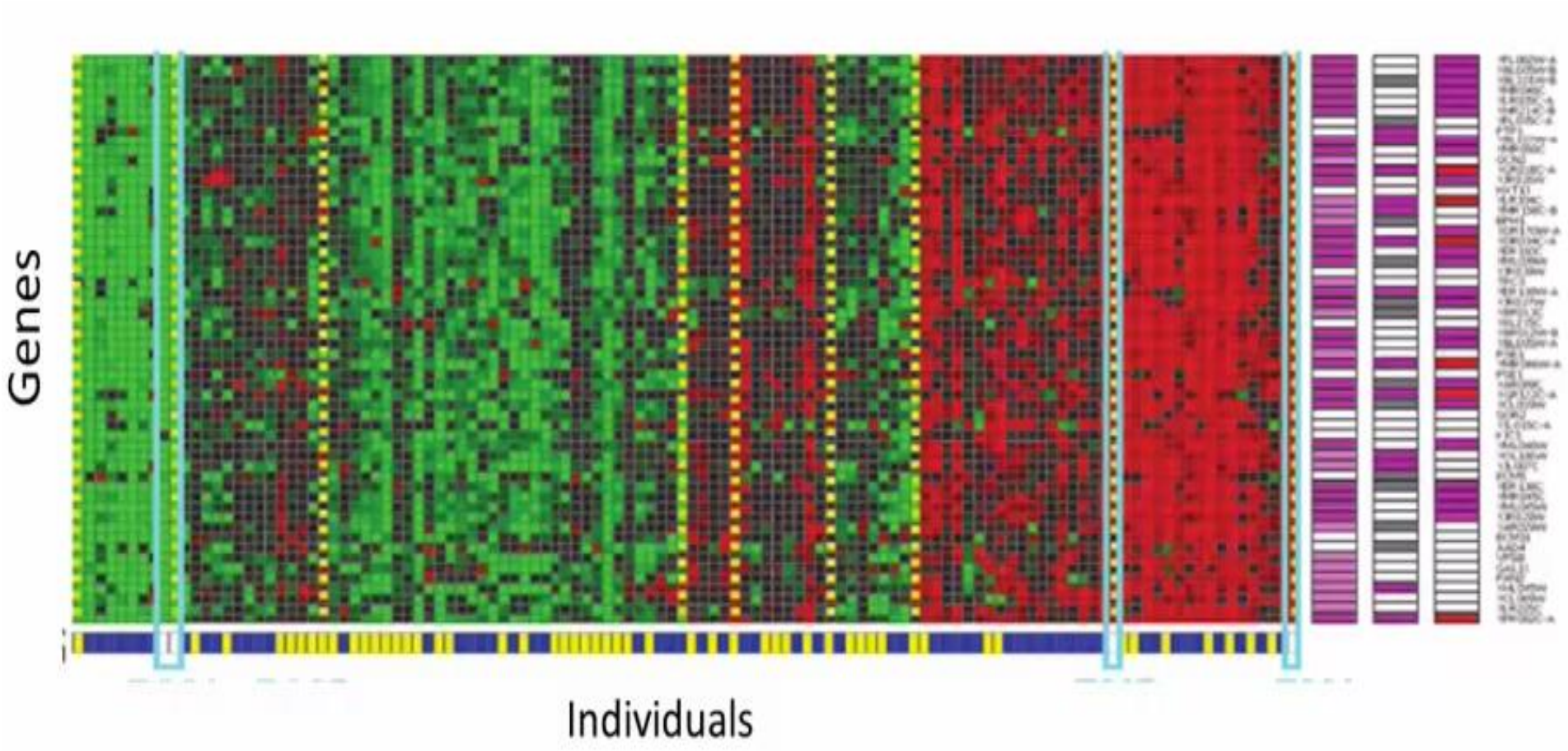
Market segmentation


Astronomical data analysis

# Applications



- Real Applications: Genetics Analysis



# Applications of clustering in IR

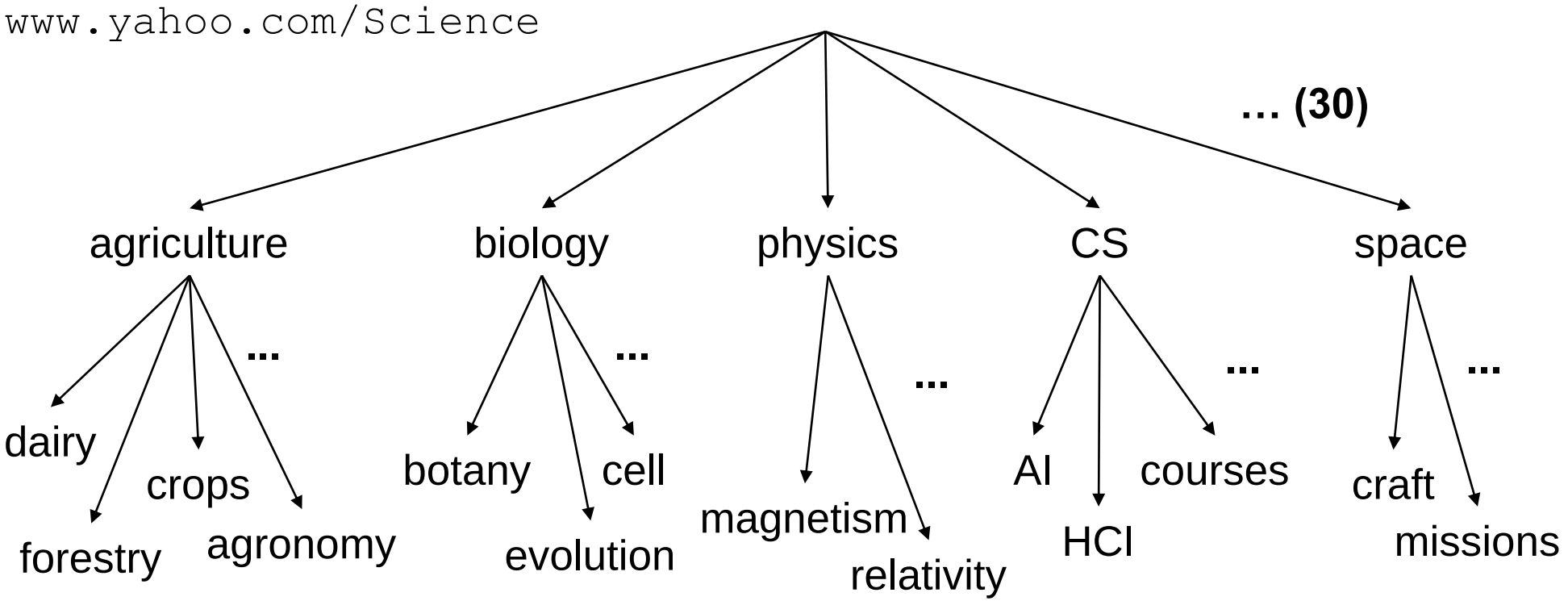


- Whole corpus analysis/navigation
  - Better user interface: search without typing
- For improving recall in search applications
  - Better search results (like pseudo RF)
- For better navigation of search results
  - Effective “user recall” will be higher
- For speeding up vector space retrieval
  - Cluster-based retrieval gives faster search

# Yahoo! Hierarchy

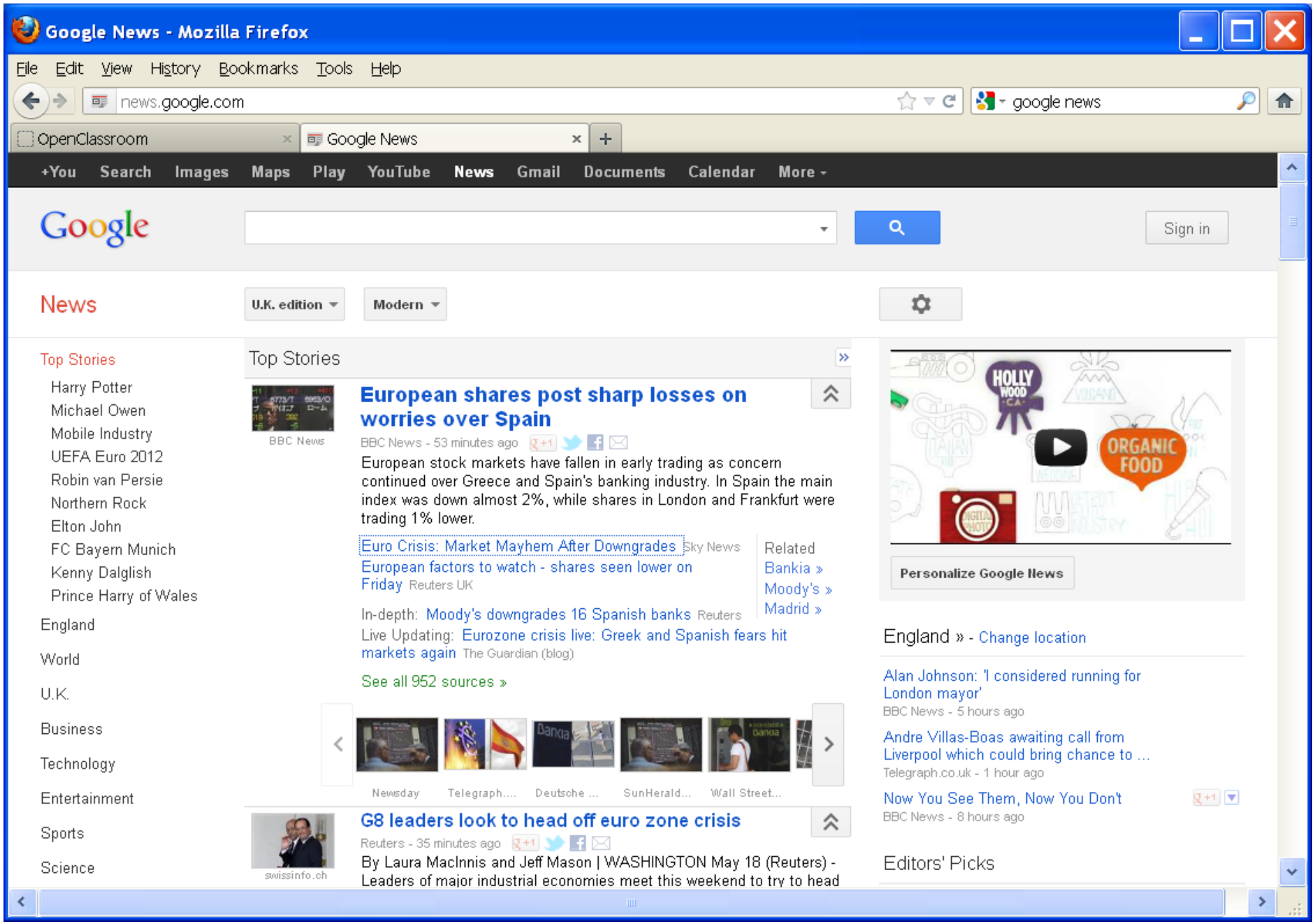
## (*isn't* clustering but *is* the kind of output you want from clustering)





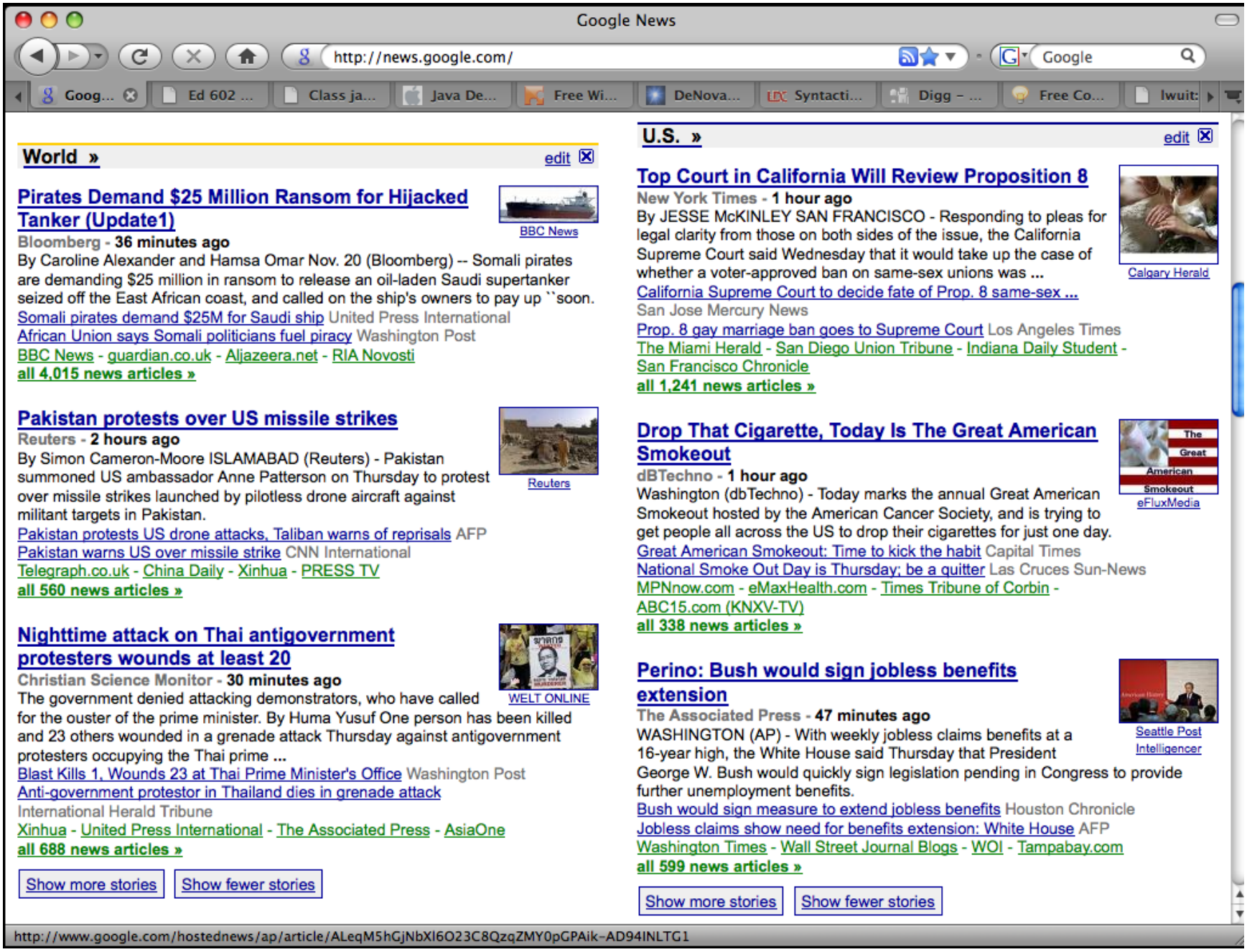
# Google News




Google News - Mozilla Firefox
File Edit View History Bookmarks Tools Help
news.google.com
google news
OpenClassroom
Google News
+You Search Images Maps Play YouTube News Gmail Documents Calendar More
Google
Sign in
News
U.K. edition
Modern
Top Stories
Harry Potter
Michael Owen
Mobile Industry
UEFA Euro 2012
Robin van Persie
Northern Rock
Elton John
FC Bayern Munich
Kenny Dalglish
Prince Harry of Wales
England
World
U.K.
Business
Technology
Entertainment
Sports
Science
Top Stories
European shares post sharp losses on worries over Spain
BBC News
BBC News - 53 minutes ago
European stock markets have fallen in early trading as concern continued over Greece and Spain's banking industry. In Spain the main index was down almost 2%, while shares in London and Frankfurt were trading 1% lower.
Euro Crisis: Market Mayhem After Downgrades Sky News
European factors to watch - shares seen lower on Friday Reuters UK
In-depth: Moody's downgrades 16 Spanish banks Reuters
Live Updating: Eurozone crisis live: Greek and Spanish fears hit markets again The Guardian (blog)
Related
Bankia »
Moody's »
Madrid »
See all 952 sources »
Newsday
Telegraph....
Deutsche ...
SunHerald...
Wall Street...
G8 leaders look to head off euro zone crisis
swissinfo.ch
Reuters - 35 minutes ago
By Laura MacInnis and Jeff Mason | WASHINGTON May 18 (Reuters) - Leaders of major industrial economies meet this weekend to try to head
HOLLY WOOD CA
ORGANIC FOOD
Personalize Google News
England » - Change location
Alan Johnson: 'I considered running for London mayor'
BBC News - 5 hours ago
Andre Villas-Boas awaiting call from Liverpool which could bring chance to ...
Telegraph.co.uk - 1 hour ago
Now You See Them, Now You Don't
BBC News - 8 hours ago
Editors' Picks

# Google News




Google News
http://news.google.com/
Google
World »
edit
Pirates Demand $25 Million Ransom for Hijacked Tanker (Update1)
BBC News
Bloomberg - 36 minutes ago
By Caroline Alexander and Hamsa Omar Nov. 20 (Bloomberg) -- Somali pirates are demanding $25 million in ransom to release an oil-laden Saudi supertanker seized off the East African coast, and called on the ship's owners to pay up ``soon.
Somali pirates demand $25M for Saudi ship United Press International
African Union says Somali politicians fuel piracy Washington Post
BBC News - guardian.co.uk - Aljazeera.net - RIA Novosti
all 4,015 news articles »
Pakistan protests over US missile strikes
Reuters
Reuters - 2 hours ago
By Simon Cameron-Moore ISLAMABAD (Reuters) - Pakistan summoned US ambassador Anne Patterson on Thursday to protest over missile strikes launched by pilotless drone aircraft against militant targets in Pakistan.
Pakistan protests US drone attacks, Taliban warns of reprisals AFP
Pakistan warns US over missile strike CNN International
Telegraph.co.uk - China Daily - Xinhua - PRESS TV
all 560 news articles »
Nighttime attack on Thai antigovernment protesters wounds at least 20
WELT ONLINE
Christian Science Monitor - 30 minutes ago
The government denied attacking demonstrators, who have called for the ouster of the prime minister. By Huma Yusuf One person has been killed and 23 others wounded in a grenade attack Thursday against antigovernment protesters occupying the Thai prime ...
Blast Kills 1, Wounds 23 at Thai Prime Minister's Office Washington Post
Anti-government protestor in Thailand dies in grenade attack International Herald Tribune
Xinhua - United Press International - The Associated Press - AsiaOne
all 688 news articles »
Show more stories
Show fewer stories
U.S. »
edit
Top Court in California Will Review Proposition 8
Calgary Herald
New York Times - 1 hour ago
By JESSE McKINLEY SAN FRANCISCO - Responding to pleas for legal clarity from those on both sides of the issue, the California Supreme Court said Wednesday that it would take up the case of whether a voter-approved ban on same-sex unions was ...
California Supreme Court to decide fate of Prop. 8 same-sex ... San Jose Mercury News
Prop. 8 gay marriage ban goes to Supreme Court Los Angeles Times
The Miami Herald - San Diego Union Tribune - Indiana Daily Student - San Francisco Chronicle
all 1,241 news articles »
Drop That Cigarette, Today Is The Great American Smokeout
The Great American Smokeout
eFluxMedia
dBTechno - 1 hour ago
Washington (dbTechno) - Today marks the annual Great American Smokeout hosted by the American Cancer Society, and is trying to get people all across the US to drop their cigarettes for just one day.
Great American Smokeout: Time to kick the habit Capital Times
National Smoke Out Day is Thursday; be a quitter Las Cruces Sun-News
MPNnow.com - eMaxHealth.com - Times Tribune of Corbin - ABC15.com (KNXV-TV)
all 338 news articles »
Perino: Bush would sign jobless benefits extension
Seattle Post Intelligencer
The Associated Press - 47 minutes ago
WASHINGTON (AP) - With weekly jobless claims benefits at a 16-year high, the White House said Thursday that President George W. Bush would quickly sign legislation pending in Congress to provide further unemployment benefits.
Bush would sign measure to extend jobless benefits Houston Chronicle
Jobless claims show need for benefits extension: White House AFP
Washington Times - Wall Street Journal Blogs - WOI - Tampabay.com
all 599 news articles »
Show more stories
Show fewer stories
http://www.google.com/hostednews/ap/article/ALeqM5hGjNbXI6O23C8QzqZMY0pGPAik-AD94INLTG1

# Scatter/Gather



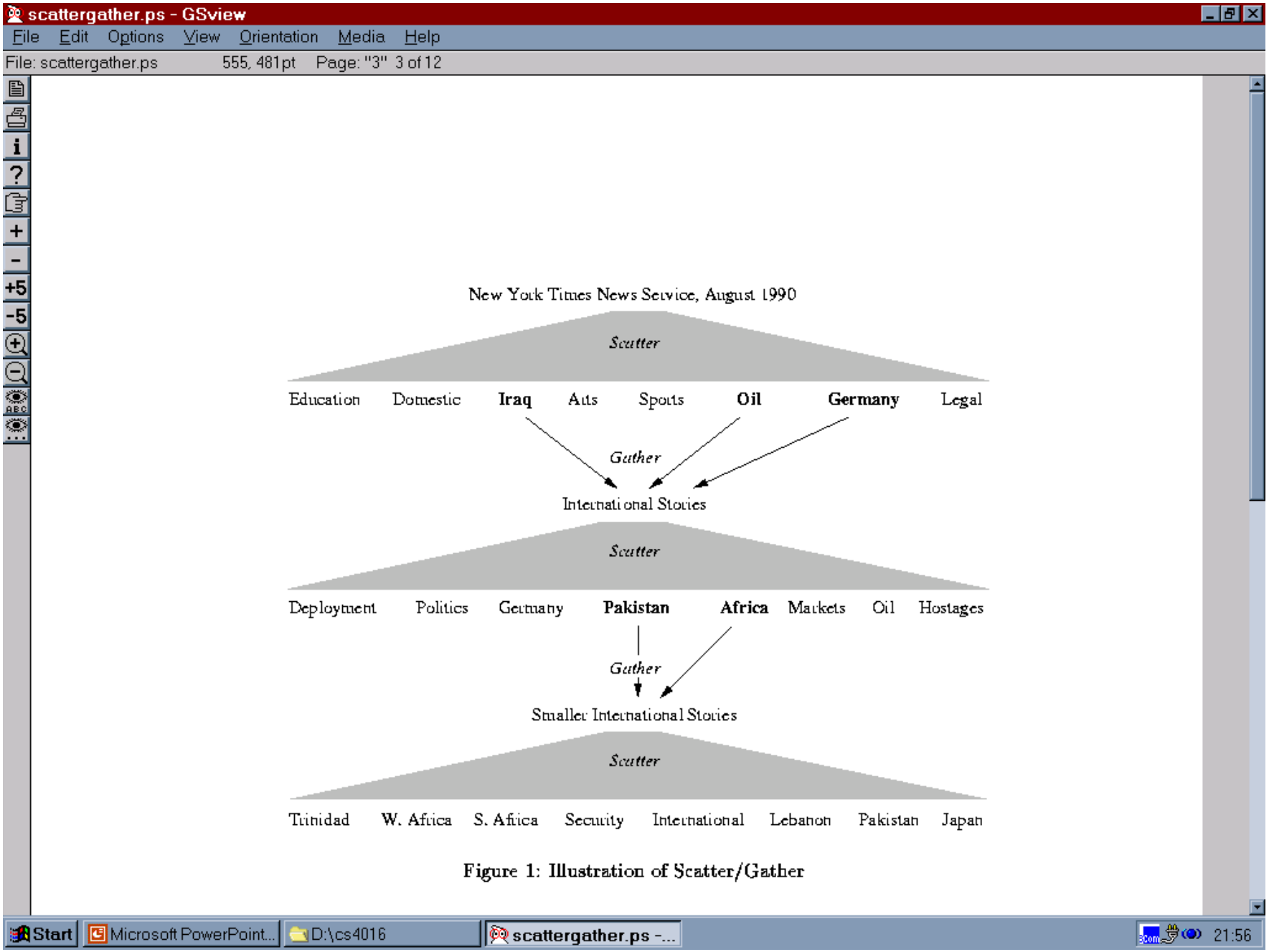
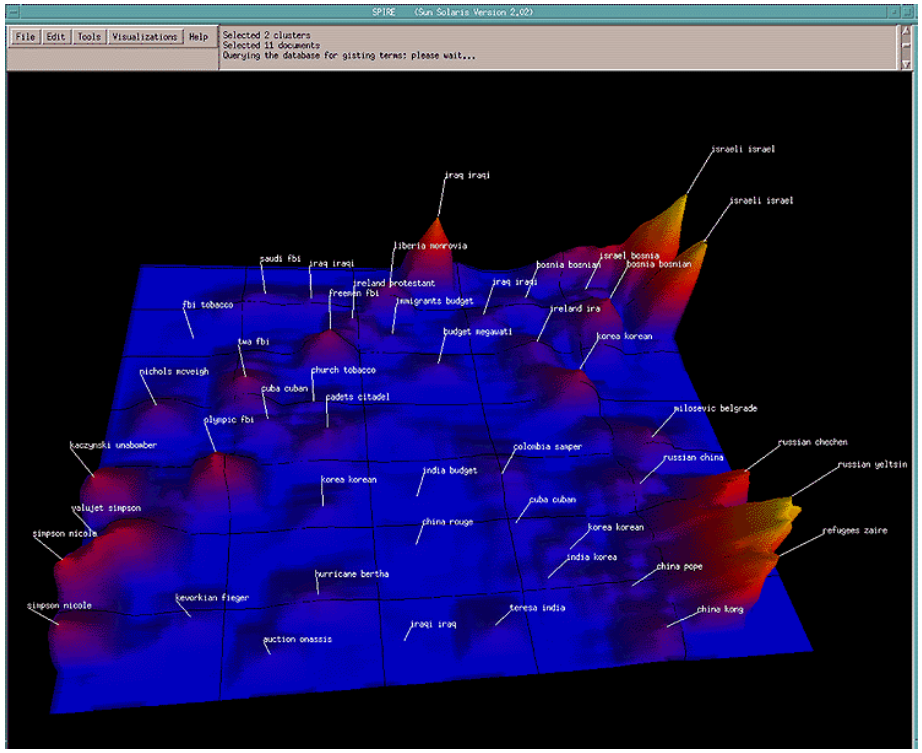


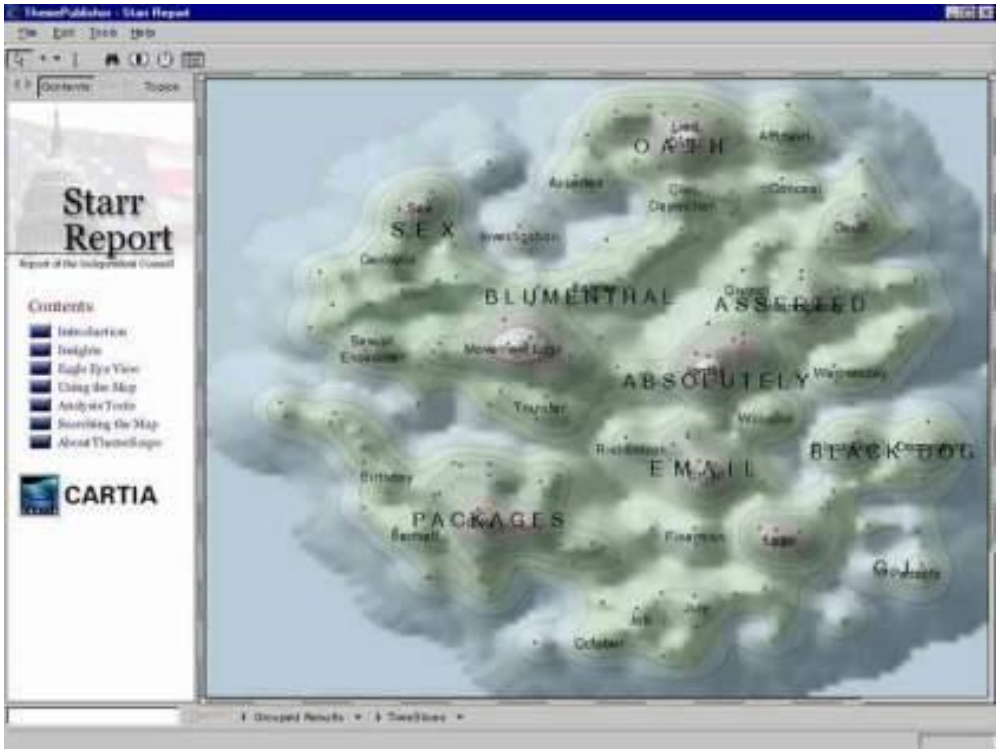

Figure 1: Illustration of Scatter/Gather

# Document Collection Visualization



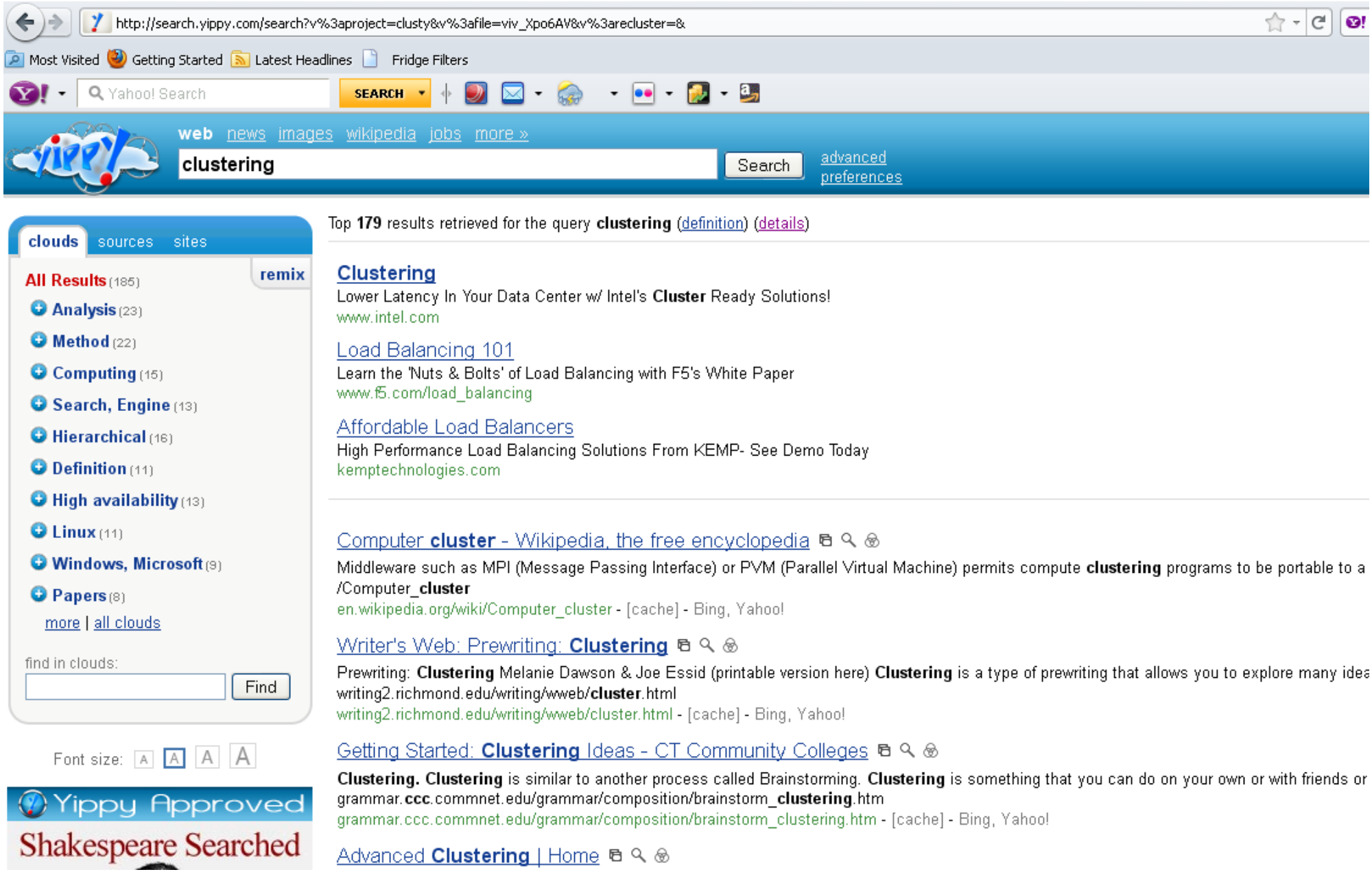
- Wise et al, “Visualizing the non-visual” PNNL
- ThemeScapes, Cartia
  - [Mountain height = cluster size]



# yippy.com – grouping search results





# Clustering as a Preprocessing Tool



- Summarization:
  - Preprocessing for regression, PCA, classification, and association analysis
- Compression:
  - Image processing: vector quantization
- Finding K-nearest Neighbors
  - Localizing search to one or a small number of clusters
- Outlier detection
  - Outliers are often viewed as those “far away” from any cluster

# Requirements for Clustering



- Scalability
- Ability to deal with different types of attributes
- Discovery of clusters with arbitrary shape
- Minimal domain knowledge required to determine input parameters
- Ability to deal with noise and outliers
- Insensitivity to order of input records
- Robustness wrt high dimensionality
- Incorporation of user-specified constraints
- Interpretability and usability

# Considerations for Cluster Analysis



- A good clustering method will produce high quality clusters
  - high intra-class similarity: cohesive within clusters
  - low inter-class similarity: distinctive between clusters
- The quality of a clustering method depends on
  - the similarity measure used by the method
  - its implementation, and
  - Its ability to discover some or all of the hidden patterns
- Precise definition of clustering quality is difficult
  - Application-dependent
  - Ultimately subjective
- Representation for clustering
  - Document representation: Vector space? Normalization?
    - Centroids aren't length normalized
  - Need a notion of similarity/distance
- How many clusters?
  - Fixed a priori?
  - Completely data driven?

# Considerations for Cluster Analysis



- Partitioning criteria
  - Single level vs. hierarchical partitioning (often, multi-level hierarchical partitioning is desirable)
- Separation of clusters
  - Exclusive (e.g., one customer belongs to only one region) vs. non-exclusive (e.g., one document may belong to more than one class)
- Similarity measure
  - Distance-based (e.g., Euclidian, road network, vector) vs. connectivity-based (e.g., density or contiguity)
- Clustering space
  - Full space (often when low dimensional) vs. subspaces (often in high-dimensional clustering)

# Hard vs. soft clustering



- Hard clustering: Each object belongs to exactly one cluster
  - More common and easier to do
- Soft clustering: An object can belong to more than one cluster.
  - Makes more sense for applications like creating browsable hierarchies
  - You may want to put a pair of sneakers in two clusters: (i) sports apparel and (ii) shoes
  - You can only do that with a soft clustering approach.

# Measure the Quality of Clustering



- Dissimilarity/Similarity metric
  - Similarity is expressed in terms of a distance function, typically metric: $d(i, j)$
  - The definitions of distance functions are usually rather different for interval-scaled, boolean, categorical, ordinal ratio, and vector variables
  - Weights should be associated with different variables based on applications and data semantics
- Quality of clustering:
  - There is usually a separate “quality” function that measures the “goodness” of a cluster.
  - It is hard to define “similar enough” or “good enough”
  - The answer is typically highly subjective

# Type of data in clustering analysis



- Interval-scaled variables:
- Binary variables:
- Nominal, ordinal, and ratio variables:
- Variables of mixed types:

# Interval-valued variables



- Standardize data
  - Calculate the mean absolute deviation:

$$s_f = \frac{1}{n}(|x_{1f} - m_f| + |x_{2f} - m_f| + ... + |x_{nf} - m_f|)$$

where $$m_f = \frac{1}{n}(x_{1f} + x_{2f} + ... + x_{nf})$$

  - Calculate the standardized measurement (*z-score*)

$$z_{if} = \frac{x_{if} - m_f}{s_f}$$

- Using mean absolute deviation is more robust than using standard deviation

# Similarity and Dissimilarity Between Objects



- Distances are normally used to measure the similarity or dissimilarity between two data objects
- Some popular ones include: *Minkowski distance*:

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

where $i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ and $j = (x_{j1}, x_{j2}, \ldots, x_{jp})$ are two $p$-dimensional data objects, and $q$ is a positive integer

- If $q = 1$, $d$ is Manhattan distance

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

# Similarity and Dissimilarity Between Objects (Cont.)



- *If q = 2*, *d* is Euclidean distance:
- Properties
  - *d(i,j)* $\geq 0$
  - *d(i,i)* = 0
  - *d(i,j)* = *d(j,i)*
  - *d(i,j)* $\leq$ *d(i,k)* + *d(k,j)*

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

- Also, one can use weighted distance, parametric Pearson product moment correlation, or other dissimilarity measures

# Binary Variables



- A contingency table for binary data

| | | Object $j$ | | |
|---|---|---|---|---|
| | | 1 | 0 | *sum* |
| Object $i$ | 1 | $a$ | $b$ | $a+b$ |
| | 0 | $c$ | $d$ | $c+d$ |
| | *sum* | $a+c$ | $b+d$ | $p$ |

- Simple matching coefficient (invariant, if the binary variable is *symmetric*):
- Jaccard coefficient (non-invariant if the binary variable is *asymmetric*):

$$d(i,j) = \frac{b+c}{a+b+c+d} \qquad d(i,j) = \frac{b+c}{a+b+c}$$

# Dissimilarity between Binary Variables



- Example

| Name | Gender | Fever | Cough | Test-1 | Test-2 | Test-3 | Test-4 |
|---|---|---|---|---|---|---|---|
| Jack | M | Y | N | P | N | N | N |
| Mary | F | Y | N | P | N | P | N |
| Jim | M | Y | P | N | N | N | N |

  - gender is a symmetric attribute
  - the remaining attributes are asymmetric binary
  - let the values Y and P be set to 1, and the value N be set to 0

$$d(jack, mary) = \frac{0+1}{2+0+1} = 0.33$$

$$d(jack, jim) = \frac{1+1}{1+1+1} = 0.67$$

$$d(jim, mary) = \frac{1+2}{1+1+2} = 0.75$$

# Nominal Variables



- A generalization of the binary variable in that it can take more than 2 states, e.g., red, yellow, blue, green
- Method 1: Simple matching
  - *m*: # of matches, *p*: total # of variables

$$d(i, j) = \frac{p - m}{p}$$

- Method 2: use a large number of binary variables
  - creating a new binary variable for each of the *M* nominal states

# Ordinal Variables



- An ordinal variable can be discrete or continuous
- Order is important, e.g., rank
- Can be treated like interval-scaled
  - replace $x_{if}$ by their rank
  - map the range of each variable onto [0, 1] by replacing *i*-th object in the *f*-th variable by $r_{if} \in \{1, \ldots, M_f\}$
  - compute the dissimilarity using methods for interval-scaled variables

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

# Ratio-Scaled Variables



- Ratio-scaled variable: a positive measurement on a nonlinear scale, approximately at exponential scale,such as $Ae^{Bt}$ or $Ae^{-Bt}$
- Methods:
  - treat them like interval-scaled variables—*not a good choice!* (why?—the scale can be distorted)
  - apply logarithmic transformation

$$y_{if} = log(x_{if})$$

  - treat them as continuous ordinal data treat their rank as interval-scaled

# Variables of Mixed Types



- A database may contain all the six types of variables
  - symmetric binary, asymmetric binary, nominal, ordinal, interval and ratio
- One may use a weighted formula to combine their effects

$$d(i,j) = \frac{\sum_{f=1}^{p} \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^{p} \delta_{ij}^{(f)}}$$

  - *f* is binary or nominal:
    $d_{ij}^{(f)} = 0$ if $x_{if} = x_{jf}$ , or $d_{ij}^{(f)} = 1$
  - *f* is interval-based: use the normalized distance
  - *f* is ordinal or ratio-scaled
    - compute ranks $r_{if}$ and
    - and treat $z_{if}$ as interval-scaled

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

# Major Clustering Approaches

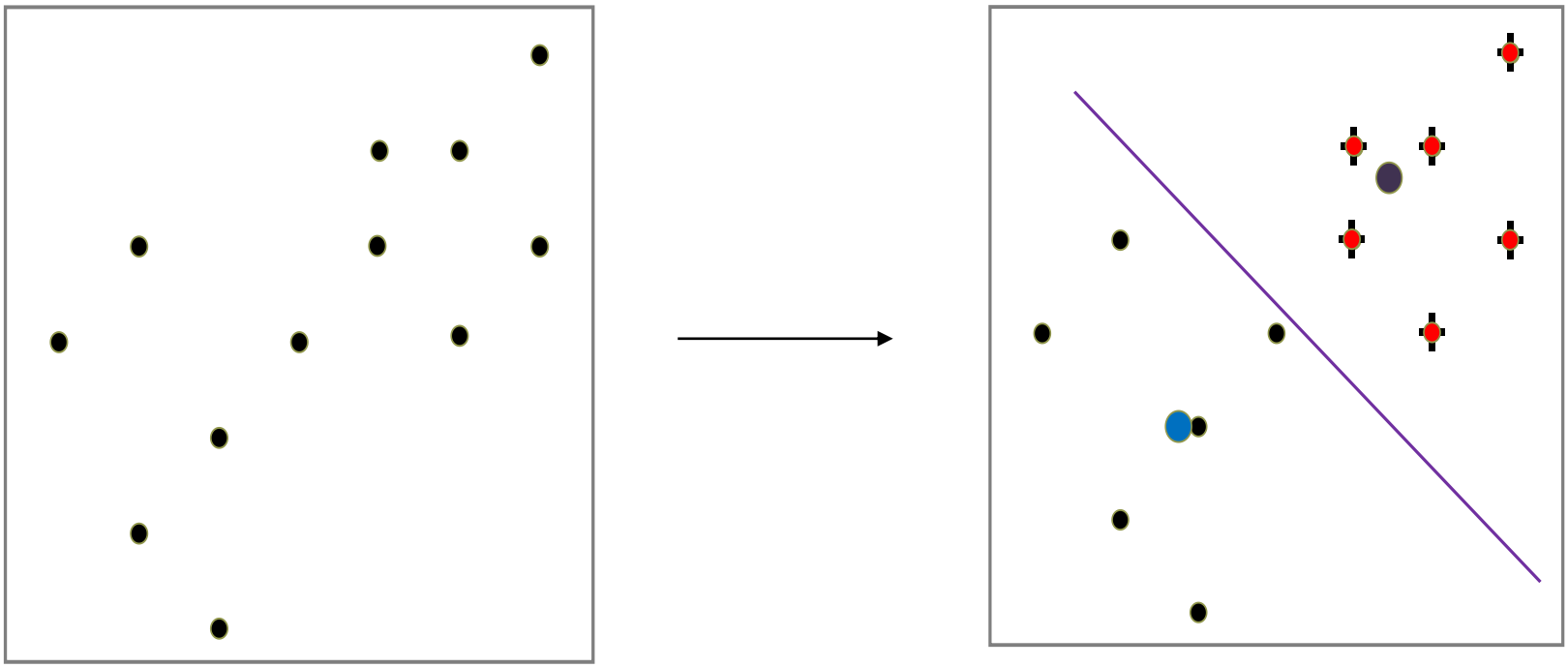


- Partitioning approach
- Hierarchical approach
- Density-based approach
- Grid-based approach
- Model-based
- Frequent pattern-based
- User-guided or constraint-based
- Link-based clustering

# Major Clustering Approaches

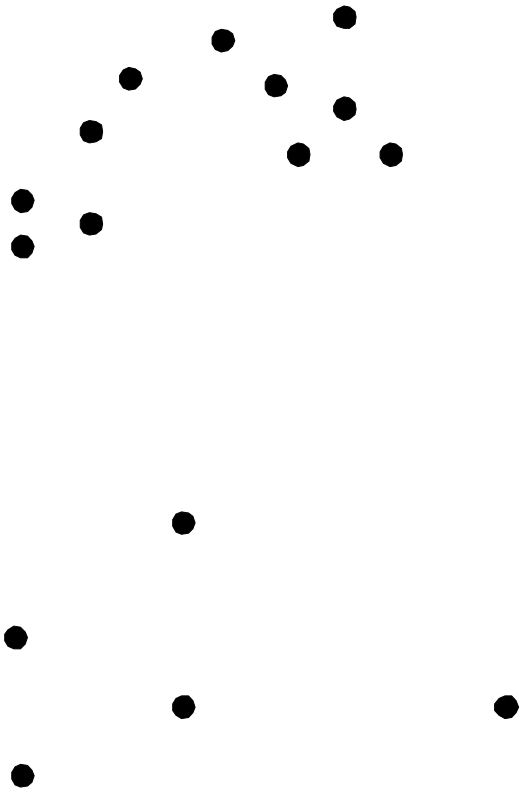


- Partitioning Methodology
  - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square distance cost, …
  - Typical methods: **K-means**, K-medoids, CLARANS, ……

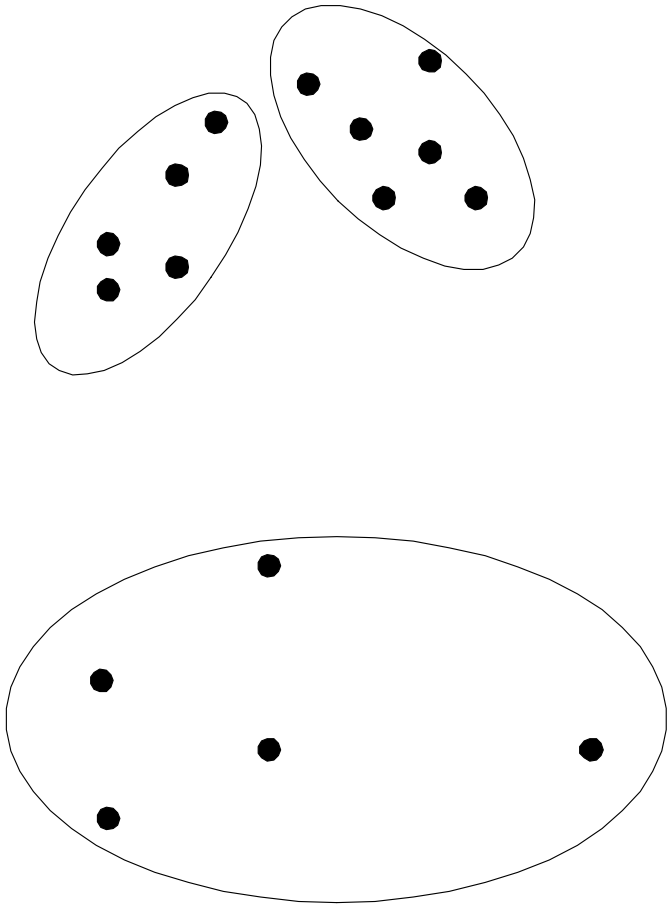


# Partitional Clustering





Original Points

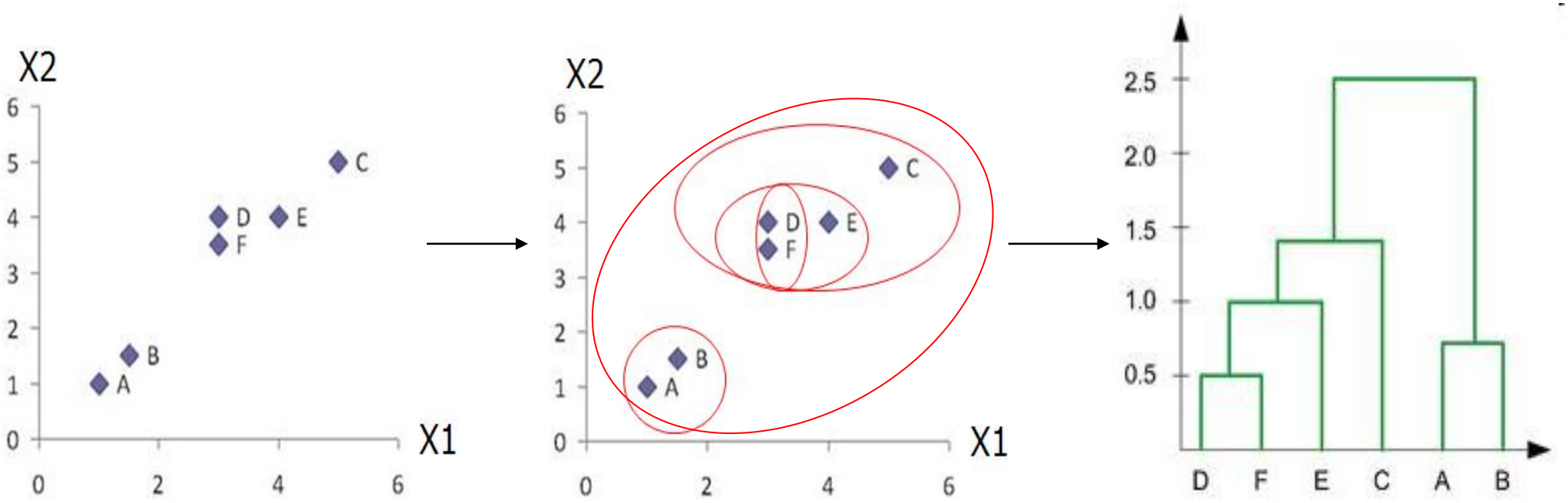


A Partitional Clustering

# Major Clustering Approaches

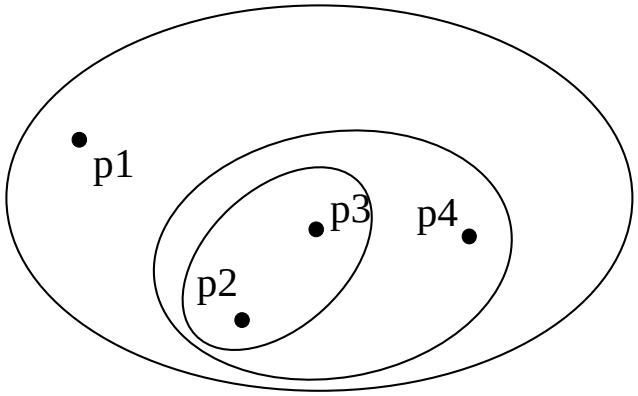


- Hierarchical Methodology
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion
  - Typical methods: **Agglomerative**, Diana, Agnes, BIRCH, ROCK, ……

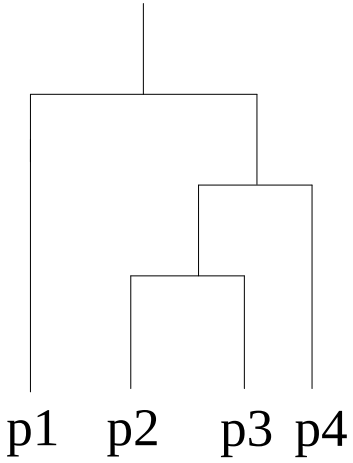


# Hierarchical Clustering

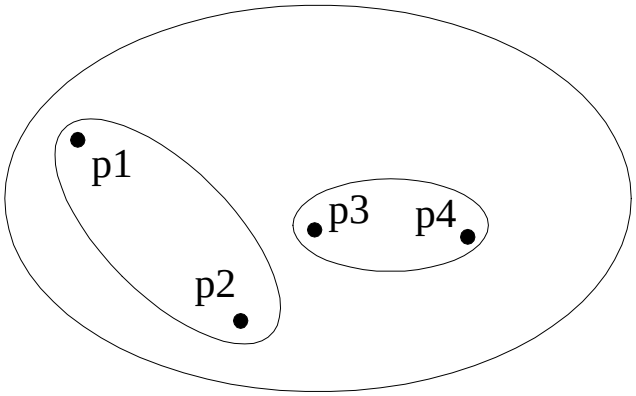




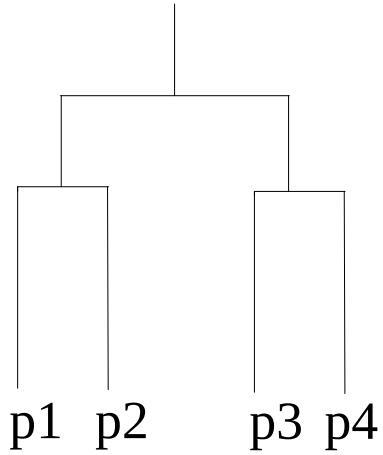

Traditional Hierarchical Clustering




Traditional Dendrogram




Non-traditional Hierarchical Clustering

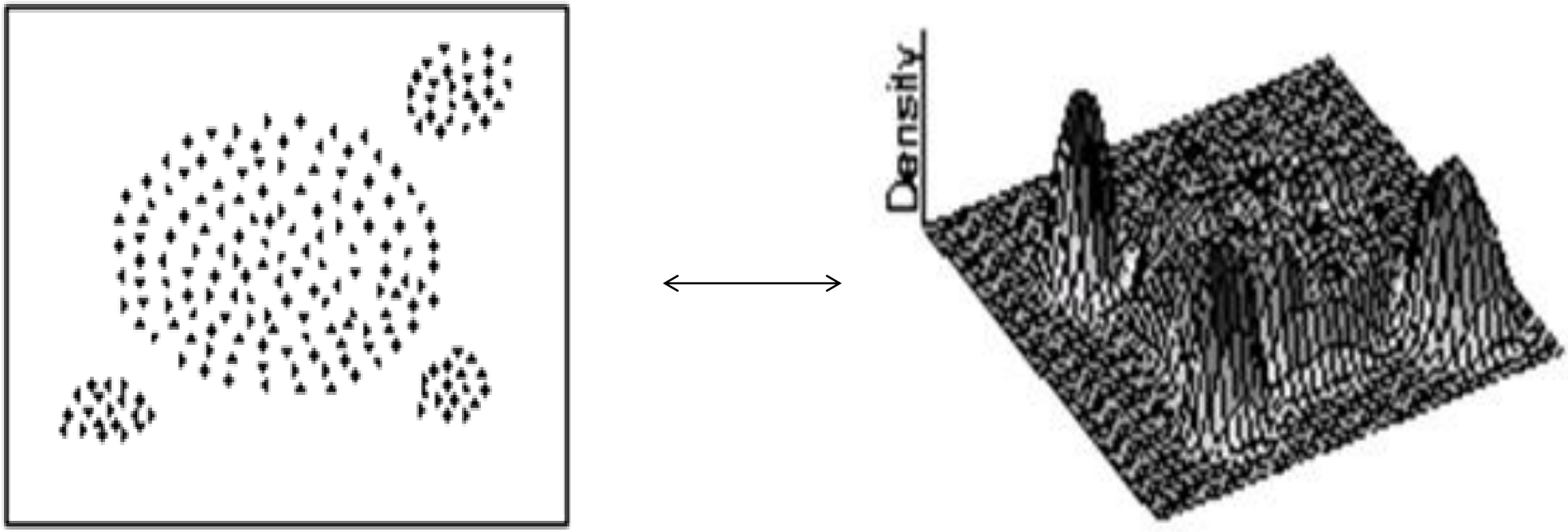



Non-traditional Dendrogram

# Major Clustering Approaches



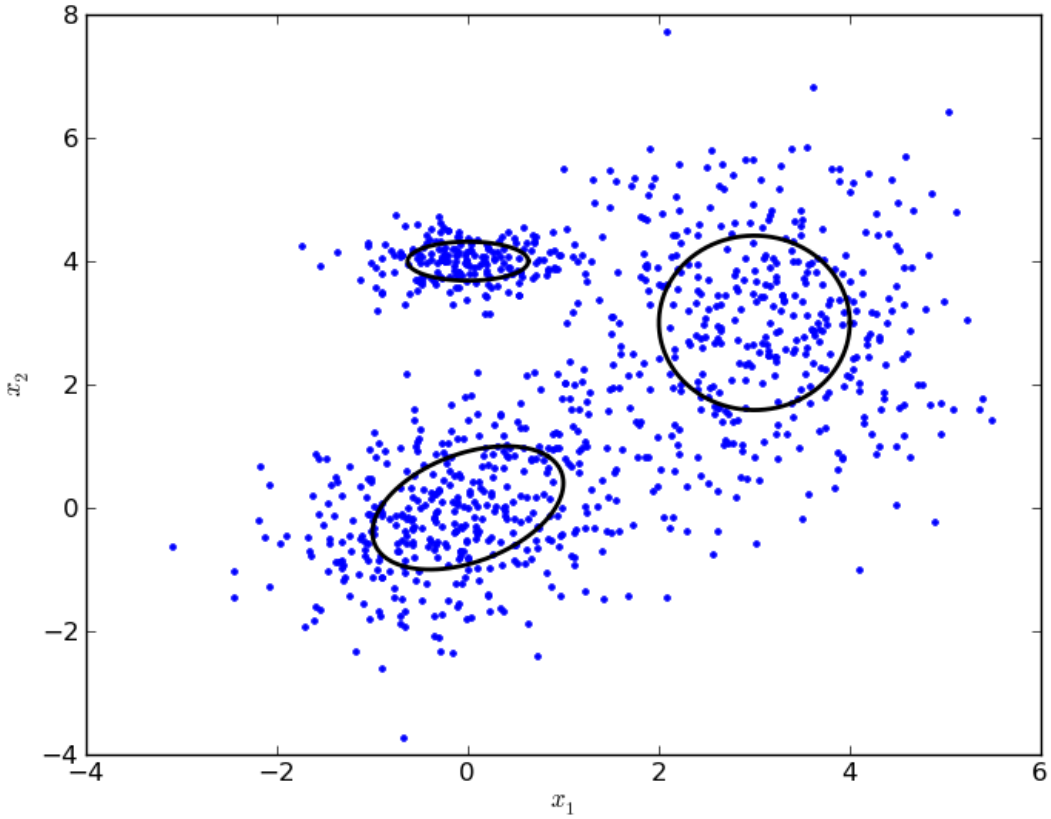
- Density-based Methodology
  - Based on connectivity and density functions
  - Typical methods: DBSACN, OPTICS, DenClue, ……



# Major Clustering Approaches



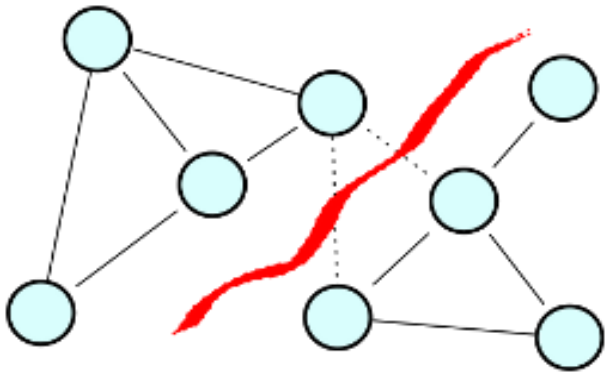
- Model-based Methodology
  - A generative model is hypothesized for each of the clusters and tries to find the best fit of that model to each other
  - Typical methods: Gaussian Mixture Model (GMM), COBWEB,…

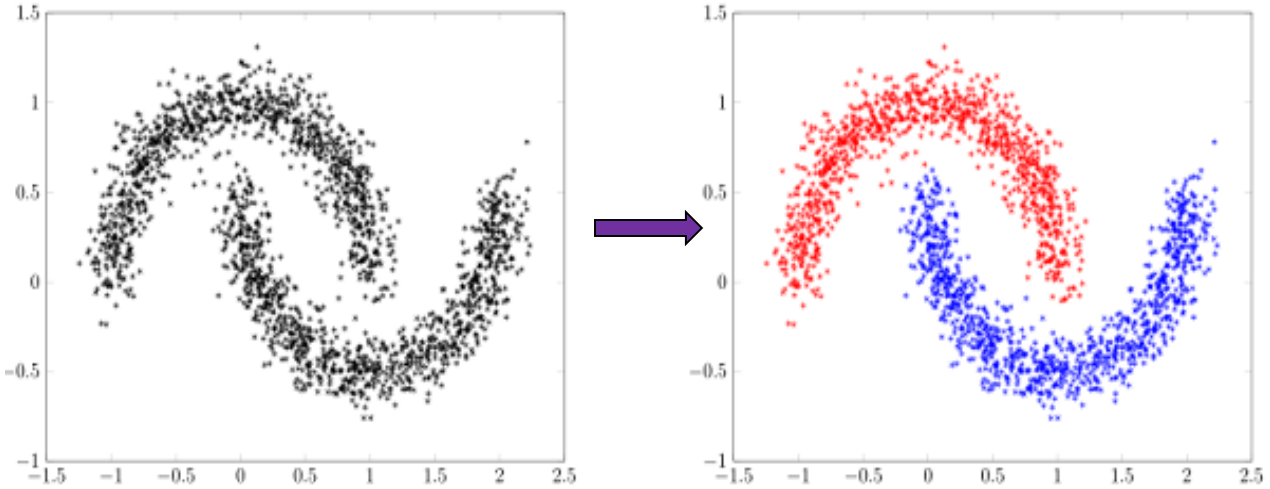


# Major Clustering Approaches



- Spectral clustering Methodology
  - Convert data set into weighted graph (vertex, edge), spectral analysis on the weighted “distance” matrix for feature extraction, apply a simple clustering method for analysis in the new feature space.
  - Typical methods: Normalised-Cuts,…

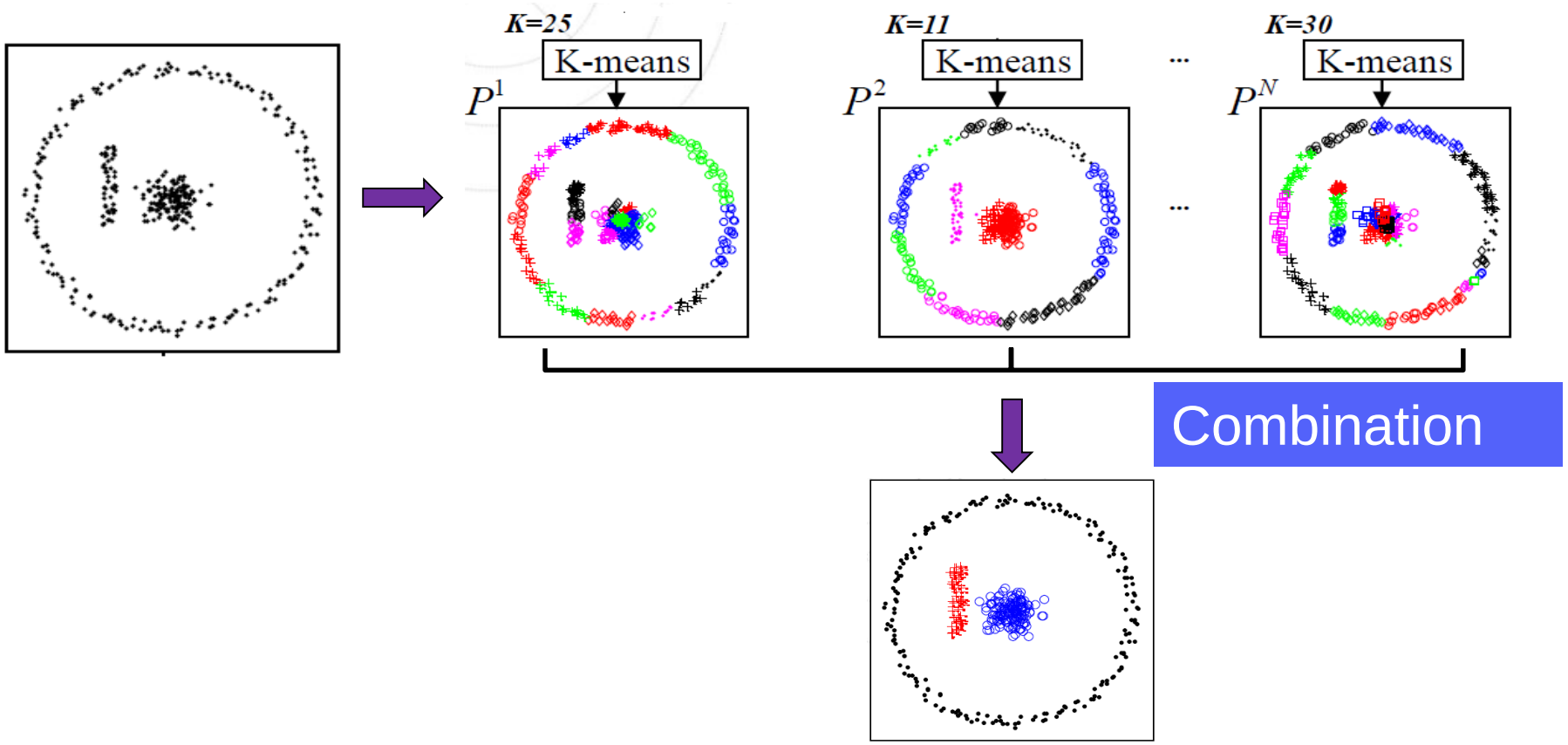




# Major Clustering Approaches



- Clustering ensemble Methodology
  - Combine multiple clustering results (different partitions) via multiple clustering analyses
  - Typical methods: **Evidence accumulation**, Graph-based …



# Types of Clusters

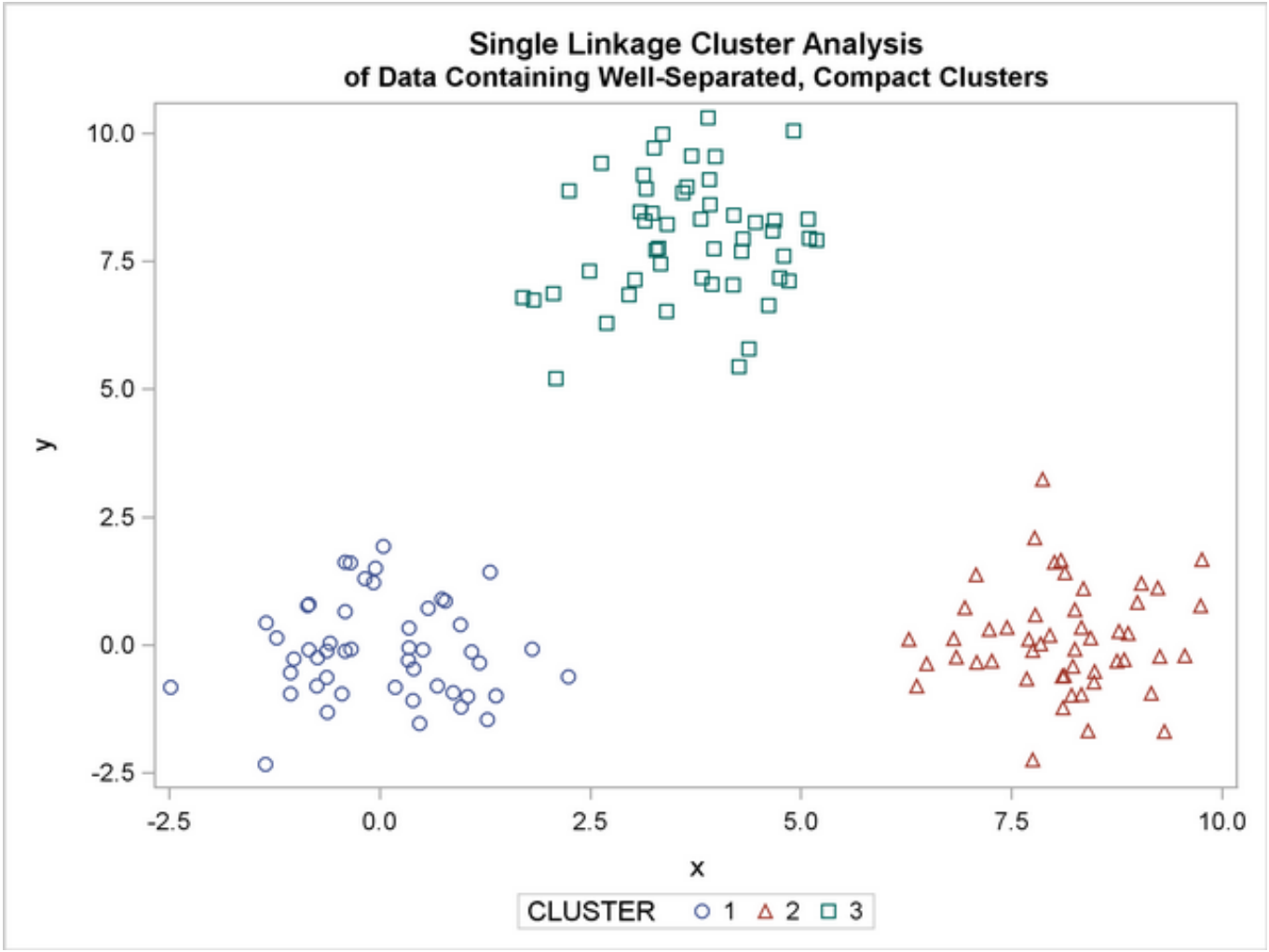


- Well-separated clusters
- Center-based clusters
- Contiguous clusters
- Density-based clusters
- Property or Conceptual based clusters
- Clusters described by an objective function

# Types of Clusters: Well-Separated



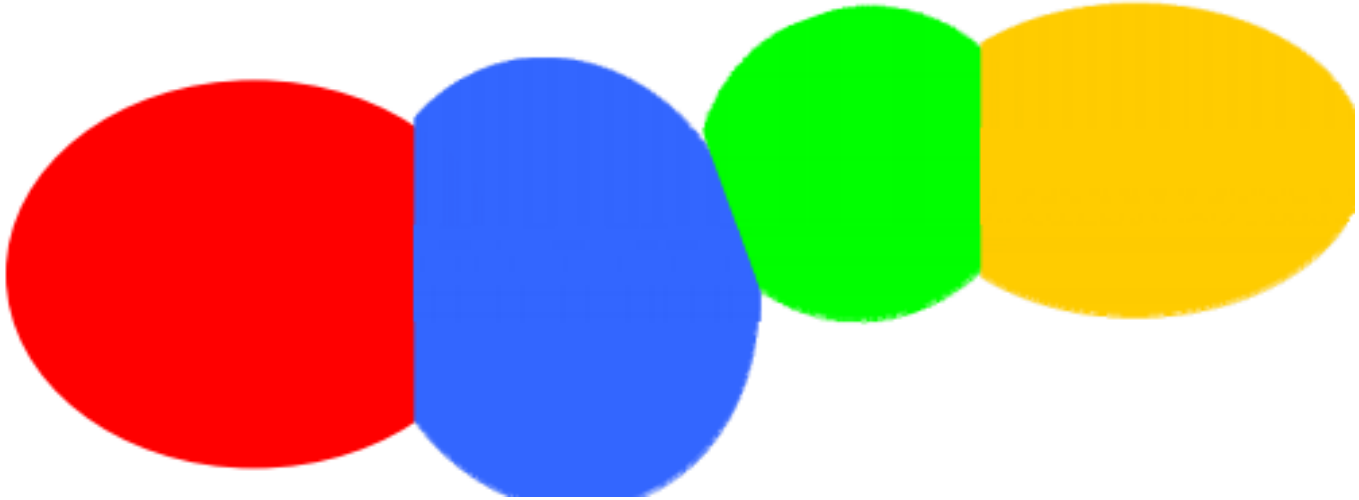
- Well-Separated Clusters:
  - A cluster is a set of points such that any point in a cluster is closer (or more similar) to every other point in the cluster than to any point not in the cluster.



# Types of Clusters: Center-Based



- Center-based
  - A cluster is a set of objects such that an object in a cluster is closer (more similar) to the “center” of a cluster, than to the center of any other cluster
  - The center of a cluster is often a centroid, the average of all the points in the cluster, or a medoid, the most “representative” point of a cluster

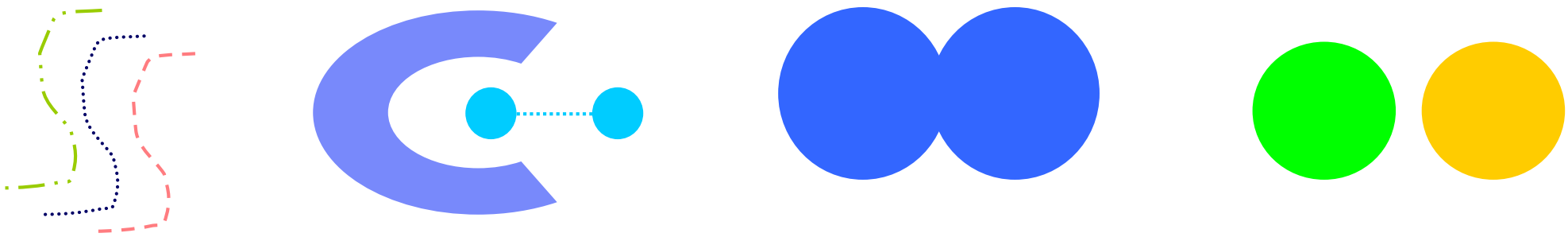


**4 center-based clusters**

# Types of Clusters: Contiguity-Based



- Contiguous Cluster (Nearest neighbor or Transitive)
  - A cluster is a set of points such that a point in a cluster is closer (or more similar) to one or more other points in the cluster than to any point not in the cluster.

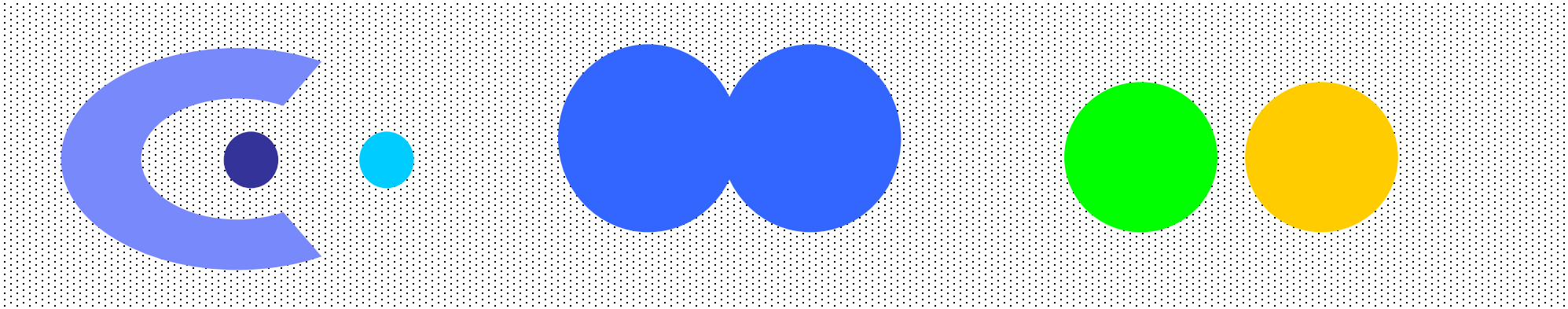


**8 contiguous clusters**

# Types of Clusters: Density-Based



- Density-based
  - A cluster is a dense region of points, which is separated by low-density regions, from other regions of high density.
  - Used when the clusters are irregular or intertwined, and when noise and outliers are present.

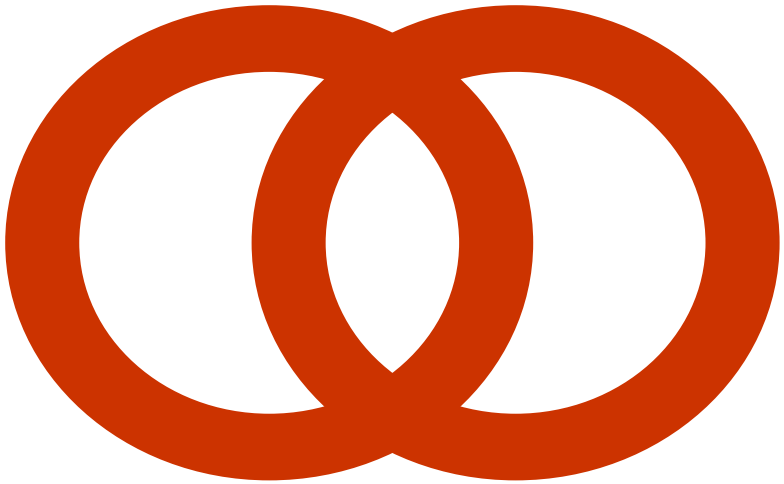


**6 density-based clusters**

# Types of Clusters: Conceptual Clusters



- Shared Property or Conceptual Clusters
  - Finds clusters that share some common property or represent a particular concept.



**2 Overlapping Circles**

# Types of Clusters: Objective Function



- Clusters Defined by an Objective Function
  - Finds clusters that minimize or maximize an objective function.
  - Enumerate all possible ways of dividing the points into clusters and evaluate the `goodness' of each potential set of clusters by using the given objective function. (NP Hard)
  - Can have global or local objectives.
    - Hierarchical clustering algorithms typically have local objectives
    - Partitional algorithms typically have global objectives
  - A variation of the global objective function approach is to fit the data to a parameterized model.
    - Parameters for the model are determined from the data.
    - Mixture models assume that the data is a ‘mixture' of a number of statistical distributions.

# Types of Clusters: Objective Function



- Map the clustering problem to a different domain and solve a related problem in that domain
  - Proximity matrix defines a weighted graph, where the nodes are the points being clustered, and the weighted edges represent the proximities between points
  - Clustering is equivalent to breaking the graph into connected components, one for each cluster.
  - Want to minimize the edge weight between clusters and maximize the edge weight within clusters

# Cluster Centroid and Distances



- Cluster centroid :
  - The centroid of a cluster is a point whose parameter values are the mean of the parameter values of all the points in the clusters.

- Distance
  - Generally, the distance between two points is taken as a common metric to assess the similarity among the components of a population. The commonly used distance measure is the Euclidean metric which defines the distance between two points p= ( p1, p2, ....) and q = ( q1, q2, ....) is given by :

$$d = \sqrt{\sum_{i=1}^{k} (p_i - q_i)^2}$$

# Similarity and Dissimilarity Between Objects



- Euclidean distance (p = 2):
- Properties of a metric *d(i,j)*:
  - *d(i,j)* $\geq$ 0
  - *d(i,i)* = 0
  - *d(i,j)* = *d(j,i)*
  - *d(i,j)* $\leq$ *d(i,k)* + *d(k,j)*

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

# Distance Measures



- Minkowski Distance (http://en.wikipedia.org/wiki/Minkowski_distance)

  For $\mathbf{x} = (x_1\ x_2 \cdots x_n)$ and $\mathbf{y} = (y_1\ y_2 \cdots y_n)$

  - $p = 1$: Manhattan (

$$d(\mathbf{x},\mathbf{y}) = \left(|x_1 - y_1|^p + |x_2 - y_2|^p \cdots + |x_n - y_n|^p\right)^{\frac{1}{p}}, \quad p > 0$$

$$d(\mathbf{x},\mathbf{y}) = |x_1 - y_1| + |x_2 - y_2| \cdots + |x_n - y_n|$$

  - $p = 2$: Euclidean distance

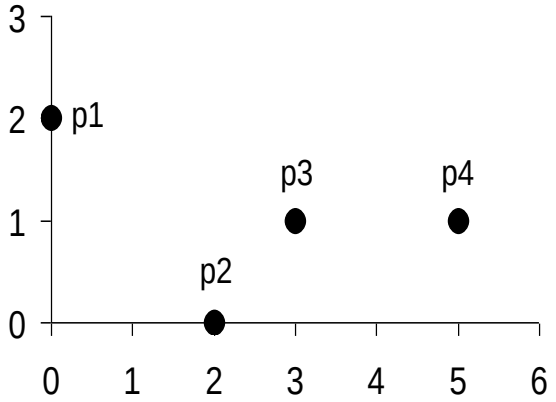
$$d(\mathbf{x},\mathbf{y}) = \sqrt{|x_1 - y_1|^2 + |x_2 - y_2|^2 \cdots + |x_n - y_n|^2}$$

  - Do not confuse $p$ with $n$, i.e., all these distances are defined based on all numbers of features (dimensions).
  - A generic measure for metric data: use appropriate $p$ in different applications

# Distance Measures



- Example: Manhatten and Euclidean distances




| point | x | y |
|---|---|---|
| p1 | 0 | 2 |
| p2 | 2 | 0 |
| p3 | 3 | 1 |
| p4 | 5 | 1 |

Data Matrix

| L1 | p1 | p2 | p3 | p4 |
|---|---|---|---|---|
| p1 | 0 | 4 | 4 | 6 |
| p2 | 4 | 0 | 2 | 4 |
| p3 | 4 | 2 | 0 | 2 |
| p4 | 6 | 4 | 2 | 0 |

Distance Matrix for Manhattan Distance

| L2 | p1 | p2 | p3 | p4 |
|---|---|---|---|---|
| p1 | 0 | 2.828 | 3.162 | 5.099 |
| p2 | 2.828 | 0 | 1.414 | 3.162 |
| p3 | 3.162 | 1.414 | 0 | 2 |
| p4 | 5.099 | 3.162 | 2 | 0 |

Distance Matrix for Euclidean Distance

# Distance Measures



- Cosine Measure (Similarity vs. Distance) for non-metric data

For $\mathbf{x} = (x_1\ x_2 \cdots x_n)$ and $\mathbf{y} = (y_1\ y_2 \cdots y_n)$

$$\cos(\mathbf{x}, \mathbf{y}) = \frac{x_1 y_1 + \cdots + x_n y_n}{\sqrt{x_1^2 + \cdots + x_n^2}\sqrt{y_1^2 + \cdots + y_n^2}}$$

$$d(\mathbf{x}, \mathbf{y}) = 1 - \cos(\mathbf{x}, \mathbf{y})$$

$0 \le d(\mathbf{x}, \mathbf{y}) \le 2$

- Property:
  - Nonmetric vector objects: keywords in documents, gene features in micro-arrays, …
  - Applications: information retrieval, biologic taxonomy, ...

# Distance Measures



- Example: Cosine measure

$$\mathbf{x}_1 = (3, 2, 0, 5, 2, 0, 0), \mathbf{x}_2 = (1,0, 0, 0,1, 0, 2)$$

$$3\times1+2\times0+0\times0+5\times0+2\times1+0\times0+0\times2=5$$

$$\sqrt{3^2+2^2+0^2+5^2+2^2+0^2+0^2}=\sqrt{42}\approx6.48$$

$$\sqrt{1^2+0^2+0^2+0^2+1^2+0^2+2^2}=\sqrt{6}\approx2.45$$

$$\cos(\mathbf{x}_1,\mathbf{x}_2)=\frac{5}{6.48\times2.45}\approx0.32$$

$$d(\mathbf{x}_1,\mathbf{x}_2)=1-\cos(\mathbf{x}_1,\mathbf{x}_2)=1-0.32=0.68$$

# Distance Measures



- Distance for Binary Features
  - For binary features, their value can be converted into 1 or 0.
  - Contingency table for binary feature vectors, and

| | | y = 1 | y = 0 |
|---|---|---|---|
| x | 1 | $a$ | $b$ |
| | 0 | $c$ | $d$ |

$a$ : number of features that equal 1 for both **x** and **y**

$b$ : number of features that equal 1 for **x** but that are 0 for **y**

$c$ : number of features that equal 0 for **x** but that are 1 for **y**

$d$ : number of features that equal 0 for both **x** and **y**

# Distance Measures



- Distance for Binary Features
  - Distance for symmetric binary features

    Both of their states equally valuable and carry the same weight; i.e., no preference on which outcome should be coded as 1 or 0 , e.g., gender

$$d(\mathbf{x},\mathbf{y}) = \frac{b+c}{a+b+c+d}$$

$$d(\mathbf{x},\mathbf{y}) = \frac{b+c}{a+b+c}$$

  - Distance for asymmetric binary features

    Outcomes of the states not equally important, e.g., the *positive* and *negative* e.g., outcomes of a disease test ; the rarest one is set to 1 and the other is 0.

# Distance Measures



- Example: Distance for binary features

  - "Y": Yes → 1
  - "P": Positive → 1
  - "N": Negative/No → 0

| Name | Gender | Fever | Cough | Test-1 | Test-2 | Test-3 | Test-4 |
|---|---|---|---|---|---|---|---|
| Jack | M | 1 | 0 | 1 | 0 | 0 | 0 |
| Mary | F | 1 | 0 | 1 | 0 | 1 | 0 |
| Jim | M | 1 | 1 | 0 | 0 | 0 | 0 |

  - gender is symmetric (not used in real applications unless having prior knowledge)
  - the remaining features are binary

Mary

| Jack | | |
|---|---|---|
| | 2 | 0 |
| | 1 | 3 |

Jim

| Jack | | |
|---|---|---|
| | 1 | 1 |
| | 1 | 3 |

Mary

| Jim | | |
|---|---|---|
| | 1 | 1 |
| | 2 | 2 |

$$d(\text{Jack},\text{Mary}) = \frac{0+1}{2+0+1} = 0.33$$

$$d(\text{Jack},\text{Jim}) = \frac{1+1}{1+1+1} = 0.67$$

$$d(\text{Jim},\text{Mary}) = \frac{1+2}{1+1+2} = 0.75$$

# Distance Measures



- Distance for nominal features
  - A generalization of the binary feature so that it can take more than two states/values, e.g., red, yellow, blue, green, ……
  - There are two methods to handle variables of such features.
- **Simple mis-matching**

$$d(\mathbf{x},\mathbf{y}) = \frac{\text{number of mis-matching features between } \mathbf{x} \text{ and } \mathbf{y}}{\text{total number of features}}$$

- **Convert it into binary variables**

**creating new binary features for all of its nominal states**

e.g., if an feature has three possible nominal states: red, yellow and blue, then this feature will be expanded into three binary features accordingly. Thus, distance measures for binary features are now applicable!

# Distance Measures



- Distance for nominal features (cont.)
  - Example: Play tennis

| | Outlook | Temperature | Humidity | Wind |
|---|---|---|---|---|
| $D_1$ | 010 | 100 | 10 | 10 |
| $D_2$ | 100 | 100 | 01 | 10 |

- **Simple mis-matching**

$$d(D_1, D_2) = \frac{2}{4} = 0.5$$

- **Creating new binary features**
  - **Using the same number of bits as those features can take**

Outlook = {Sunny, Overcast, Rain} ⟶ (100, 010, 001)

Temperature = {High, Mild, Cool} ⟶ (100, 010, 001)

Humidity = {High, Normal} ⟶ (10, 01)

Wind = {Strong, Weak} ⟶ (10, 01)

$$d(D_1, D_2) = \frac{2+2}{10} = 0.4$$

# Introduction

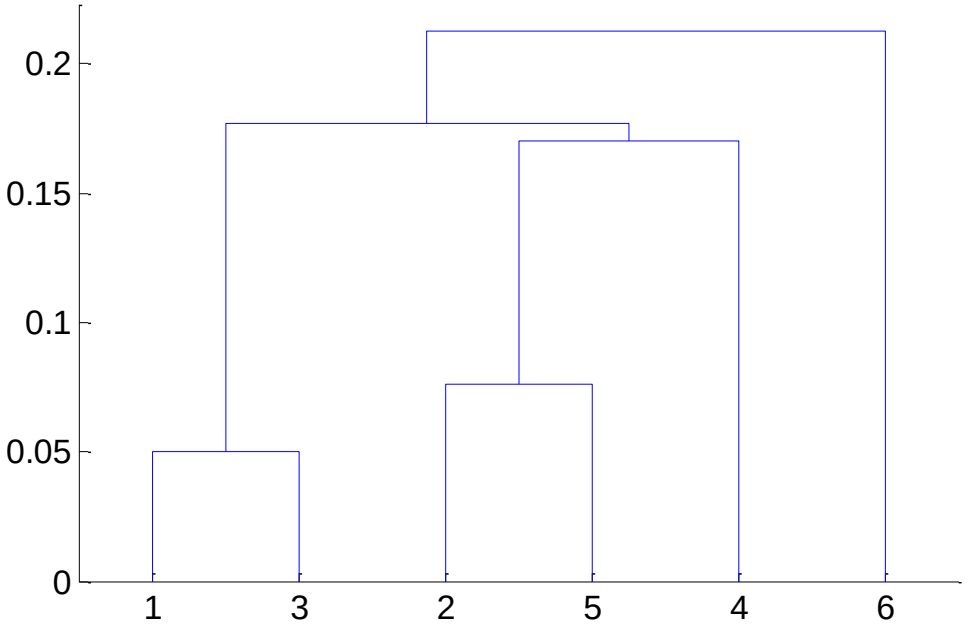


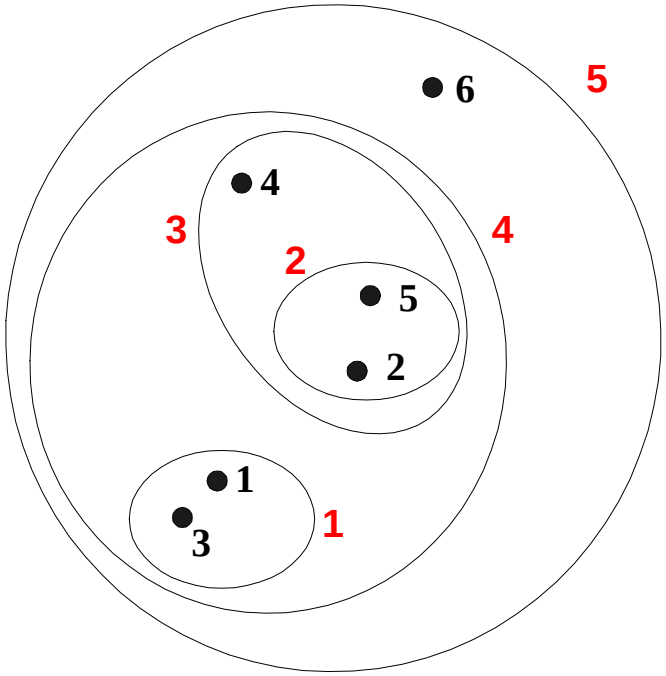
- Hierarchical Clustering Approach
  - A typical clustering analysis approach via partitioning data set **sequentially**
  - Construct nested partitions layer by layer via grouping objects into a tree of clusters (without the need to know the number of clusters in advance)
  - Use (generalised) distance matrix as clustering criteria
- Agglomerative vs. Divisive
  - Agglomerative: a bottom-up strategy
    - Initially each data object is in its own (atomic) cluster
    - Then merge these atomic clusters into larger and larger clusters
    - Agglomerative methods are commonly used in marketing research. They consist of linkage methods, variance methods, and centroid methods.
  - Divisive: a top-down strategy
    - Initially all objects are in one single cluster
    - Then the cluster is subdivided into smaller and smaller clusters
- Clustering Ensemble
  - Using multiple clustering results for robustness and overcoming weaknesses of single clustering algorithms.

# 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
- Does not require the number of clusters ***k*** in advance
- Needs a termination/readout condition

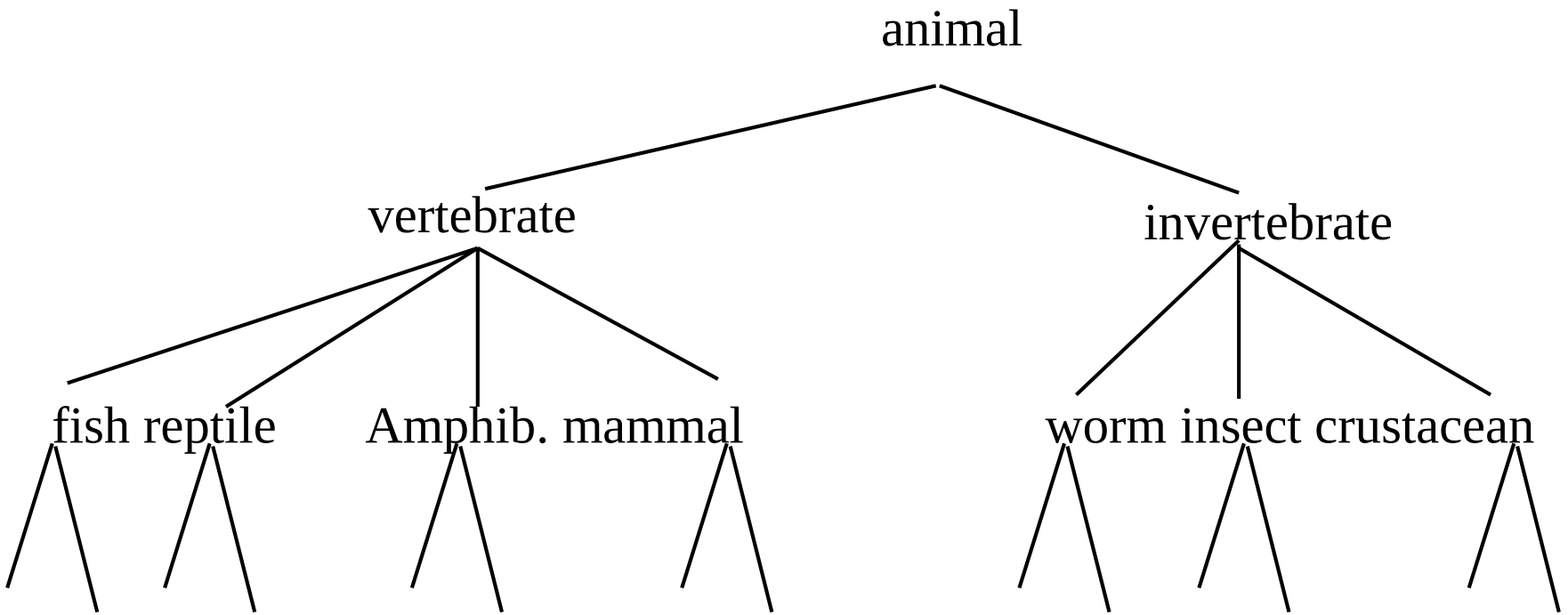


# Hierarchical Clustering



- Build a tree-based hierarchical taxonomy (*dendrogram*) from a set of documents.

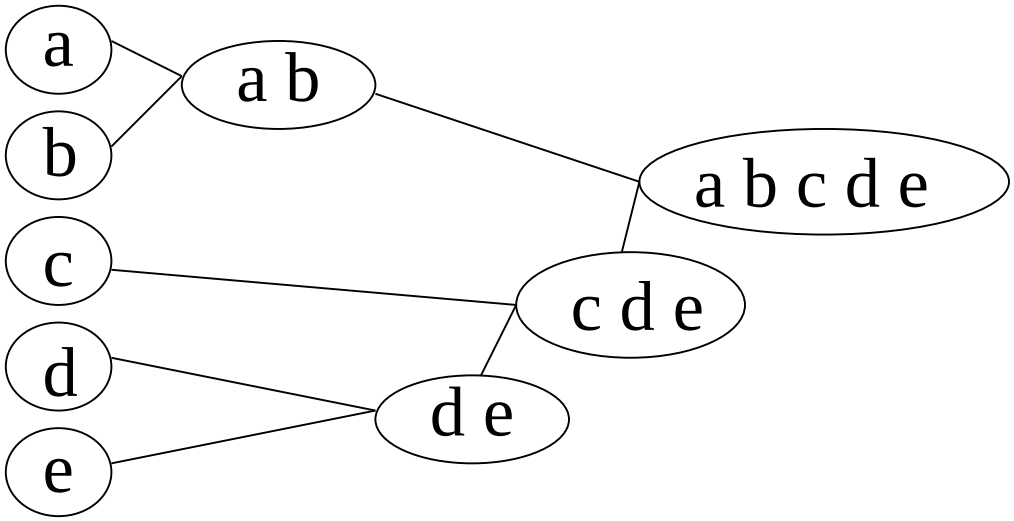



- One approach: recursive application of a partitional clustering algorithm.

# Hierarchical Clustering



- Agglomerative approach




Initialization:

Each object is a cluster

Iteration:

Merge two clusters which are most similar to each other;

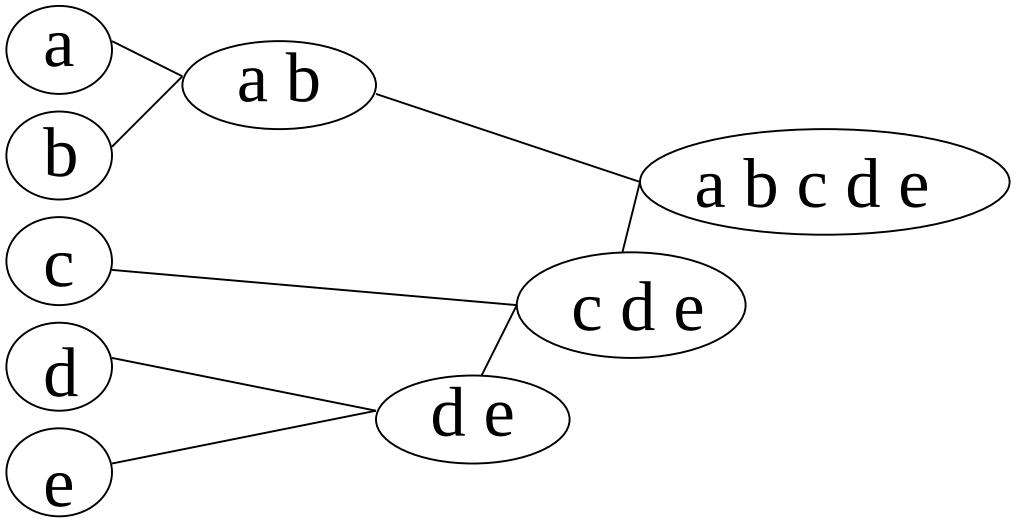
Until all objects are merged into a single cluster


bottom-up

# Hierarchical Clustering



- Divisive Approaches



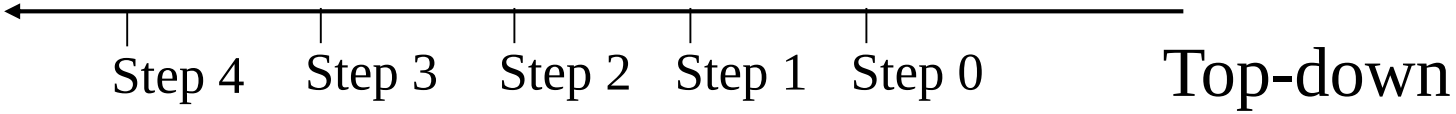

Initialization:

All objects stay in one cluster

Iteration:

Select a cluster and split it into two sub clusters

Until each leaf cluster contains only one object



# Hierarchical Agglomerative Clustering (HAC)



- Starts with each doc in a separate cluster
  - then repeatedly joins the *closest pair* of clusters, until there is only one cluster.
- The history of merging forms a binary tree or hierarchy.

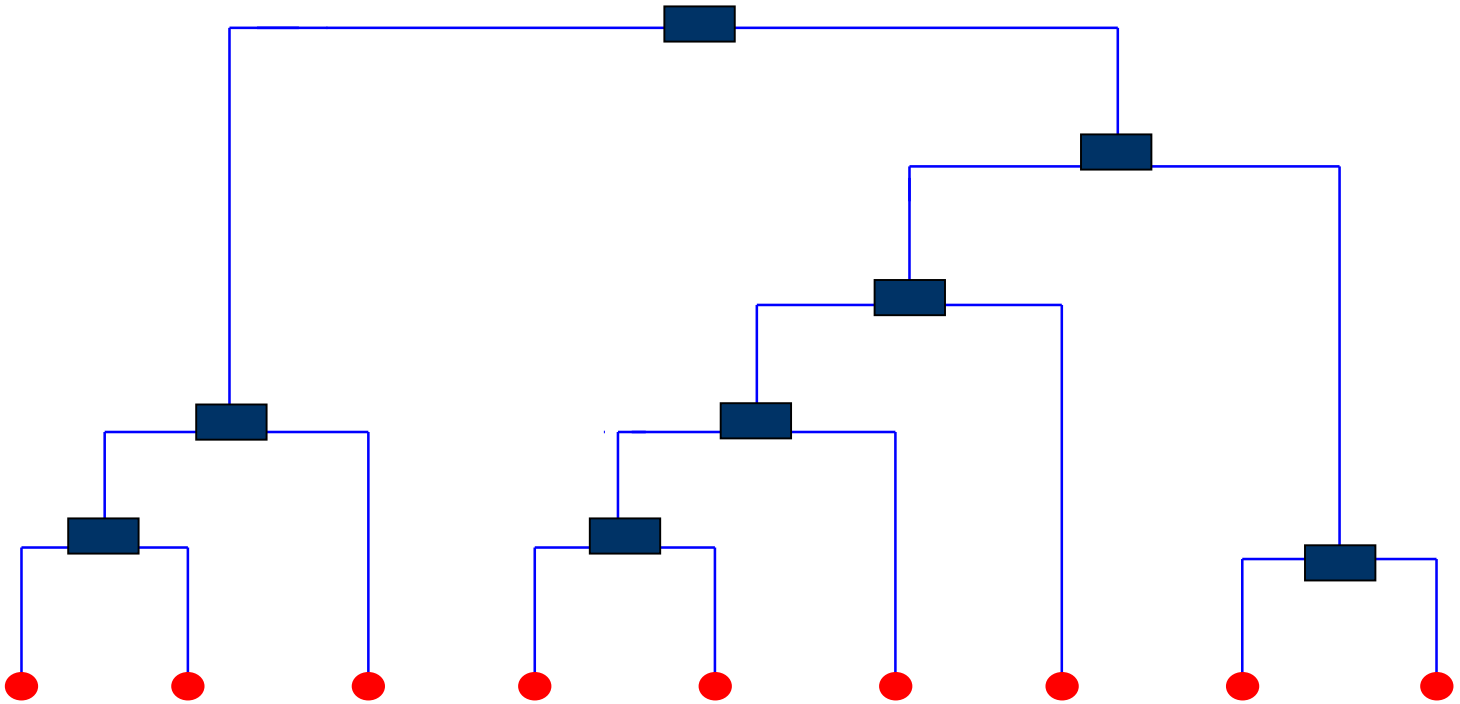
How to measure distance of clusters??

- Assumes a similarity function for determining the similarity of two instances.
- Starts with all instances in a separate cluster and then repeatedly joins the two clusters that are most similar until there is only one cluster.
- The history of merging forms a binary tree or hierarchy.

# Dendrogram



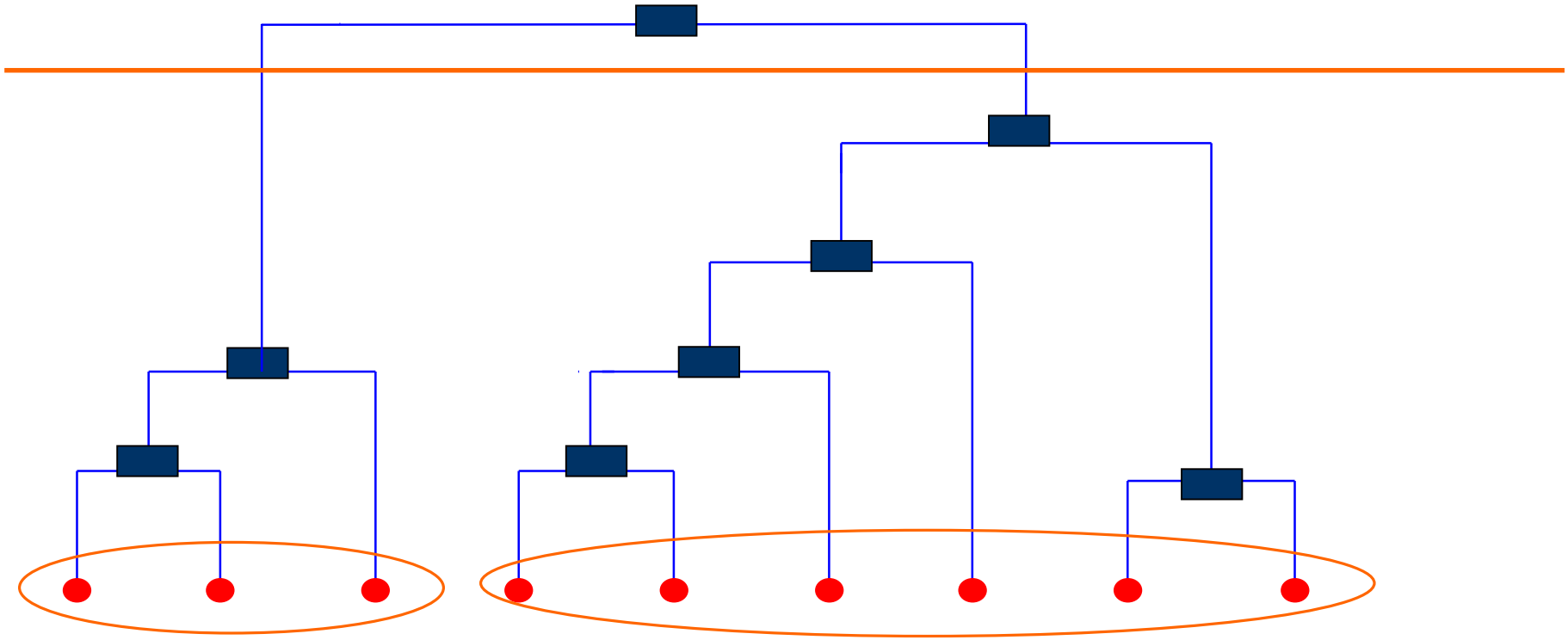
- A binary tree that shows how clusters are merged/split hierarchically
- Each node on the tree is a cluster; each leaf node is a singleton cluster



# Dendrogram



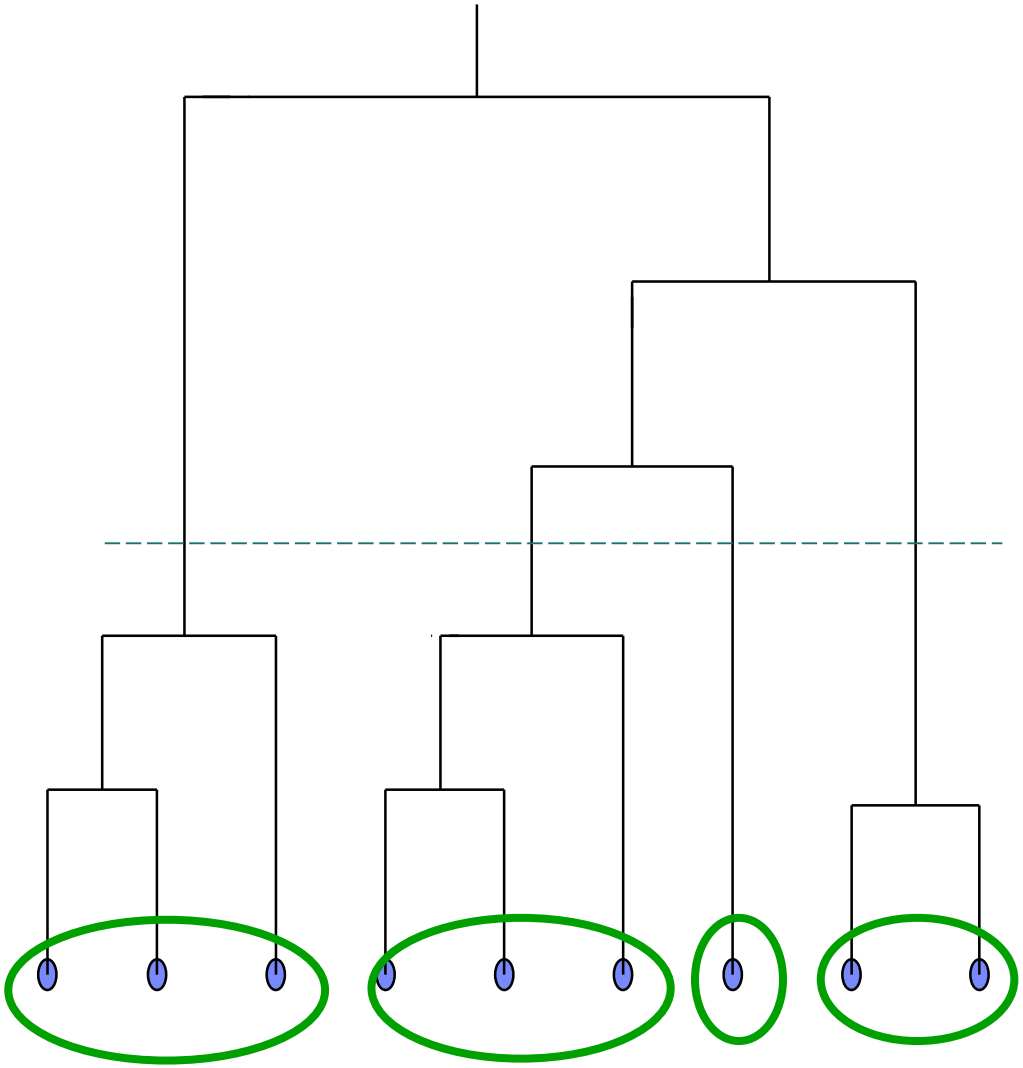
- A clustering of the data objects is obtained by cutting the *dendrogram* at the desired level, then each connected component forms a cluster



# Dendrogram: Hierarchical Clustering

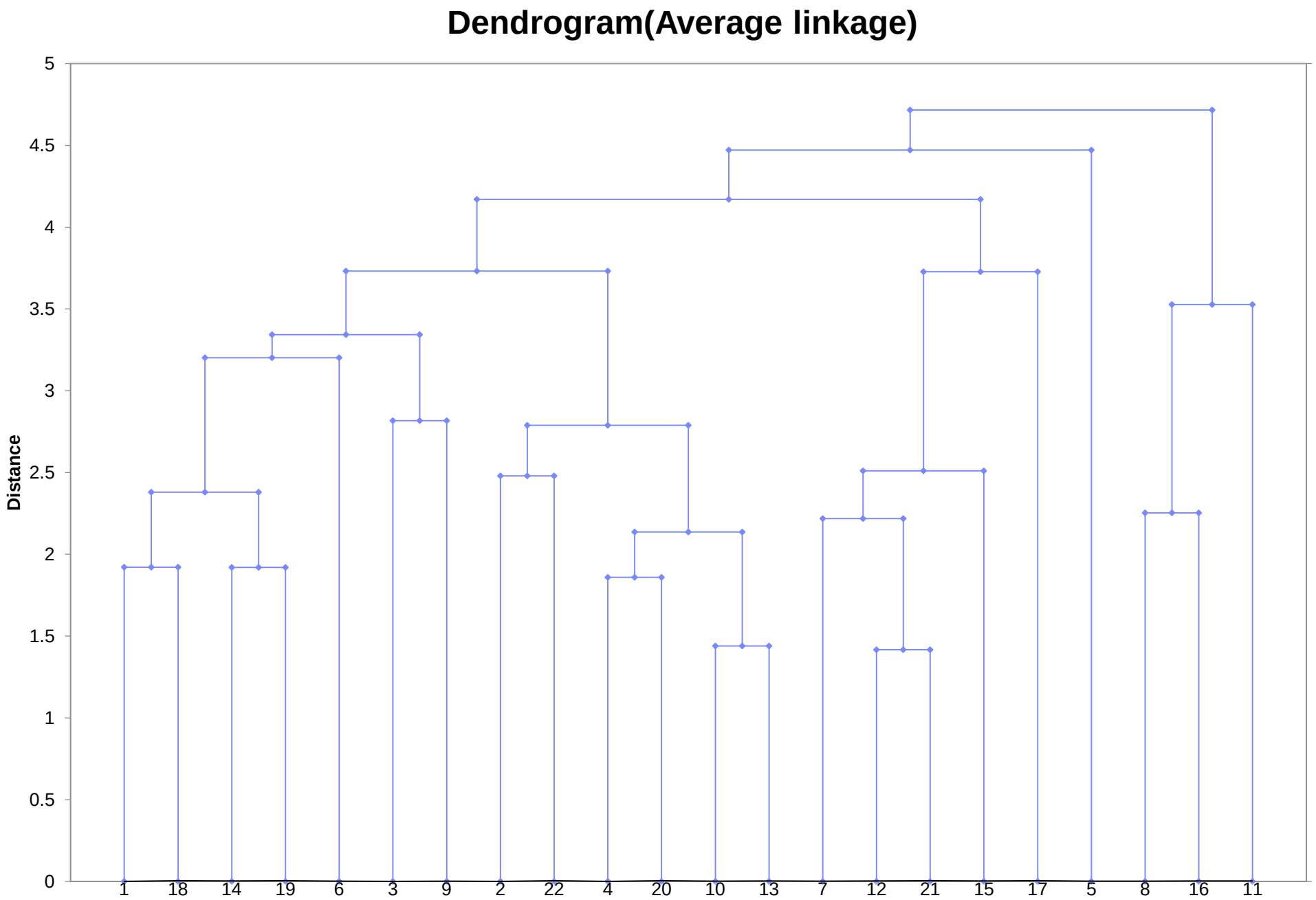


- Clustering obtained by cutting the Dendrogram at a desired level: each connected component forms a cluster.



# The Dendrogram: Hierarchical Clustering





# Hierarchical Agglomerative Clustering- Linkage Method



- The **single linkage** method is based on minimum distance, or the nearest neighbor rule.
- The **complete linkage** method is based on the maximum distance or the furthest neighbor approach.
- The **average linkage** method the distance between two clusters is defined as the average of the distances between all pairs of objects

# Hierarchical Agglomerative Clustering- Variance and Centroid Method



- **Variance methods** generate clusters to minimize the within-cluster variance.
- **Ward's procedure** is commonly used. For each cluster, the sum of squares is calculated. The two clusters with the smallest increase in the overall sum of squares within cluster distances are combined.
- In the **centroid methods**, the distance between two clusters is the distance between their centroids (means for all the variables),
- Of the hierarchical methods, average linkage and Ward's methods have been shown to perform better than the other procedures.

# Agglomerative clustering algorithm



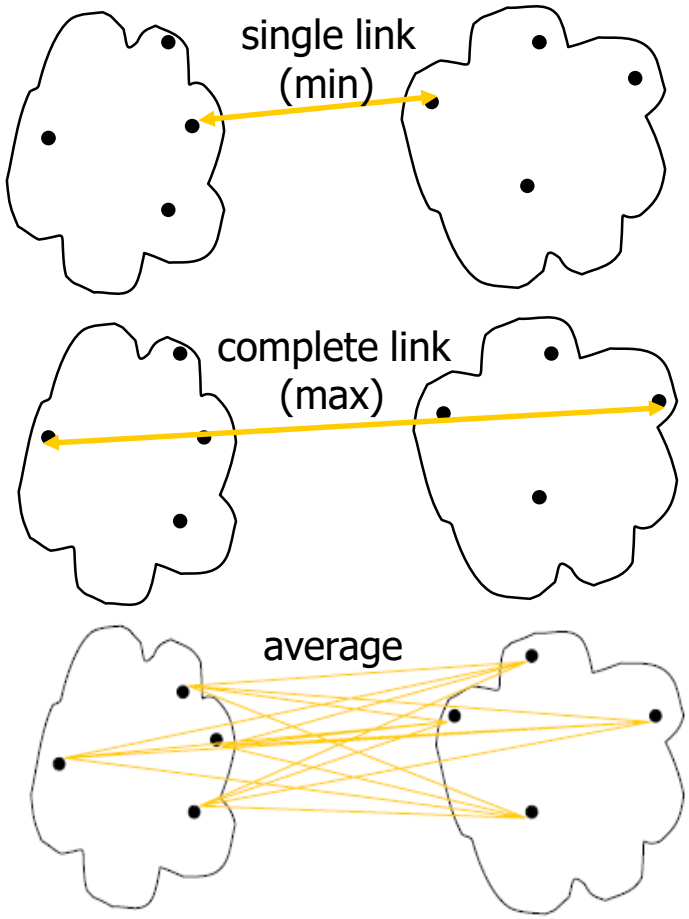
- Most popular hierarchical clustering technique
- Basic algorithm
  1. Compute the distance matrix between the input data points
  2. Let each data point be a cluster
  3. **Repeat**
  4. Merge the two closest clusters
  5. Update the distance matrix
  6. **Until** only a single cluster remains
- Key operation is the computation of the distance between two clusters
  - Different definitions of the distance between clusters lead to different algorithms

# Cluster Distance Measures



- **Single link:** Smallest distance between an element in one cluster and an element in the other, i.e., $d(C_i, C_j) = \min\{d(x_{ip}, x_{jq})\}$
- **Complete link:** Largest distance between an element in one cluster and an element in the other, i.e., $d(C_i, C_j) = \max\{d(x_{ip}, x_{jq})\}$
- **Average:** Average distance between elements in one cluster and elements in the other, i.e.,

$$d(C_i, C_j) = \text{avg}\{d(x_{ip}, x_{jq})\}$$



# Cluster Distance Measures



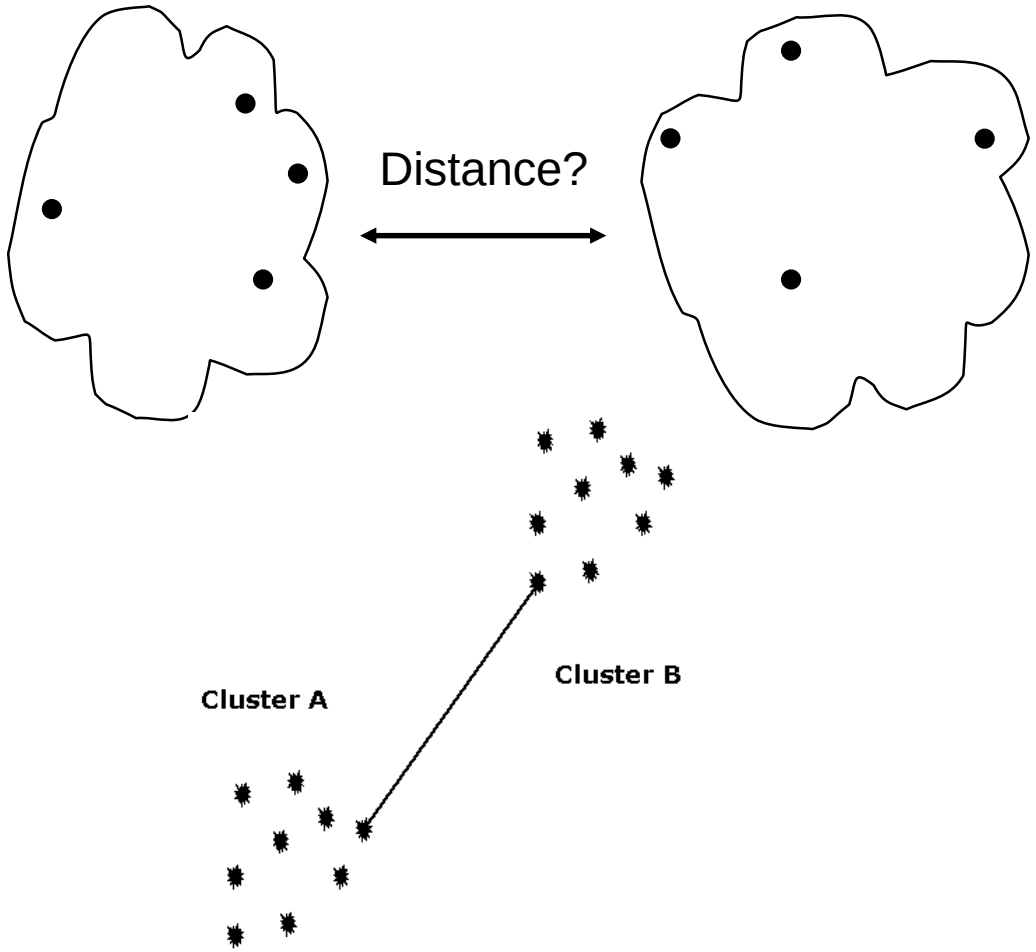
- How to measure the distance between clusters?

**Single-link**

**Complete-link**

**Average-link**

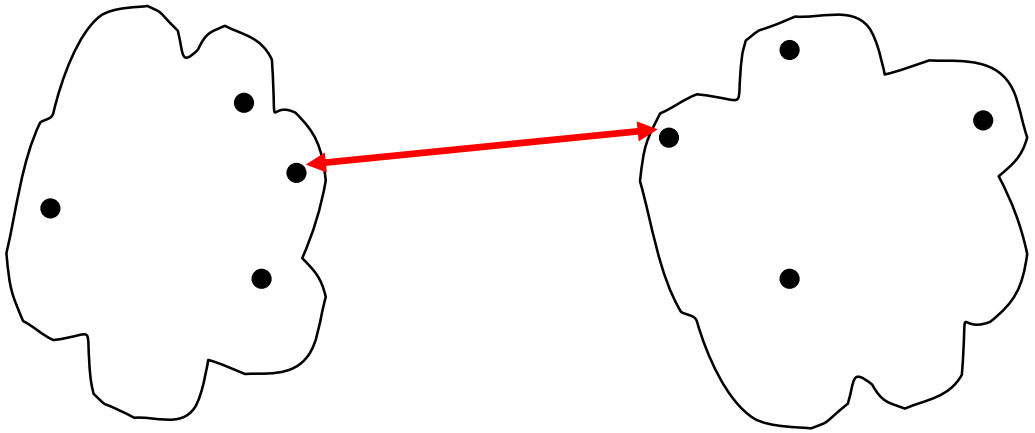
**Centroid distance**




Hint: *Distance between clusters* is usually defined on the basis of *distance between objects.*

# Cluster Distance Measures





**Single-link**

**Complete-link**

**Average-link**

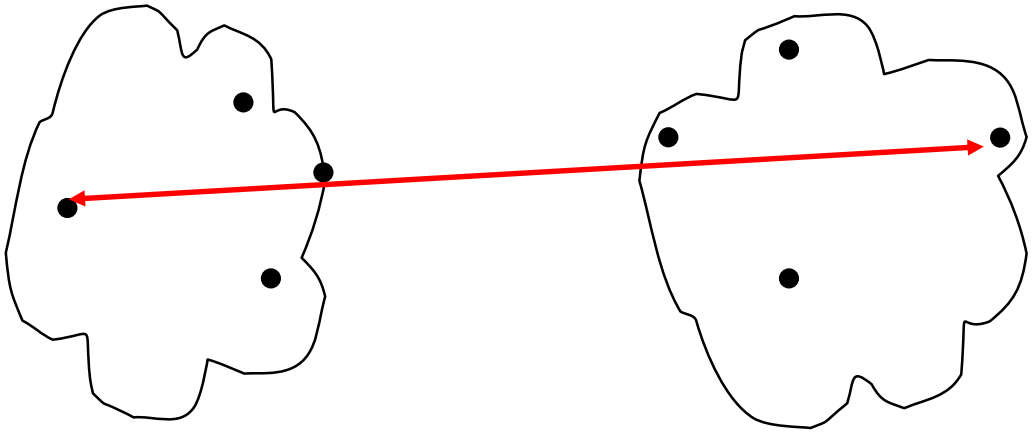
**Centroid distance**

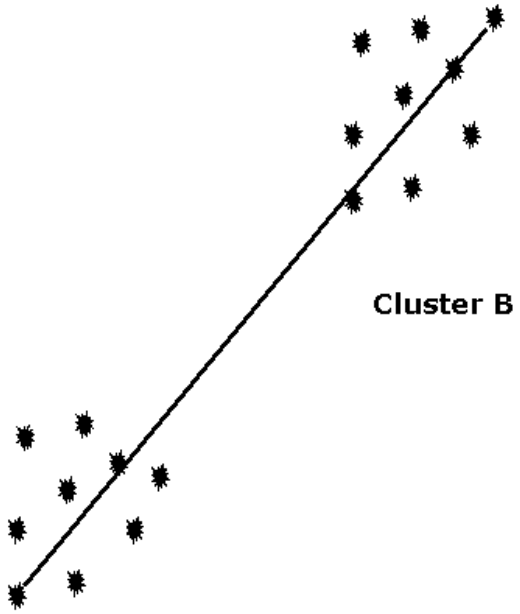
$$d_{\min}(C_i, C_j) = \min_{p \in C_i, q \in C_j} d(p, q)$$

The distance between two clusters is represented by the distance of the *closest pair of data objects* belonging to different clusters.

# Cluster Distance Measures








**Single-link**

**Complete-link**

**Average-link**

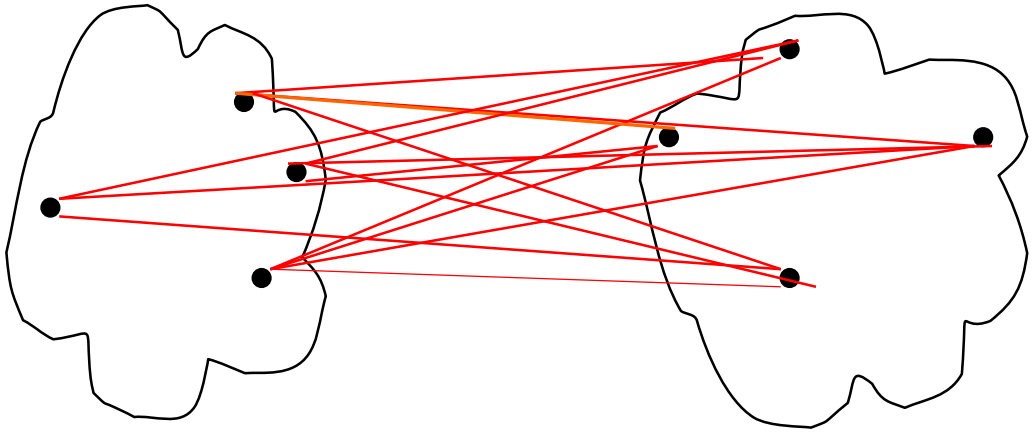
**Centroid distance**

$$d_{\min}(C_i, C_j) = \max_{p \in C_i, q \in C_j} d(p, q)$$

The distance between two clusters is represented by the distance of the *farthest pair of data objects* belonging to different clusters.

# Cluster Distance Measures





**Single-link**

**Complete-link**

**Average-link**

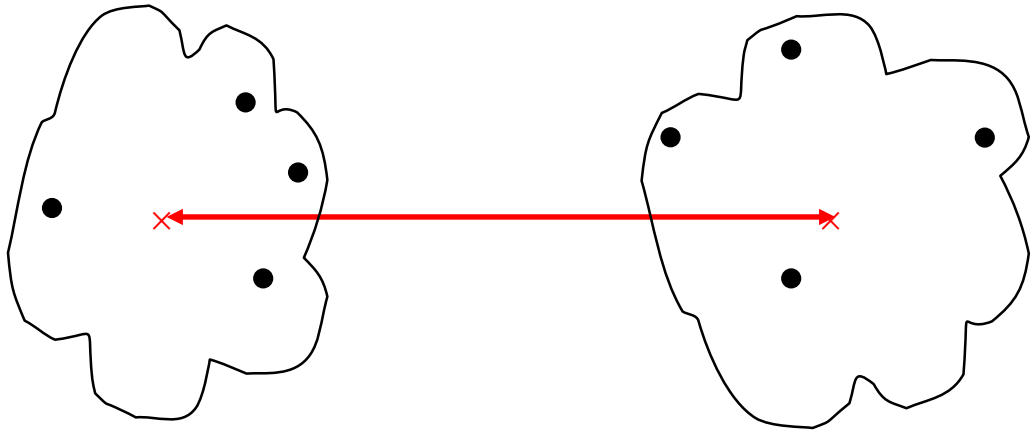
**Centroid distance**

$$d_{\min}(C_i, C_j) = \underset{p \in C_i, q \in C_j}{avg} d(p, q)$$

The distance between two clusters is represented by the *average* distance of *all pairs of data objects* belonging to different clusters.

# Cluster Distance Measures





$m_i, m_j$ are the means of $C_i$, $C_j$,

**Single-link**

**Complete-link**

**Average-link**

**Centroid distance**

$$d_{mean}(C_i, C_j) = d(m_i, m_j)$$

The distance between two clusters is represented by the distance between *the means of the cluters.*

# Cluster Distance Measures



**Example**: Given a data set of five objects characterized by a single continuous feature, assume that there are two clusters: $C_1$: {a, b} and $C_2$: {c, d, e}. (Minkowski distance for distance matrix)

| | a | b | c | d | e |
|---|---|---|---|---|---|
| Feature | 1 | 2 | 4 | 5 | 6 |

1. Calculate the distance matrix .

| | a | b | c | d | e |
|---|---|---|---|---|---|
| **a** | 0 | 1 | 3 | 4 | 5 |
| **b** | 1 | 0 | 2 | 3 | 4 |
| **c** | 3 | 2 | 0 | 1 | 2 |
| **d** | 4 | 3 | 1 | 0 | 1 |
| **e** | 5 | 4 | 2 | 1 | 0 |

2. Calculate three cluster distances between $C_1$ and $C_2$.

## Single link

$$\text{dist}(C_1, C_2) = \min\{d(a,c), d(a,d), d(a,e), d(b,c), d(b,d), d(b,e)\}$$
$$= \min\{3, 4, 5, 2, 3, 4\} = 2$$

## Complete link

$$\text{dist}(C_1, C_2) = \max\{d(a,c), d(a,d), d(a,e), d(b,c), d(b,d), d(b,e)\}$$
$$= \max\{3, 4, 5, 2, 3, 4\} = 5$$

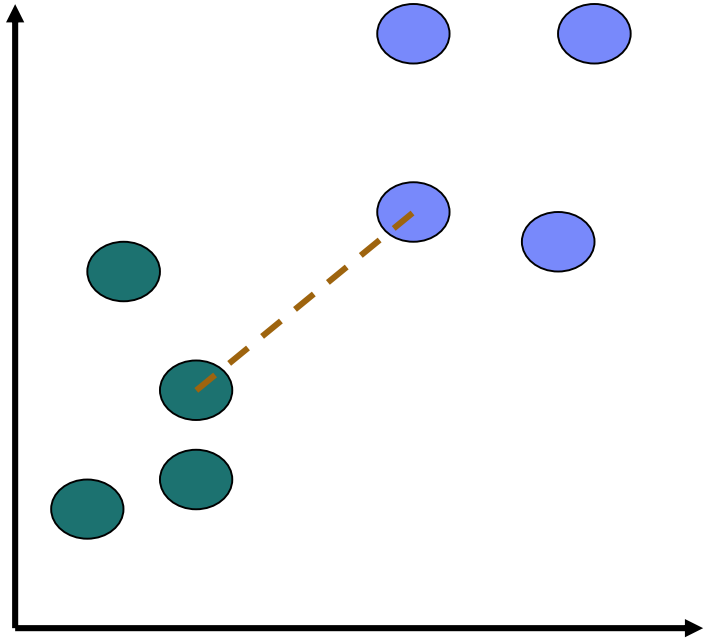
## Average

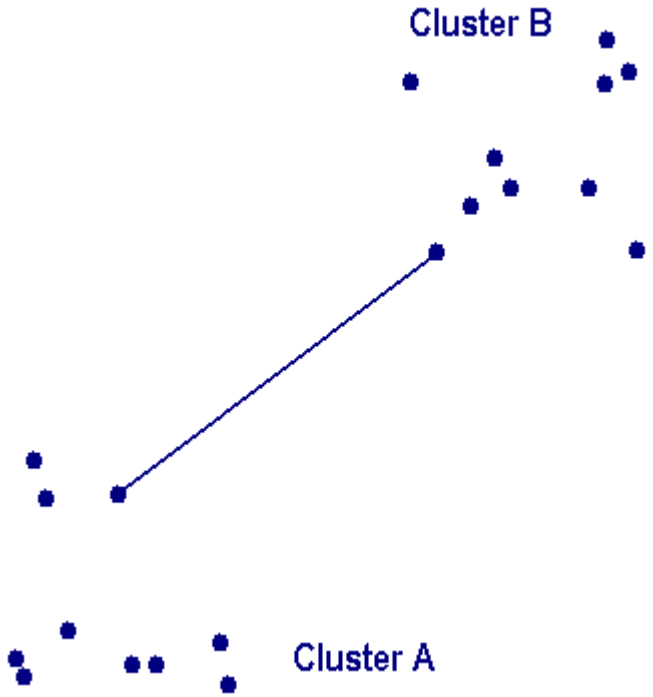
$$\text{dist}(C_1, C_2) = \frac{d(a,c) + d(a,d) + d(a,e) + d(b,c) + d(b,d) + d(b,e)}{6}$$
$$= \frac{3+4+5+2+3+4}{6} = \frac{21}{6} = 3.5$$

# Single Link Agglomerative Clustering



- The minimum of all pairwise distances between points in the two clusters
- Tends to produce long, “loose” clusters





# Single Link Agglomerative Clustering



- Use maximum similarity of pairs:

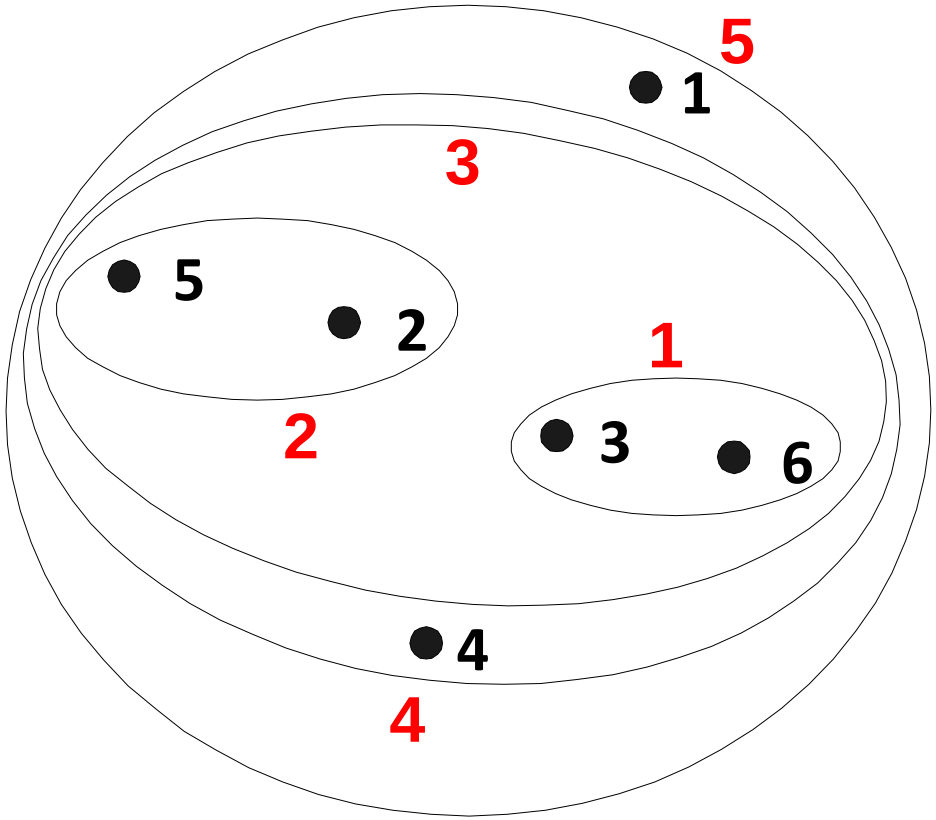
$$sim(c_i, c_j) = \max_{x \in c_i, y \in c_j} sim(x, y)$$

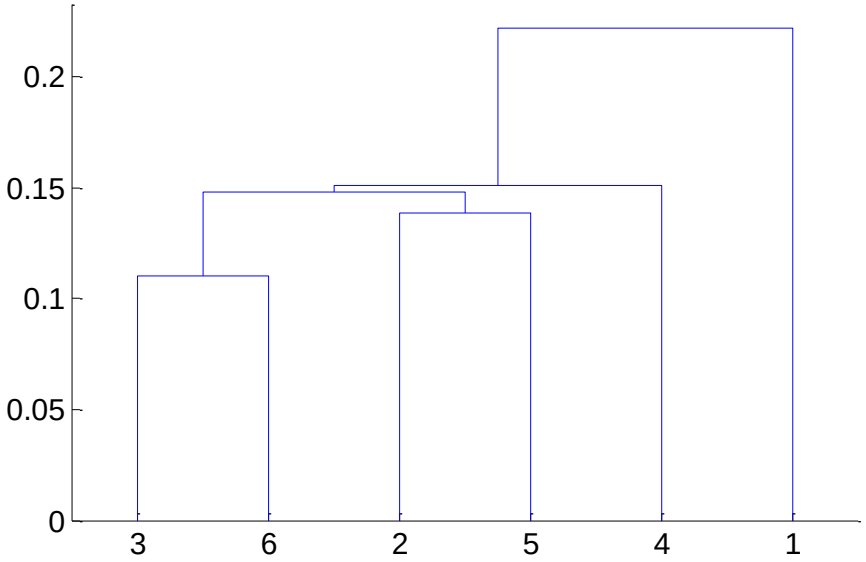
- Can result in "straggly" (long and thin) clusters due to chaining effect.
- After merging $c_i$ and $c_j$, the similarity of the resulting cluster to another cluster, $c_k$, is:

$$sim((c_i \cup c_j), c_k) = \max(sim(c_i, c_k), sim(c_j, c_k))$$

# Illustration






**Nested Clusters**

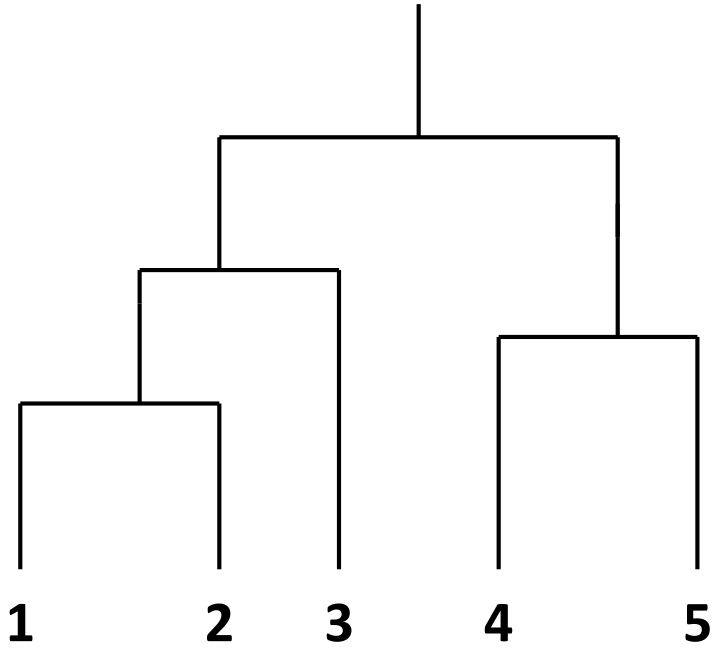
**Dendrogram**

# Single-link clustering: example



- Determined by one pair of points, i.e., by one link in the proximity graph.

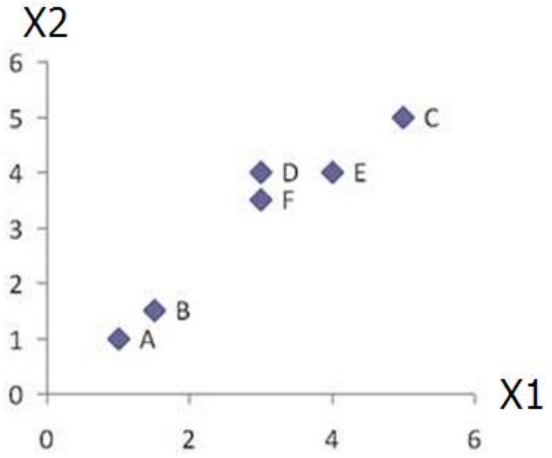
| | I1 | I2 | I3 | I4 | I5 |
|---|---|---|---|---|---|
| I1 | 1.00 | 0.90 | 0.10 | 0.65 | 0.20 |
| I2 | 0.90 | 1.00 | 0.70 | 0.60 | 0.50 |
| I3 | 0.10 | 0.70 | 1.00 | 0.40 | 0.30 |
| I4 | 0.65 | 0.60 | 0.40 | 1.00 | 0.80 |
| I5 | 0.20 | 0.50 | 0.30 | 0.80 | 1.00 |



# Single-link clustering: example – 1



- Problem: clustering analysis with agglomerative algorithm




| | X1 | X2 |
|---|---|---|
| A | 1 | 1 |
| B | 1.5 | 1.5 |
| C | 5 | 5 |
| D | 3 | 4 |
| E | 4 | 4 |
| F | 3 | 3.5 |

data matrix

$$d_{AB} = \left((1-1.5)^2 + (1-1.5)^2\right)^{\frac{1}{2}} = \sqrt{\tfrac{1}{2}} = 0.7071$$

$$d_{DF} = \left((3-3)^2 + (4-3.5)^2\right)^{\frac{1}{2}} = 0.5$$

Euclidean distance

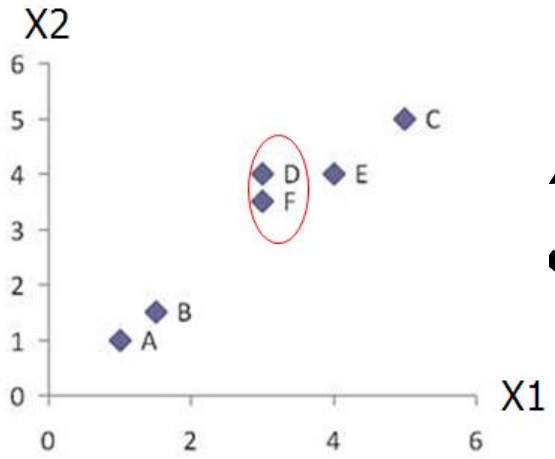
| Dist | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| A | 0.00 | 0.71 | 5.66 | 3.61 | 4.24 | 3.20 |
| B | 0.71 | 0.00 | 4.95 | 2.92 | 3.54 | 2.50 |
| C | 5.66 | 4.95 | 0.00 | 2.24 | 1.41 | 2.50 |
| D | 3.61 | 2.92 | 2.24 | 0.00 | 1.00 | 0.50 |
| E | 4.24 | 3.54 | 1.41 | 1.00 | 0.00 | 1.12 |
| F | 3.20 | 2.50 | 2.50 | 0.50 | 1.12 | 0.00 |

distance matrix

# Single-link clustering: example – 1



- Merge two closest clusters (iteration 1)




| Dist | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| A | 0.00 | 0.71 | 5.66 | 3.61 | 4.24 | 3.20 |
| B | 0.71 | 0.00 | 4.95 | 2.92 | 3.54 | 2.50 |
| C | 5.66 | 4.95 | 0.00 | 2.24 | 1.41 | 2.50 |
| D | 3.61 | 2.92 | 2.24 | 0.00 | 1.00 | 0.50 |
| E | 4.24 | 3.54 | 1.41 | 1.00 | 0.00 | 1.12 |
| F | 3.20 | 2.50 | 2.50 | 0.50 | 1.12 | 0.00 |

| Dist | A | B | C | D, F | E |
|---|---|---|---|---|---|
| A | 0.00 | 0.71 | 5.66 | ? | 4.24 |
| B | 0.71 | 0.00 | 4.95 | ? | 3.54 |
| C | 5.66 | 4.95 | 0.00 | ? | 1.41 |
| D, F | ? | ? | ? | 0.00 | ? |
| E | 4.24 | 3.54 | 1.41 | ? | 0.00 |

# Single-link clustering: example – 1



- Update distance matrix (iteration 1)

| Dist | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| A | 0.00 | 0.71 | 5.66 | 3.61 | 4.24 | 3.20 |
| B | 0.71 | 0.00 | 4.95 | 2.92 | 3.54 | 2.50 |
| C | 5.66 | 4.95 | 0.00 | 2.24 | 1.41 | 2.50 |
| D | 3.61 | 2.92 | 2.24 | 0.00 | 1.00 | 0.50 |
| E | 4.24 | 3.54 | 1.41 | 1.00 | 0.00 | 1.12 |
| F | 3.20 | 2.50 | 2.50 | 0.50 | 1.12 | 0.00 |

$$d_{(D,F)\to A} = \min\left(d_{DA}, d_{FA}\right) = \min\left(3.61, 3.20\right) = 3.20$$

$$d_{(D,F)\to B} = \min\left(d_{DB}, d_{FB}\right) = \min\left(2.92, 2.50\right) = 2.50$$

$$d_{(D,F)\to C} = \min\left(d_{DC}, d_{FC}\right) = \min\left(2.24, 2.50\right) = 2.24$$

$$d_{E\to(D,F)} = \min\left(d_{ED}, d_{EF}\right) = \min\left(1.00, 1.12\right) = 1.00$$

| Dist | A | B | C | D, F | E |
|---|---|---|---|---|---|
| A | 0.00 | 0.71 | 5.66 | ? | 4.24 |
| B | 0.71 | 0.00 | 4.95 | ? | 3.54 |
| C | 5.66 | 4.95 | 0.00 | ? | 1.41 |
| D, F | ? | ? | ? | 0.00 | ? |
| E | 4.24 | 3.54 | 1.41 | ? | 0.00 |

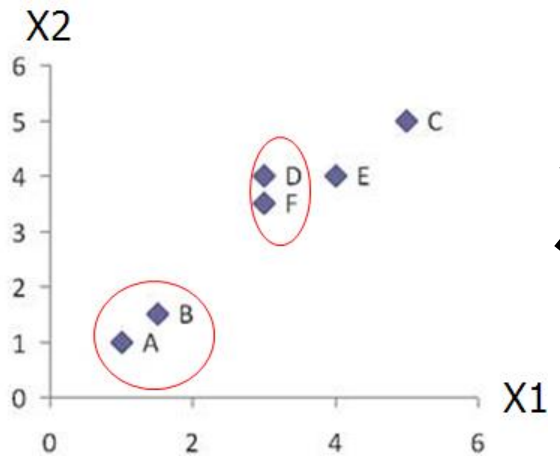
**Min Distance (Single Linkage)**

| Dist | A | B | C | D, F | E |
|---|---|---|---|---|---|
| A | 0.00 | 0.71 | 5.66 | 3.20 | 4.24 |
| B | 0.71 | 0.00 | 4.95 | 2.50 | 3.54 |
| C | 5.66 | 4.95 | 0.00 | 2.24 | 1.41 |
| D, F | **3.20** | **2.50** | **2.24** | 0.00 | 1.00 |
| E | 4.24 | 3.54 | 1.41 | **1.00** | 0.00 |

# Single-link clustering: example – 1



- Merge two closest clusters (iteration 2)




**Min Distance (Single Linkage)**

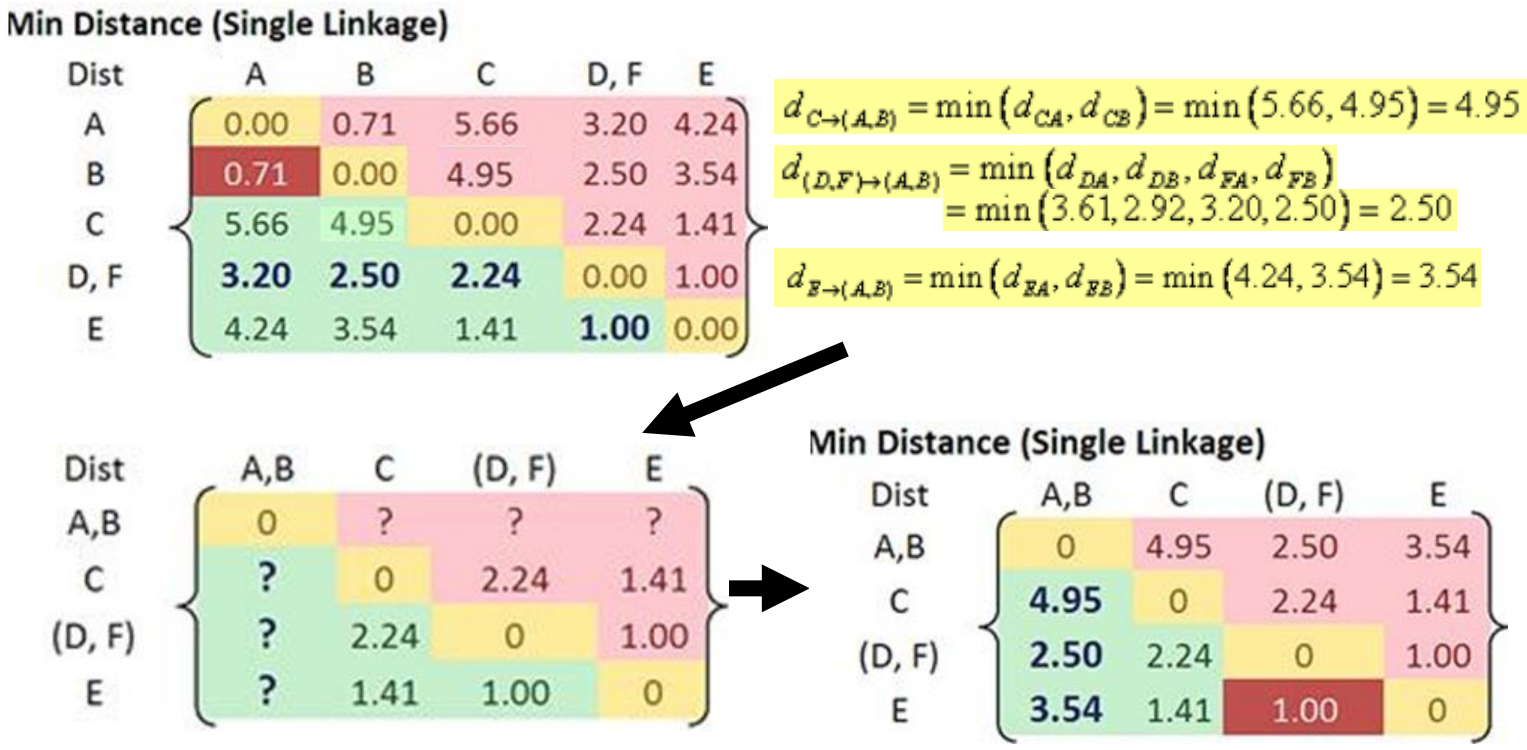
| Dist | A | B | C | D, F | E |
|---|---|---|---|---|---|
| A | 0.00 | 0.71 | 5.66 | 3.20 | 4.24 |
| B | 0.71 | 0.00 | 4.95 | 2.50 | 3.54 |
| C | 5.66 | 4.95 | 0.00 | 2.24 | 1.41 |
| D, F | **3.20** | **2.50** | **2.24** | 0.00 | 1.00 |
| E | 4.24 | 3.54 | 1.41 | **1.00** | 0.00 |

| Dist | A,B | C | (D, F) | E |
|---|---|---|---|---|
| A,B | 0 | ? | ? | ? |
| C | ? | 0 | 2.24 | 1.41 |
| (D, F) | ? | 2.24 | 0 | 1.00 |
| E | ? | 1.41 | 1.00 | 0 |

# Single-link clustering: example – 1



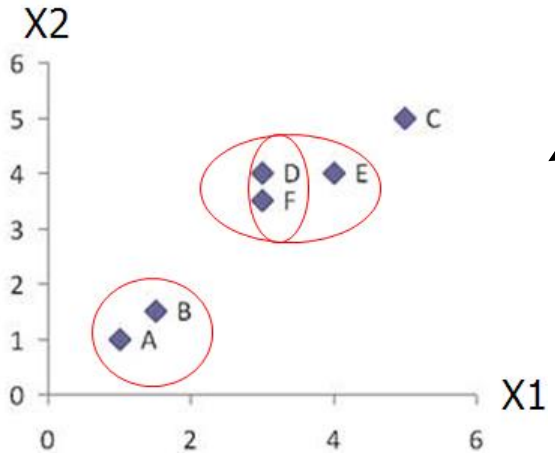
- Update distance matrix (iteration 2)



# Single-link clustering: example – 1



- Merge two closest clusters/update distance matrix (iteration 3)




**Min Distance (Single Linkage)**

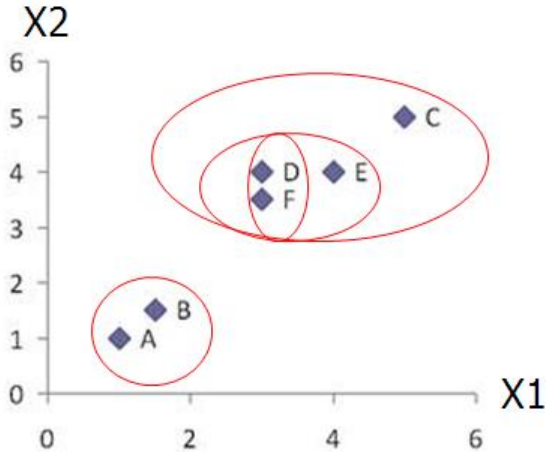
| Dist | A,B | C | (D, F) | E |
|---|---|---|---|---|
| A,B | 0 | 4.95 | 2.50 | 3.54 |
| C | **4.95** | 0 | 2.24 | 1.41 |
| (D, F) | **2.50** | 2.24 | 0 | 1.00 |
| E | **3.54** | 1.41 | 1.00 | 0 |

**Min Distance (Single Linkage)**

| Dist | (A,B) | C | (D, F), E |
|---|---|---|---|
| (A,B) | 0.00 | 4.95 | 2.50 |
| C | 4.95 | 0.00 | 1.41 |
| (D, F), E | **2.50** | **1.41** | 0.00 |

# Single-link clustering: example – 1






**Min Distance (Single Linkage)**

| Dist | (A,B) | C | (D, F), E |
|---|---|---|---|
| (A,B) | 0.00 | 4.95 | 2.50 |
| C | 4.95 | 0.00 | 1.41 |
| (D, F), E | **2.50** | **1.41** | 0.00 |

**Min Distance (Single Linkage)**

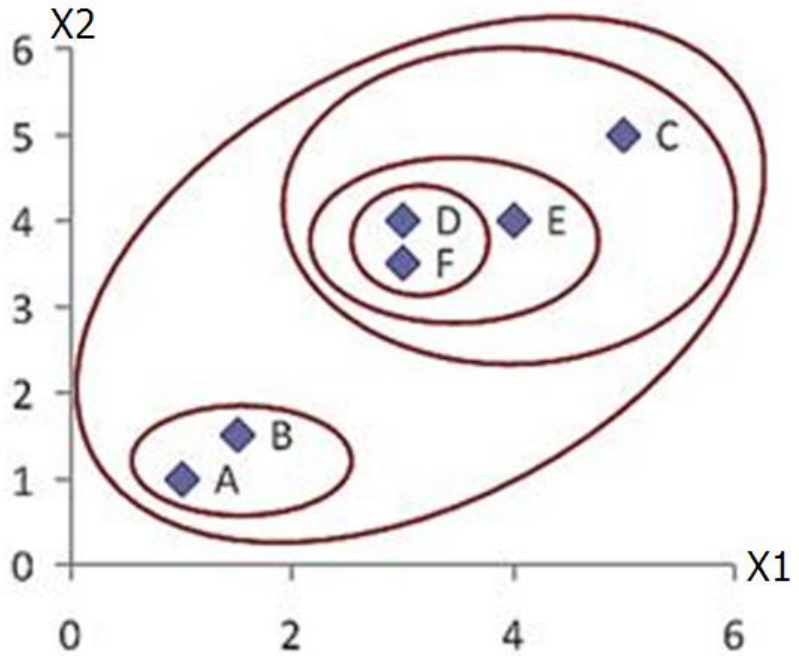
| Dist | (A,B) | ((D, F), E),C |
|---|---|---|
| (A,B) | 0.00 | 2.50 |
| ((D, F), E),C | **2.50** | 0.00 |

- Merge two closest clusters

# Single-link clustering: example – 1



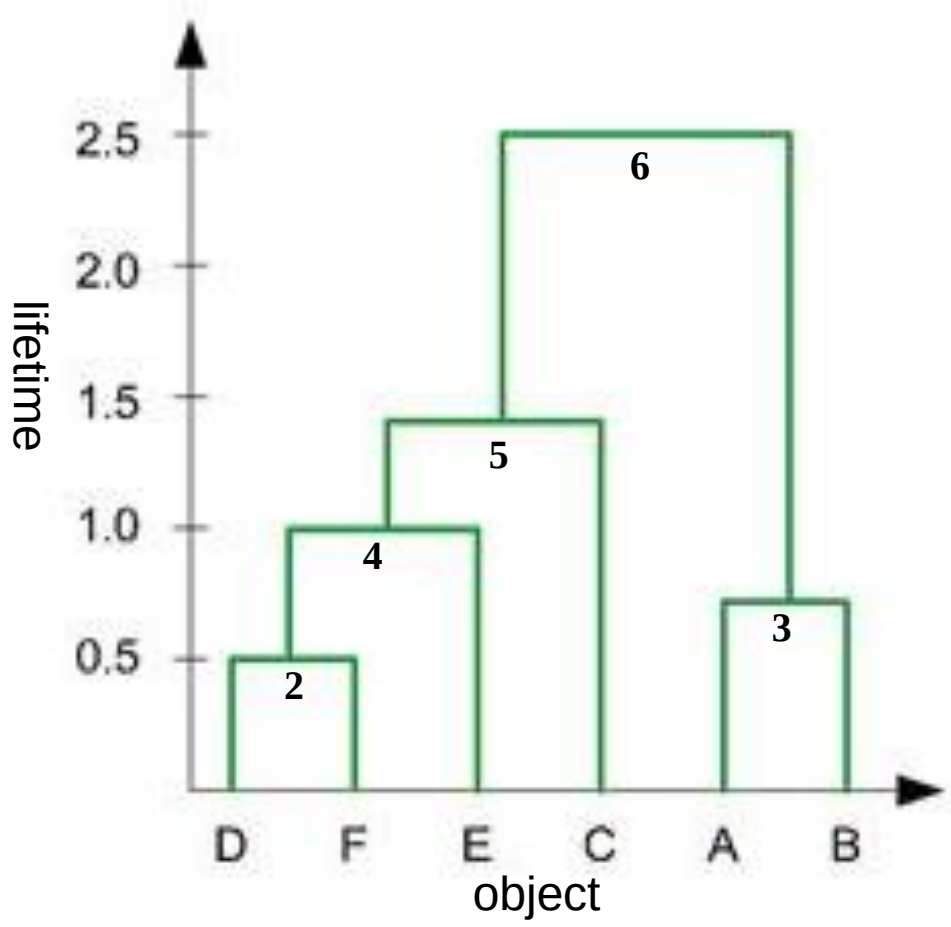
| | X1 | X2 |
|---|---|---|
| A | 1 | 1 |
| B | 1.5 | 1.5 |
| C | 5 | 5 |
| D | 3 | 4 |
| E | 4 | 4 |
| F | 3 | 3.5 |



# Key Concepts in Hierarchal Clustering



- Dendrogram tree representation




1. In the beginning we have 6 clusters: A, B, C, D, E and F
2. We merge clusters D and F into cluster (D, F) at distance 0.50
3. We merge cluster A and cluster B into (A, B) at distance 0.71
4. We merge clusters E and (D, F) into ((D, F), E) at distance 1.00
5. We merge clusters ((D, F), E) and C into (((D, F), E), C) at distance 1.41
6. We merge clusters (((D, F), E), C) and (A, B) into ((((D, F), E), C), (A, B)) at distance 2.50
7. The last cluster contain all the objects, thus conclude the computation

# Single-link clustering: example – 2



# Single-link clustering: example – 2



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

# Single-link clustering: example – 2



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

# Single-link clustering: example – 2



# Single-link clustering: example – 2



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

# Single-link clustering: example – 2



# Single-link clustering: example – 2



| | BA/NA/RM | FI | MI/TO |
|---|---|---|---|
| **BA/NA/RM** | 0 | 268 | 564 |
| **FI** | 268 | 0 | 295 |
| **MI/TO** | 564 | 295 | 0 |

# Single-link clustering: example – 2



# Single-link clustering: example – 2



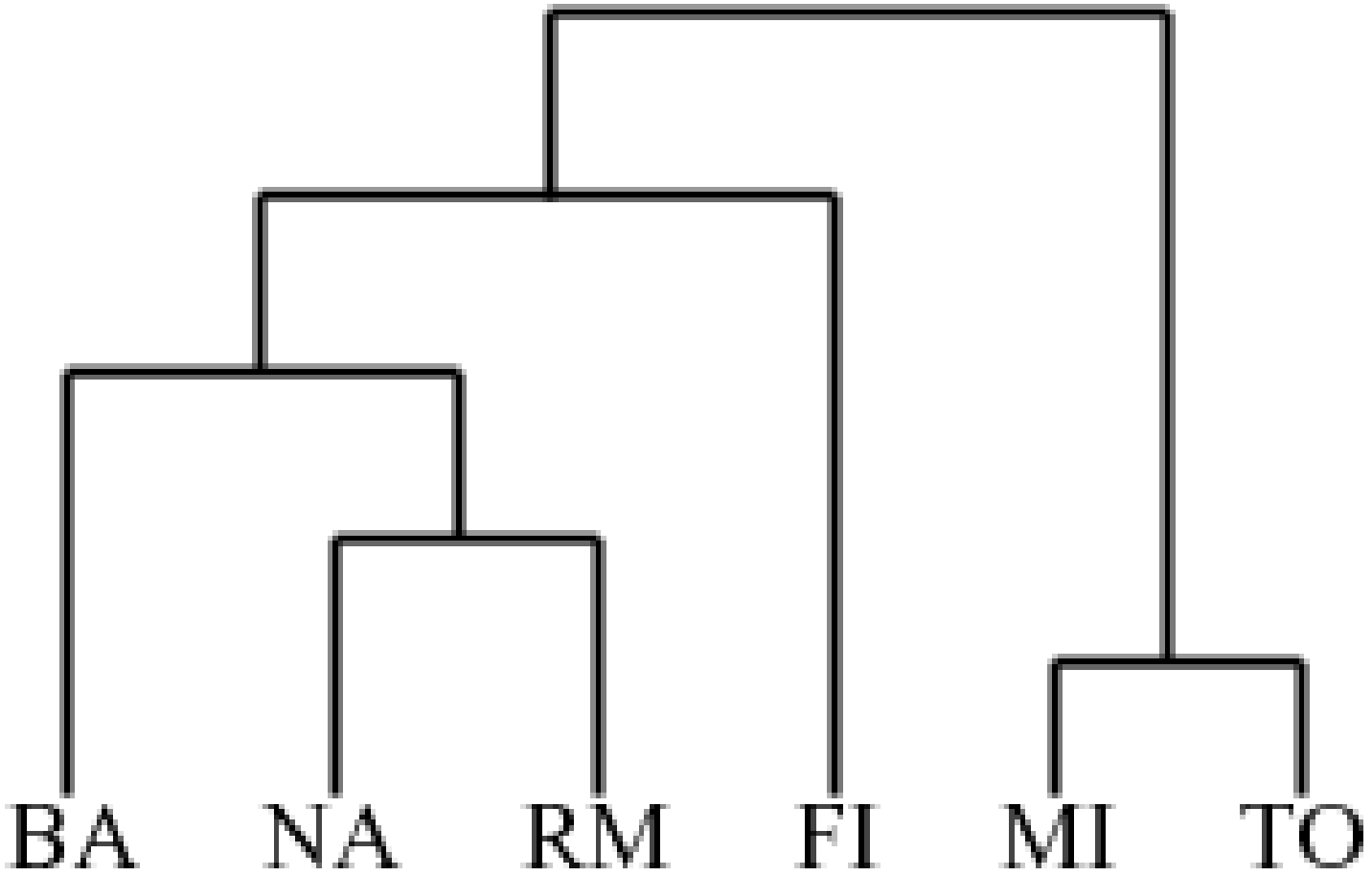
| | **BA/FI/NA/RM** | **MI/TO** |
|---|---|---|
| **BA/FI/NA/RM** | 0 | 295 |
| **MI/TO** | 295 | 0 |

# Single-link clustering: example – 2



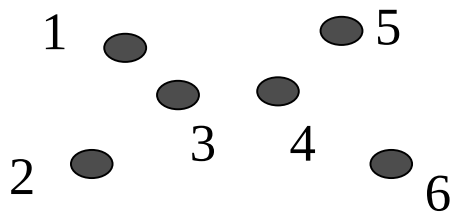
# Single-link clustering: example – 2

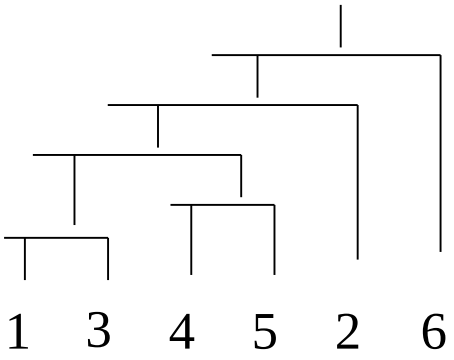




# Result of the Single-Link algorithm

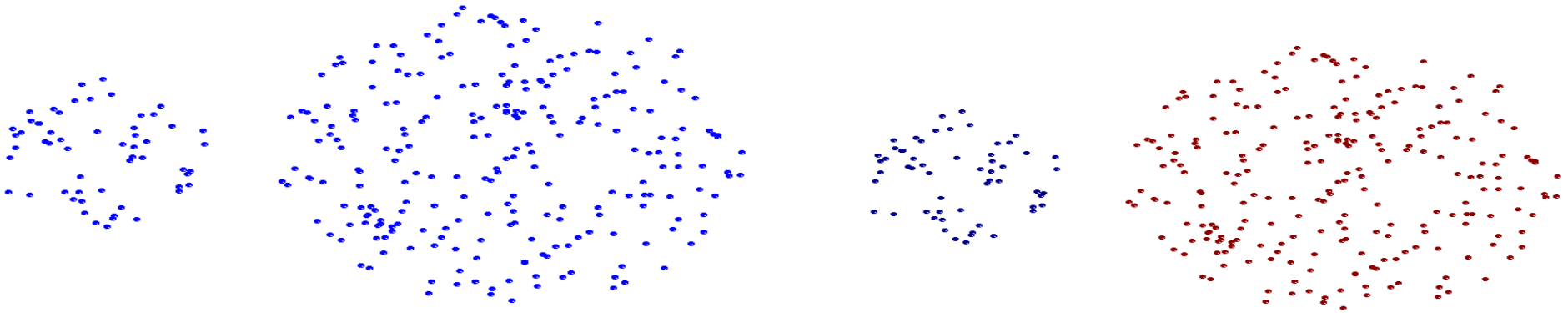




# Strengths of single-link clustering





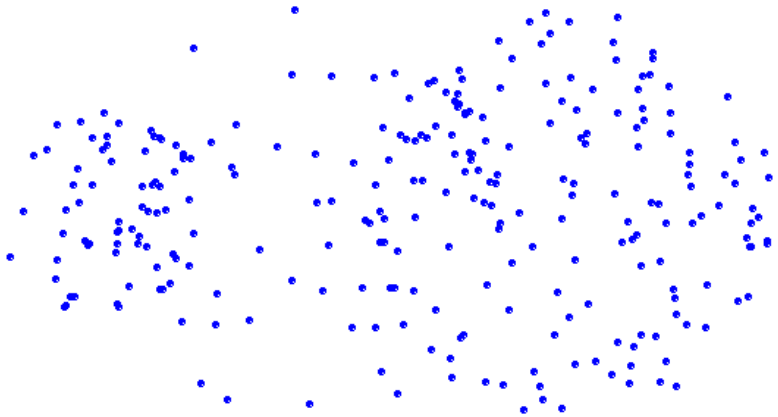
Original Points

Two Clusters

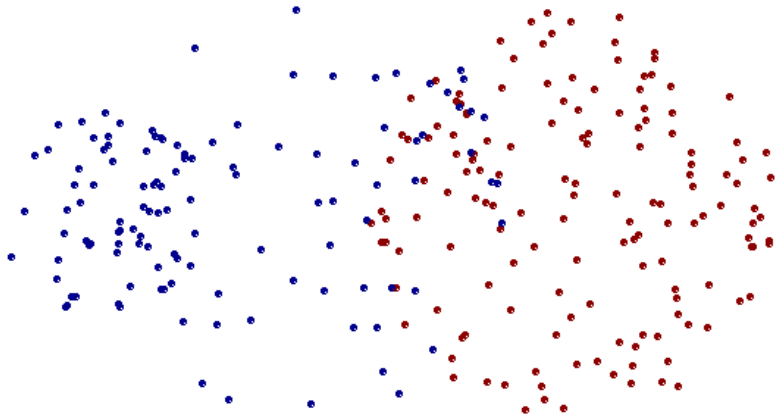
- Can handle non-elliptical shapes

# Limitations of single-link clustering





Original Points



Two Clusters

- Sensitive to noise and outliers
- It produces long, elongated clusters

# Complete-link clustering



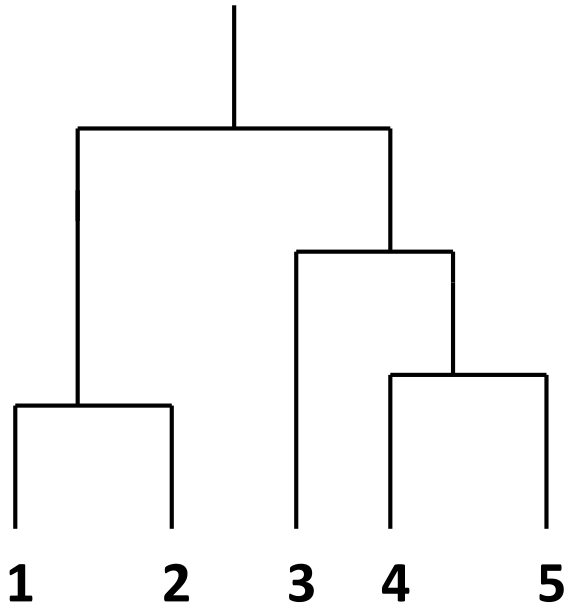
- **Complete-link distance** between clusters $\mathbf{C_i}$ and $\mathbf{C_j}$ is the ***maximum distance*** between any object in $\mathbf{C_i}$ and any object in $\mathbf{C_j}$

- The distance is **defined by the two most dissimilar objects**

$$D_{cl}(C_i, C_j) = \max_{x,y} \{ d(x, y) \mid x \in C_i, y \in C_j \}$$

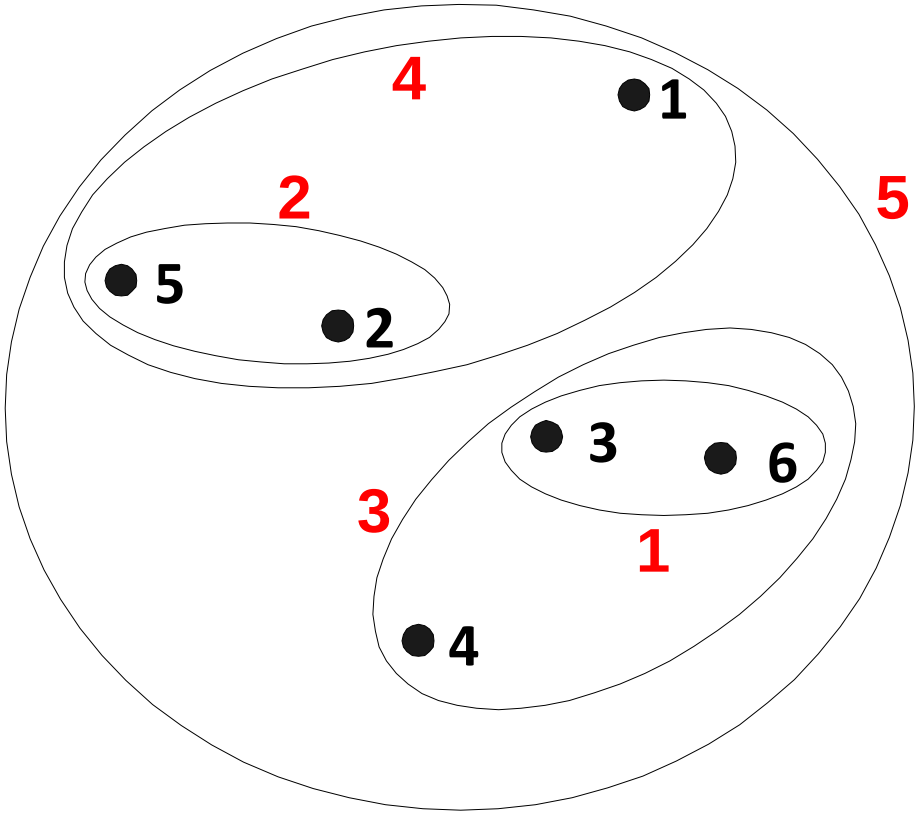
- **Example:** Distance between clusters is determined by the two most distant points in the different clusters

| | I1 | I2 | I3 | I4 | I5 |
|---|---|---|---|---|---|
| I1 | 1.00 | 0.90 | 0.10 | 0.65 | 0.20 |
| I2 | 0.90 | 1.00 | 0.70 | 0.60 | 0.50 |
| I3 | 0.10 | 0.70 | 1.00 | 0.40 | 0.30 |
| I4 | 0.65 | 0.60 | 0.40 | 1.00 | 0.80 |
| I5 | 0.20 | 0.50 | 0.30 | 0.80 | 1.00 |

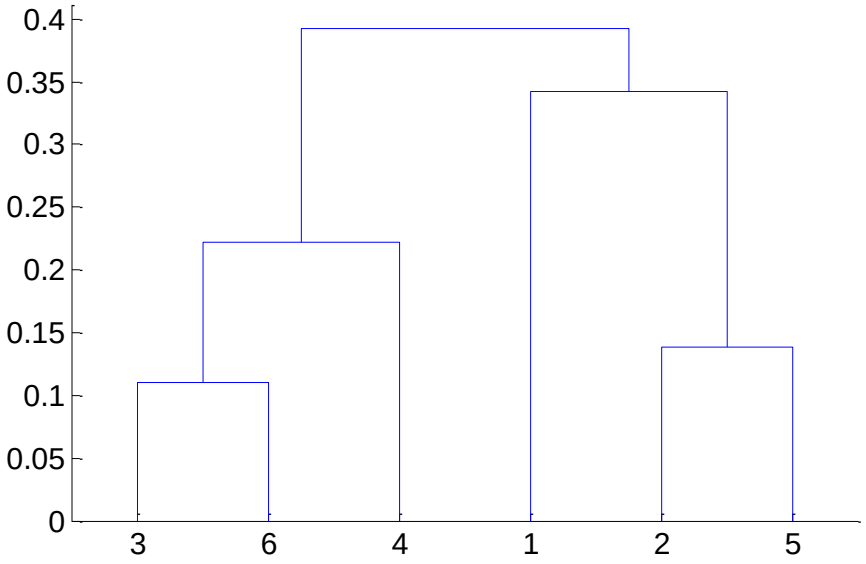


# Complete-link clustering: example





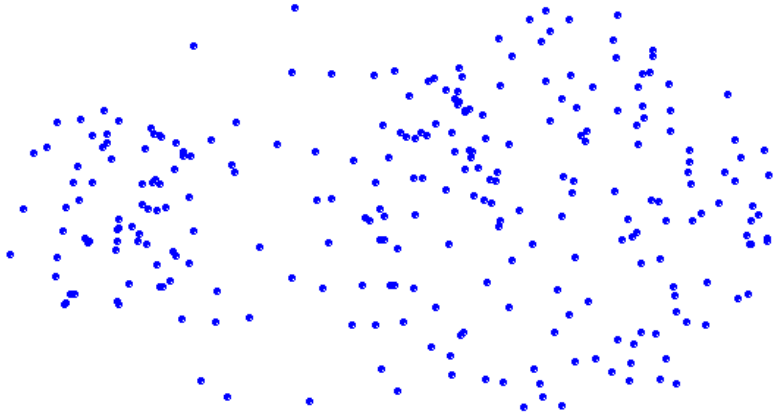

Nested Clusters



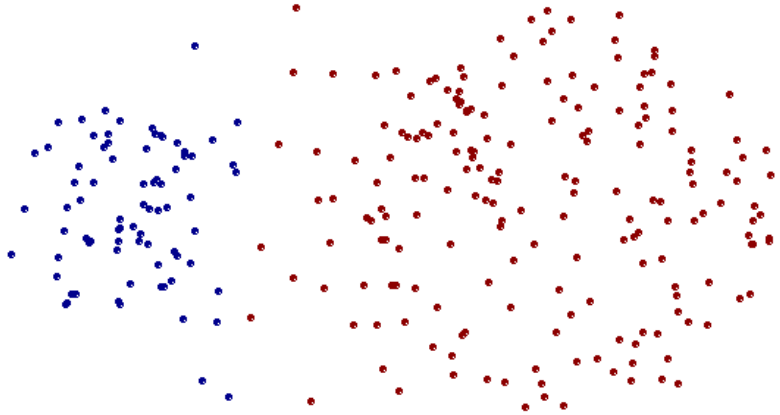

Dendrogram

# Strengths of complete-link clustering





Original Points

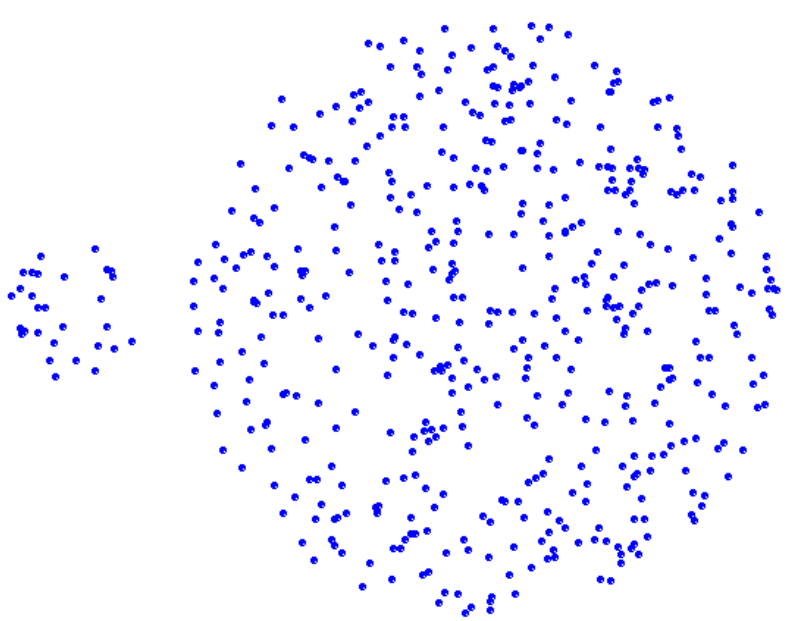


Two Clusters

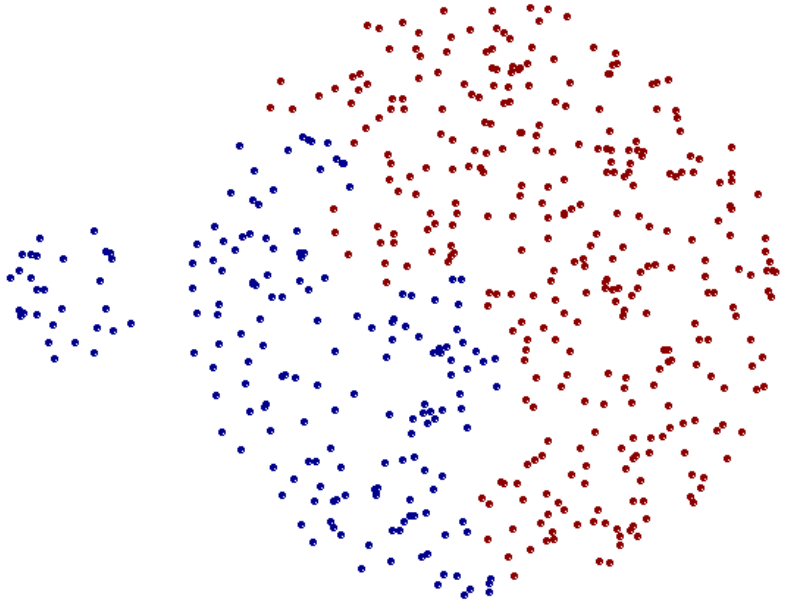
- More balanced clusters (with equal diameter)
- Less susceptible to noise

# Limitations of complete-link clustering





**Original Points**



**Two Clusters**

- Tends to break large clusters
- All clusters tend to have the same diameter – small clusters are merged with larger ones

# Average-link clustering

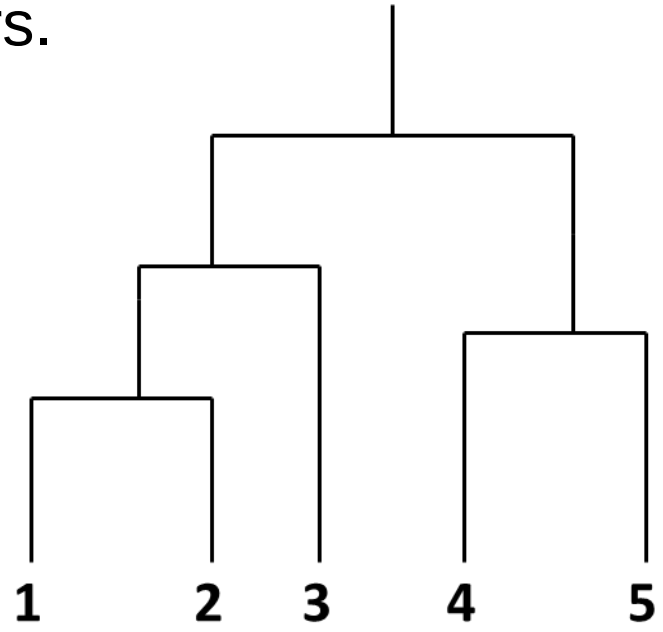


- **Group average distance** between clusters $C_i$ and $C_j$ is the ***average distance*** between any object in $C_i$ and any object in $C_j$

$$D_{avg}(C_i, C_j) = \frac{1}{|C_i| \times |C_j|} \sum_{x \in C_i, y \in C_j} d(x, y)$$

- **Example:** Proximity of two clusters is the average of pairwise proximity between points in the two clusters.

| | I1 | I2 | I3 | I4 | I5 |
|---|---|---|---|---|---|
| I1 | 1.00 | 0.90 | 0.10 | 0.65 | 0.20 |
| I2 | 0.90 | 1.00 | 0.70 | 0.60 | 0.50 |
| I3 | 0.10 | 0.70 | 1.00 | 0.40 | 0.30 |
| I4 | 0.65 | 0.60 | 0.40 | 1.00 | 0.80 |
| I5 | 0.20 | 0.50 | 0.30 | 0.80 | 1.00 |



# Group Average



- Similarity of two clusters = average similarity of all pairs within merged cluster.

$$sim(c_i, c_j) = \frac{1}{|c_i \cup c_j|(|c_i \cup c_j| - 1)} \sum_{\vec{x} \in (c_i \cup c_j)} \sum_{\vec{y} \in (c_i \cup c_j): \vec{y} \neq \vec{x}} sim(\vec{x}, \vec{y})$$

- Compromise between single and complete link.
- Two options:
  - Averaged across all ordered pairs in the merged cluster
  - Averaged over all pairs *between* the two original clusters
- No clear difference in efficacy

# Computing Group Average Similarity



- Always maintain sum of vectors in each cluster.

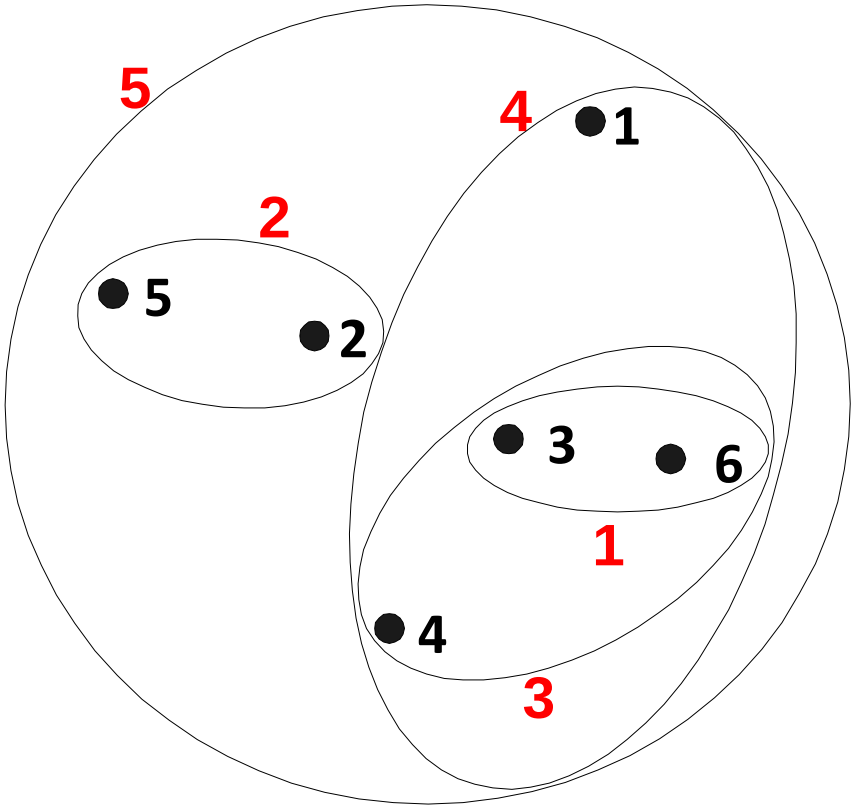
- Compute similarity of clusters in constant time:

$$\vec{s}(c_j) = \sum_{\vec{x} \in c_j} \vec{x}$$

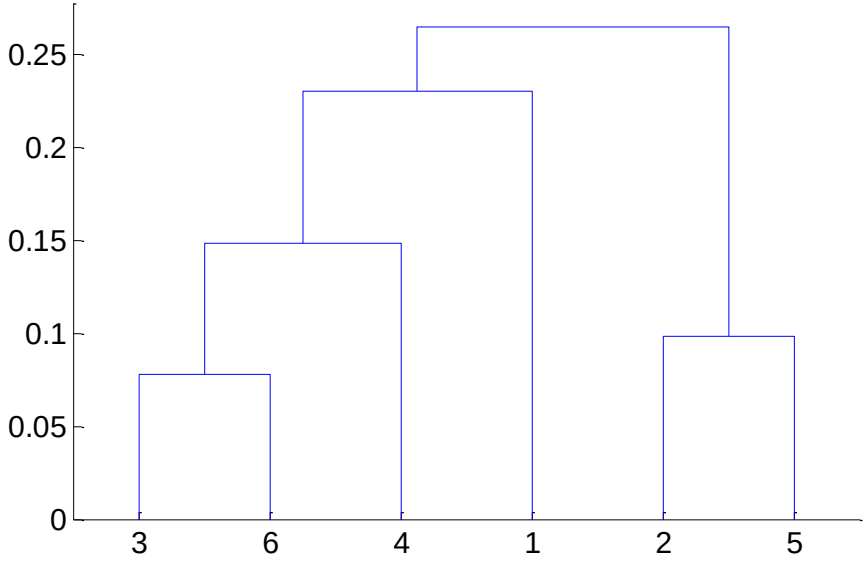
$$sim(c_i, c_j) = \frac{(\vec{s}(c_i) + \vec{s}(c_j)) \bullet (\vec{s}(c_i) + \vec{s}(c_j)) - (|c_i| + |c_j|)}{(|c_i| + |c_j|)(|c_i| + |c_j| - 1)}$$

# Average-link clustering: example






Nested Clusters




Dendrogram

# Average-link clustering: discussion

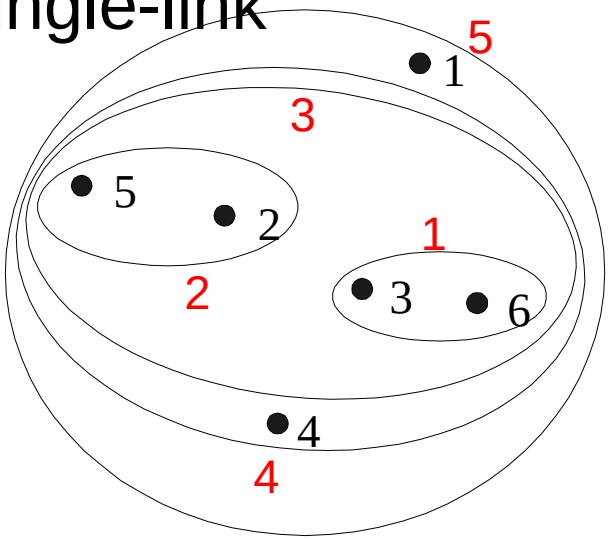


- Compromise between Single and Complete Link
- Strengths
  - Less susceptible to noise and outliers
- Limitations
  - Biased towards globular clusters

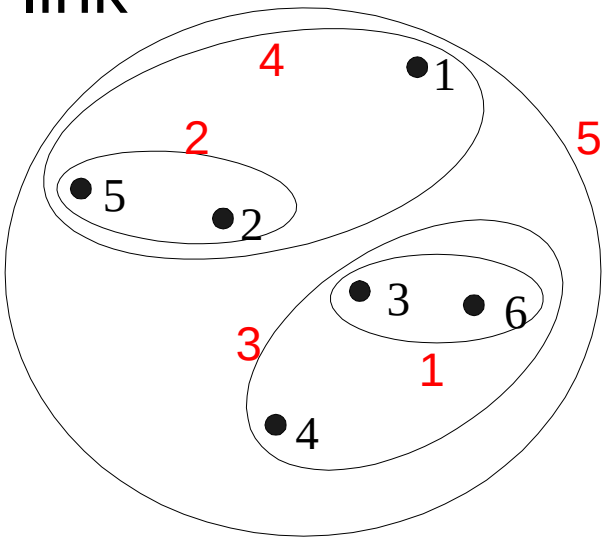
# Hierarchical Clustering: Comparison



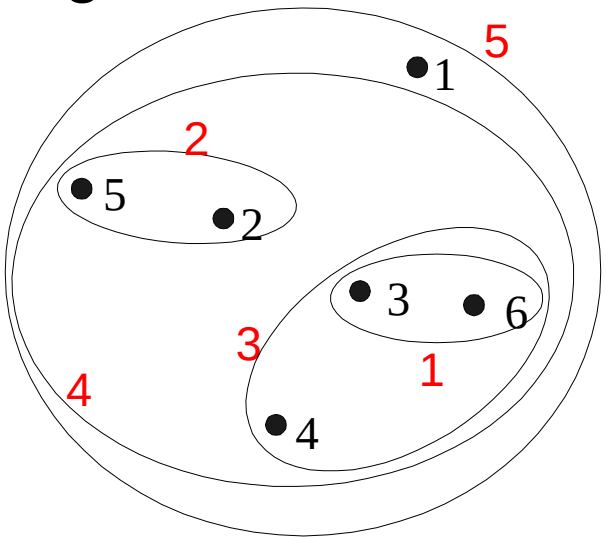
## Single-link



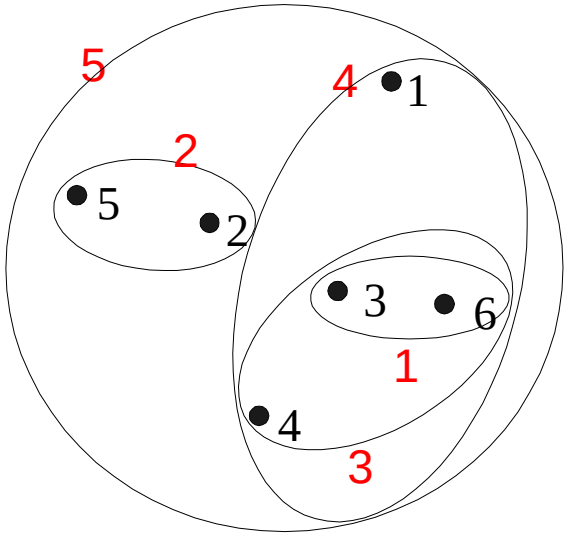

## Complete-link




## Average-link



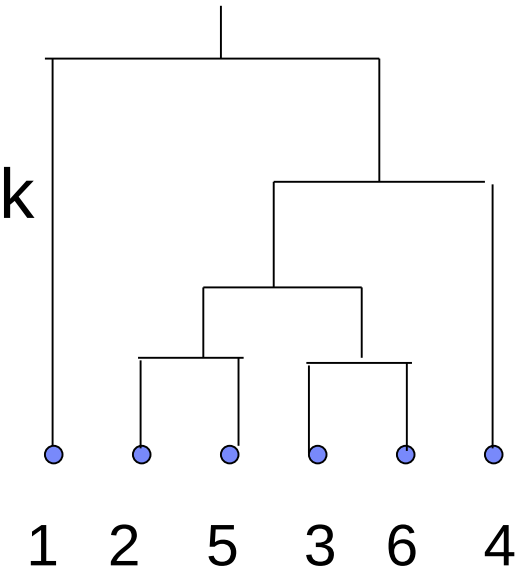

## Centroid distance



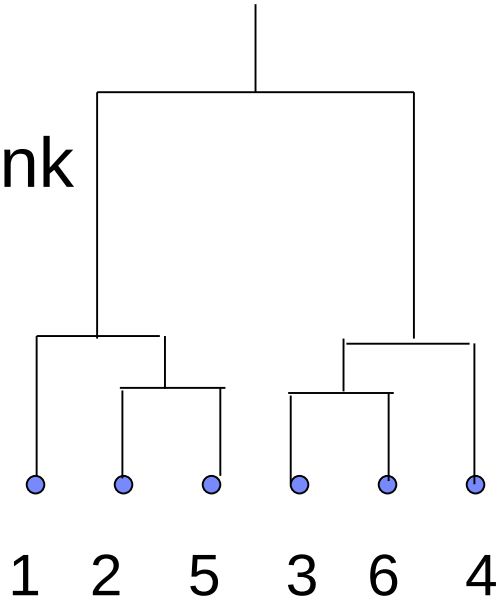
# Compare Dendrograms



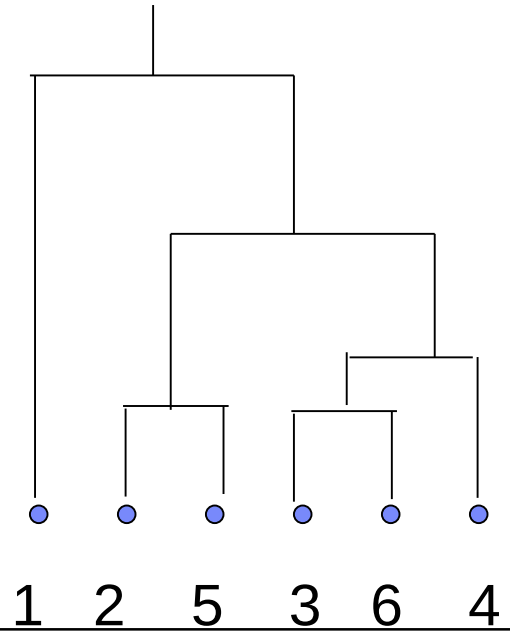
Single-link



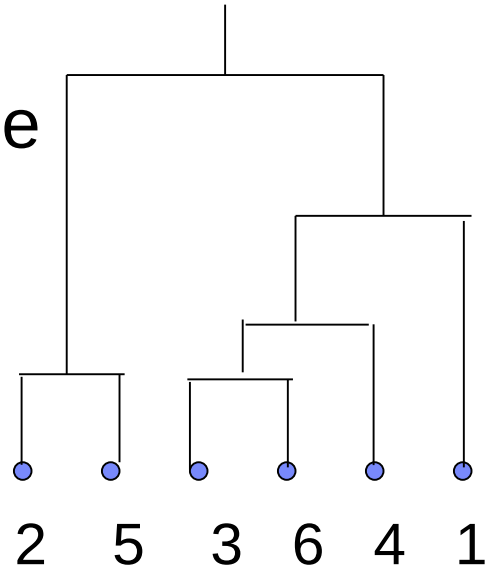

Complete-link




Average-link

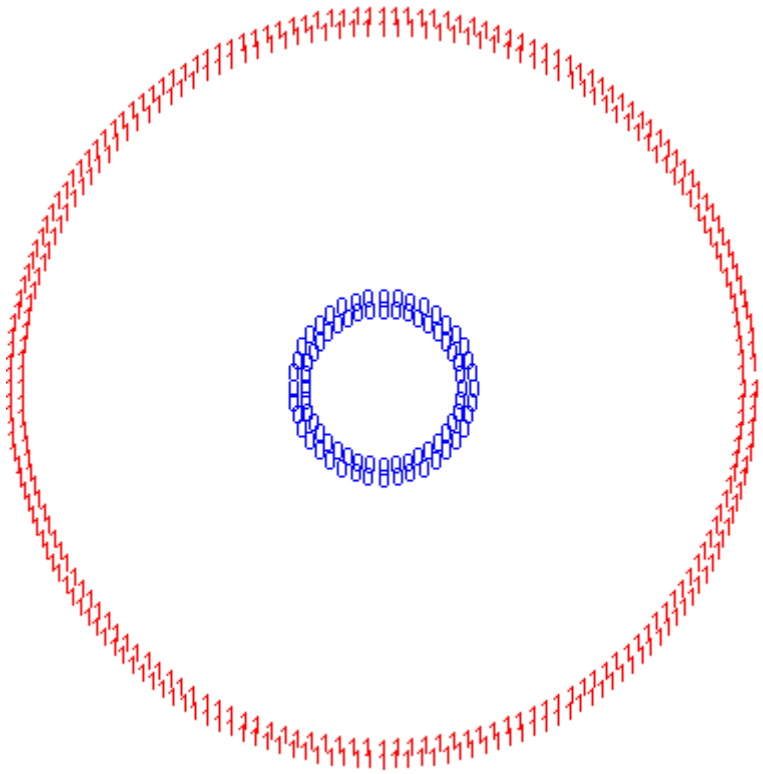



Centroid distance

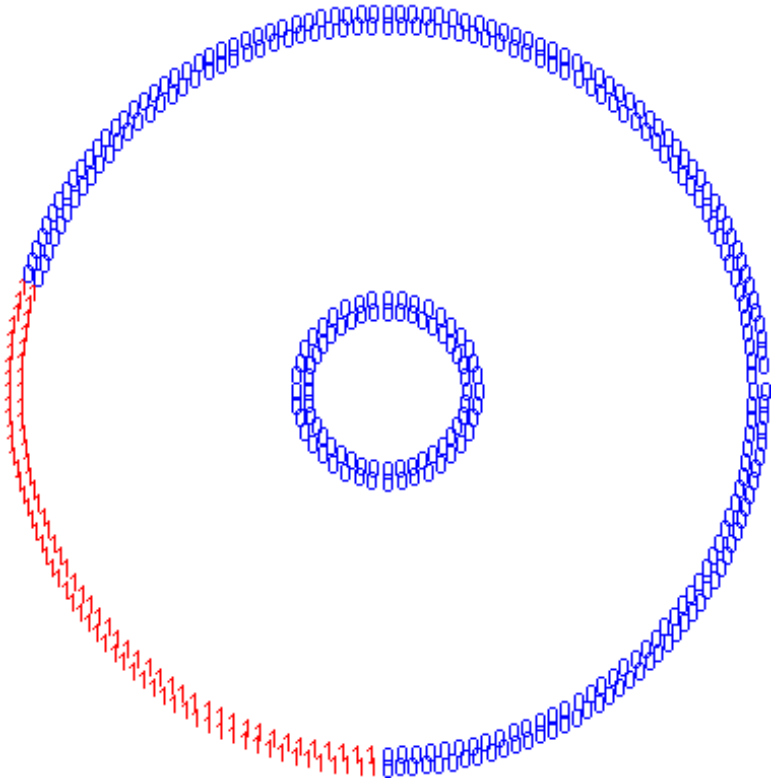


# Effect of Bias towards Spherical Clusters





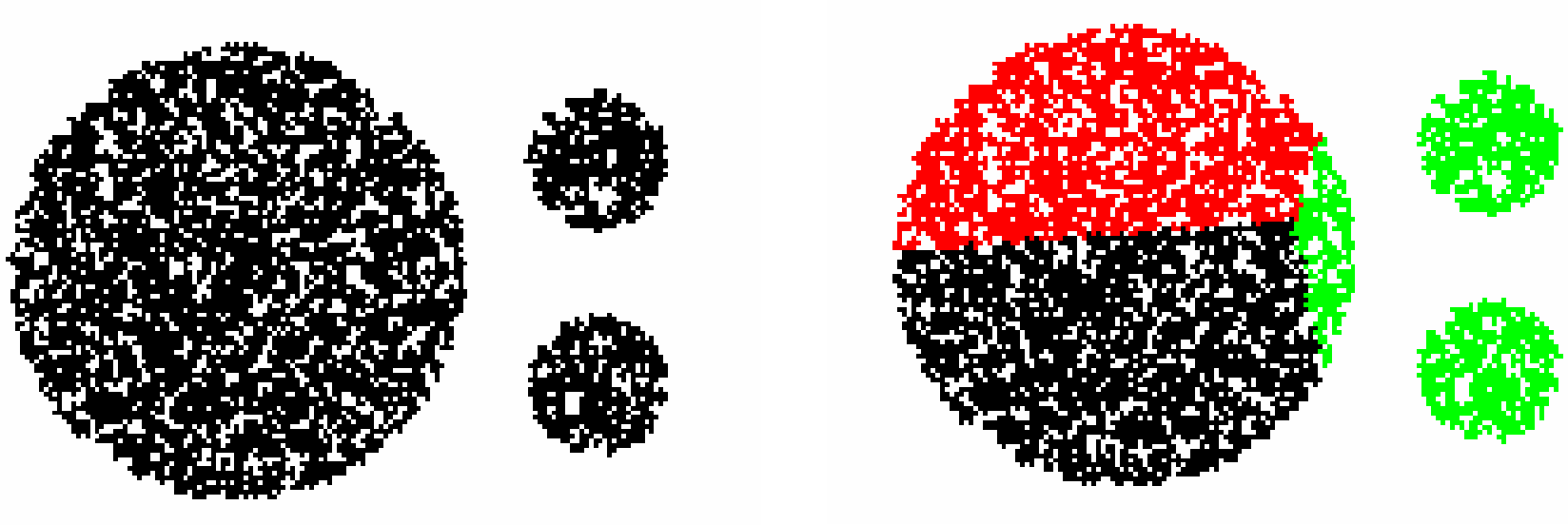
Single-link (2 clusters)



Complete-link (2 clusters)

# Limitation of Complete-Link, Average-Link, and Centroid Distance



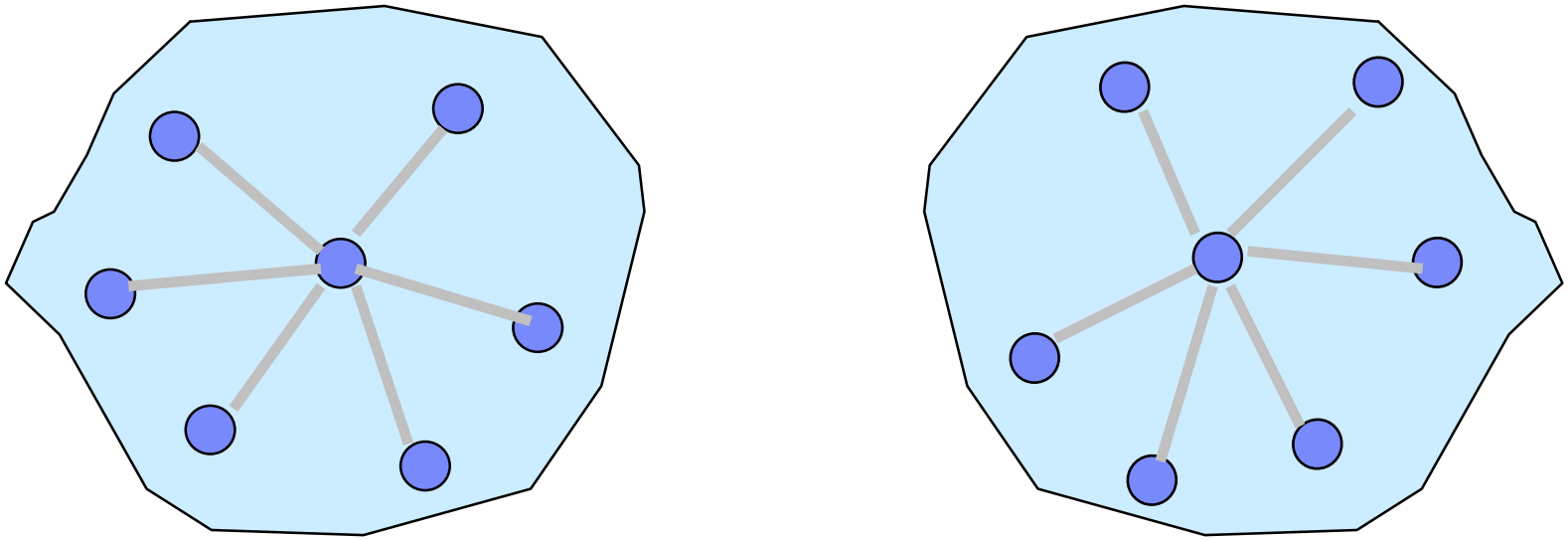


The complete-link, average-link, or centroid distance method tend to break the large cluster.

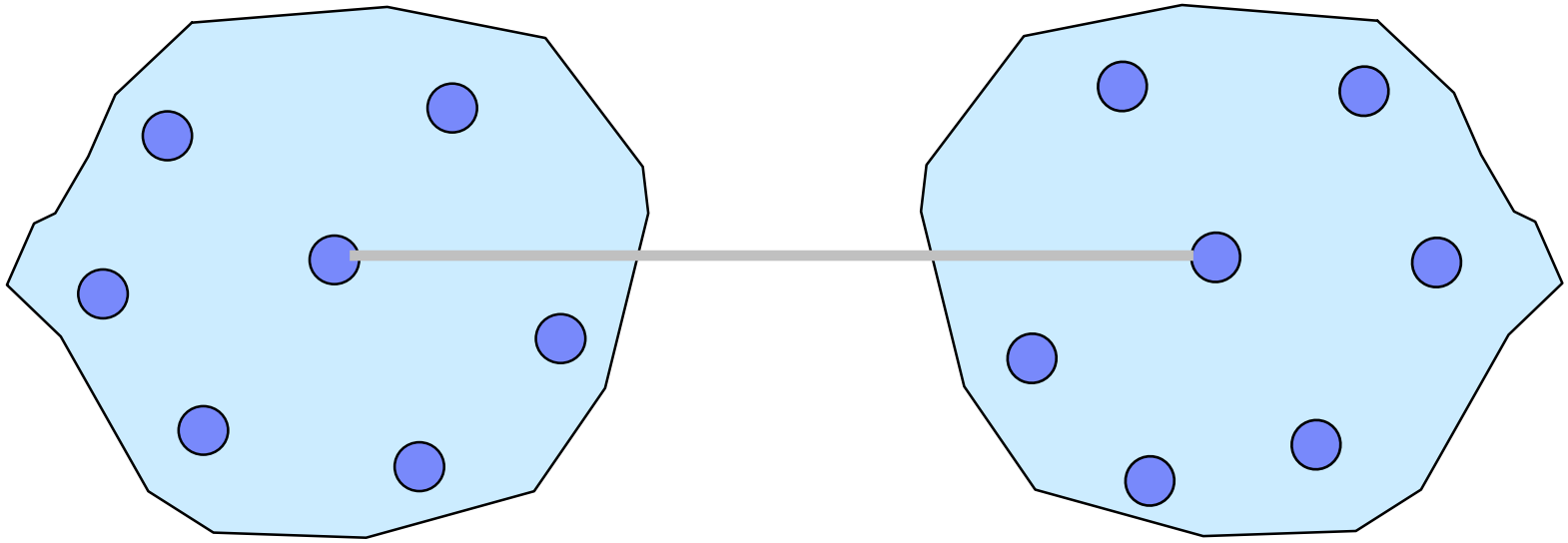
# Other Agglomerative Clustering Methods



## Ward's Procedure



## Centroid Method



# Distance between two clusters



- **Centroid distance** between clusters $C_i$ and $C_j$ is the distance between the centroid $r_i$ of $C_i$ and the centroid $r_j$ of $C_j$

$$D_{centroids}(C_i, C_j) = d(r_i, r_j)$$

- **Ward's distance** between clusters $C_i$ and $C_j$ is the difference between the total within cluster sum of squares for the two clusters separately, and the within cluster sum of squares resulting from merging the two clusters in cluster $C_{ij}$
  - $r_i$: centroid of $C_i$
  - $r_j$: centroid of $C_j$
  - $r_{ij}$: centroid of $C_{ij}$

$$D_w(C_i, C_j) = \sum_{x \in C_i} (x - r_i)^2 + \sum_{x \in C_j} (x - r_j)^2 - \sum_{x \in C_{ij}} (x - r_{ij})^2$$

# Ward's distance for clusters



- Similar to group average and centroid distance
- Less susceptible to noise and outliers
- Biased towards globular clusters
- Hierarchical analogue of k-means
  - Can be used to initialize k-means

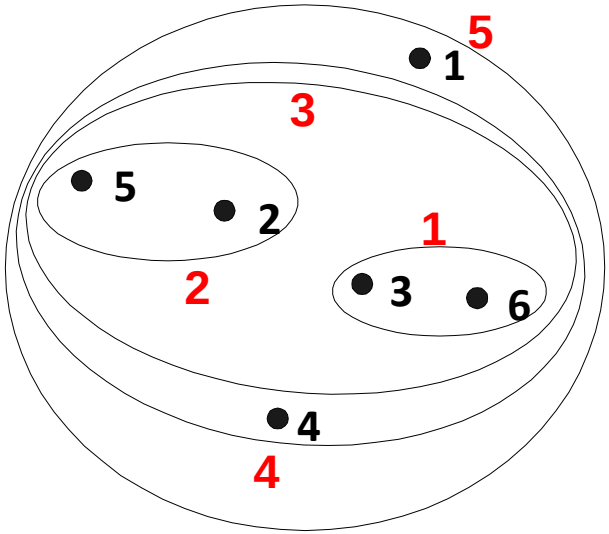
# Hierarchical Clustering: Time and Space requirements



- For a dataset $X$ consisting of $n$ points
- $O(n^2)$ space; it requires storing the distance matrix
- $O(n^3)$ time in most of the cases
  - There are $n$ steps and at each step the size $n^2$ distance matrix must be updated and searched
  - Complexity can be reduced to $O(n^2 \log(n))$ time for some approaches by using appropriate data structures

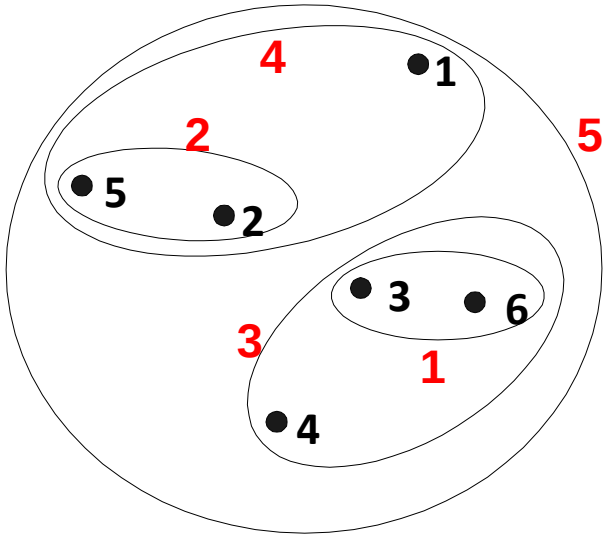
# Hierarchical Clustering: Comparison

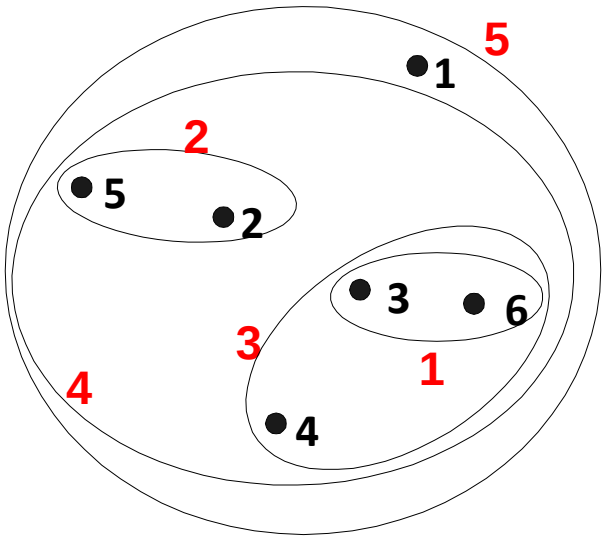





MIN

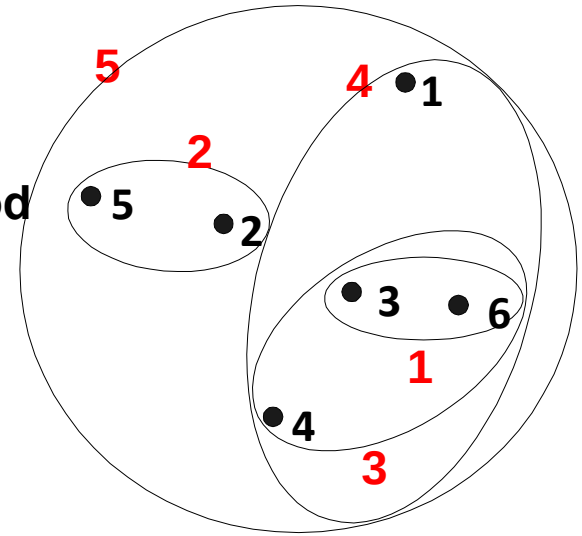
MAX




Group Average

Ward's Method



# Strengths of Hierarchical Clustering



- No assumptions on the number of clusters
  - Any desired number of clusters can be obtained by ‘cutting’ the dendogram at the proper level
- Hierarchical clustering may correspond to meaningful taxonomies
  - Example in biological sciences (e.g., phylogeny reconstruction, etc), web (e.g., product catalogs) etc.
- Complexity of hierarchical clustering
  - Distance matrix is used for deciding which clusters to merge/split
  - At least quadratic in the number of data points
  - Not usable for large datasets

# Limitations of Hierarchical Clustering



- Dendrograms are most useful when there a small number of observations (cases) to cluster.
- The Agglomerative procedure works for larger data sets but is computing intensive in that (nxn) matrices are the basic building blocks for the Agglomerative procedure.
- Hierarchical Clustering only makes one pass through the data. Therefore, early clustering decisions affect the rest of the clustering results.
- Hierarchical Clusters often have low stability. That is, adding or subtracting variables or adding or dropping observations can affect the groupings substantially.
- The determination of final clusters can be sensitive to outliers and their treatment.

# Divisive hierarchical clustering



- Start with a single cluster composed of all data points
- Split this into components
- Continue recursively
- *Monothetic* divisive methods split clusters using one variable/dimension at a time
- *Polythetic* divisive methods make splits on the basis of all variables together
- Any intercluster distance measure can be used
- Computationally intensive, less widely used than agglomerative methods

# Exercise



Given a data set of five objects characterised by a single continuous feature:

The distance matrix on this dataset is given below. Apply the agglomerative algorithm with single-link, complete-link and averaging cluster distance measures to produce three dendrogram trees, respectively.

| | a | b | C | d | e |
|---|---|---|---|---|---|
| Feature | 1 | 2 | 4 | 5 | 6 |

| | a | b | c | d | e |
|---|---|---|---|---|---|
| **a** | 0 | 1 | 3 | 4 | 5 |
| **b** | 1 | 0 | 2 | 3 | 4 |
| **c** | 3 | 2 | 0 | 1 | 2 |
| **d** | 4 | 3 | 1 | 0 | 1 |
| **e** | 5 | 4 | 2 | 1 | 0 |

# Partitioning Algorithms



- Partitioning method: Construct a partition of *n* documents into a set of *K* clusters
- Given: a set of documents and the number *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, 1967): Each cluster is represented by the center of the cluster
  - k-medoids or PAM (Partition around medoids) (Kaufman & Rousseeuw, 1987): Each cluster is represented by one of the objects in the cluster

# K – means clustering



- *K*-means clustering is a type of unsupervised learning, which is used when we have unlabeled data (i.e., data without defined categories or groups).
- *K* means algorithm will divide the given data into *K* clusters.
- This algorithm works iteratively to assign each data point to one of *K* groups.
- In order to divide the data into groups it utilises the unique feature of data objects.
- Data points are clustered based on feature match score. Match score is a measure of similarity or dissimilarity.
- The output of *K* –means algorithms are the *K* clusters and the centroids of each of the clusters.

# K – means clustering



- Let S $= \{x_1, x_2, \ldots x_i \ldots x_m\} \in \mathbb{R}^n$ be the given set of *m* elements of *n*-dimension. By learning the structures and regularities in the given sample of data the algorithm partitions the set *S* into *K* subsets (clusters) say, $C_1, C_2, \ldots C_i \ldots C_k$ satisfying the following criteria.
- Clusters are pair wise disjoint. i.e. $C_i \cap C_i = \emptyset,\ i \neq j$.
- Clusters should span the data set *S*. i.e. $\bigcup_i C_i = S$.
- Each cluster must contain similar *data* items.
- Let $\alpha_i$ be the centroids of clusters $C_i$ initially. Then, the objective function is given by,

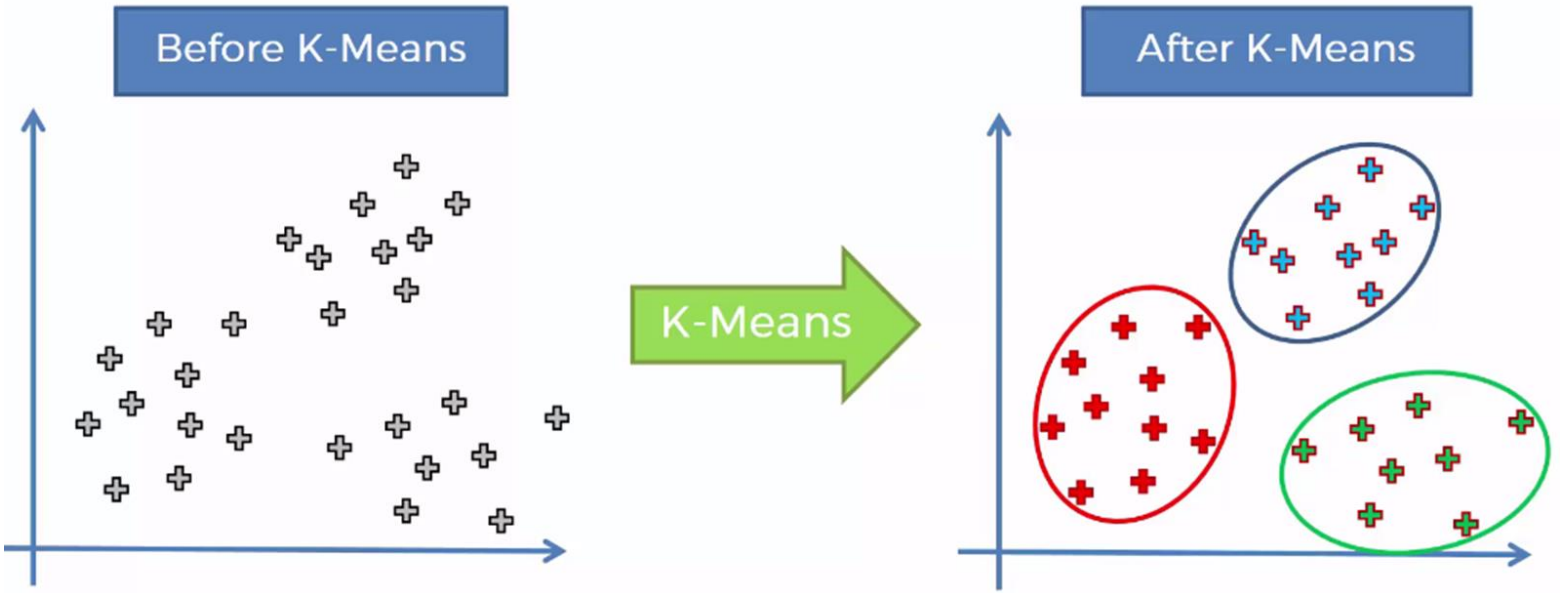
$$f(k) = \sum_{i=1}^{K} \sum_{x_j \in C_j} \left\| x_j - \alpha_i \right\|^2$$

- Task of the algorithm is now reduces to minimise the objective function at each iteration.

# Algorithm



- Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
- Assign each object to the group that has the closest centroid.
- When all objects have been assigned, recalculate the positions of the K centroids.
- Repeat the above steps until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.



# Variations of the *K-Means* Method

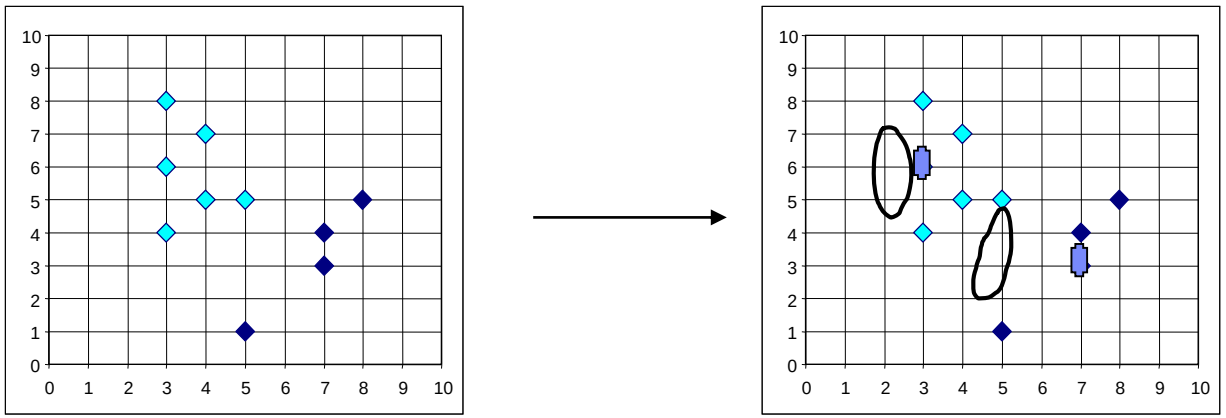


- A few variants of the *k-means* which differ in
  - Selection of the initial *k* means
  - Dissimilarity calculations
  - Strategies to calculate cluster means
- Handling categorical data: *k-modes* (Huang'98)
  - Replacing means of clusters with modes
  - Using new dissimilarity measures to deal with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

# What is the problem of k-Means Method?

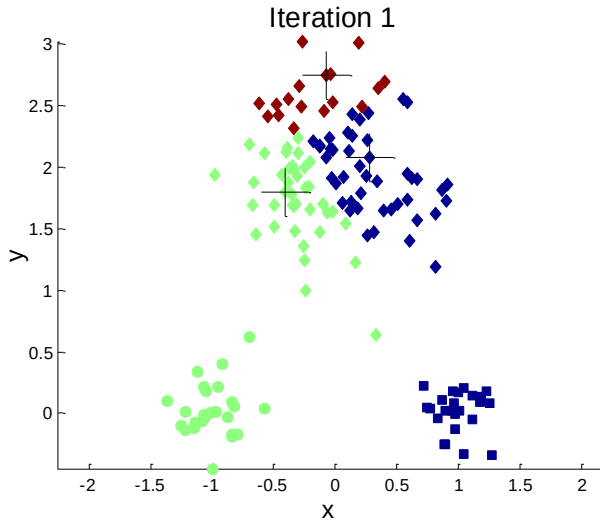


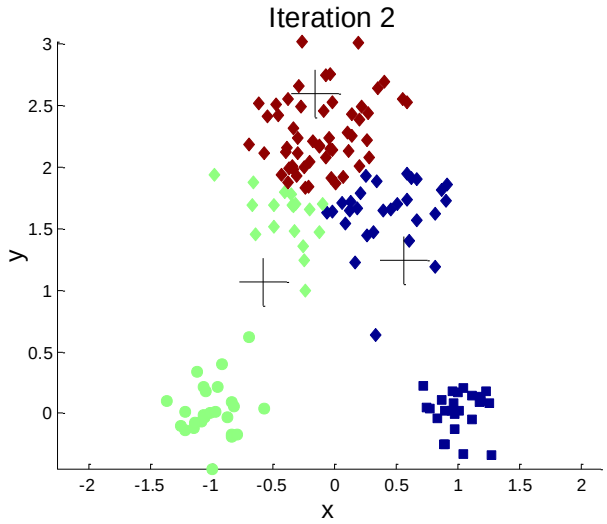
- The k-means algorithm is sensitive to outliers !
  - Since an object with an extremely large value may substantially distort the distribution of the data.
- K-Medoids: Instead of taking the **mean** value of the object in a cluster as a reference point, **medoids** can be used, which is the **most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

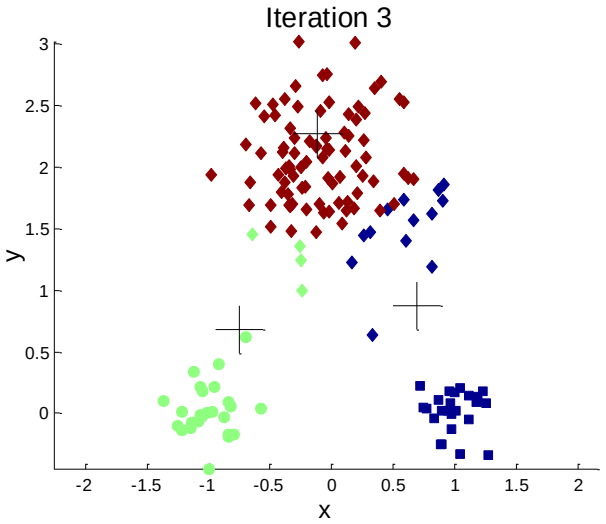


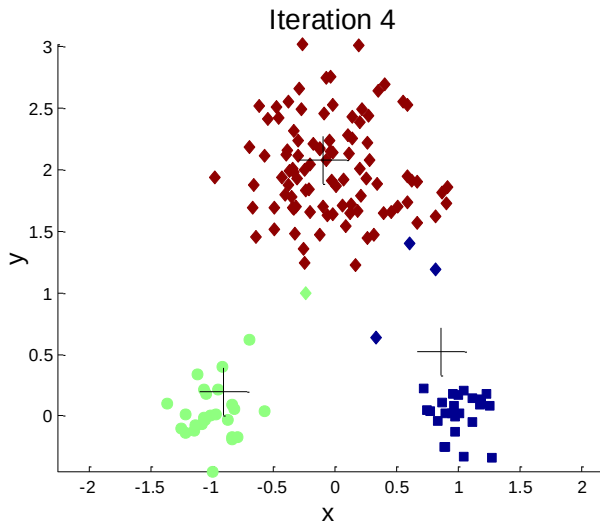
# Importance of Choosing Initial Centroids

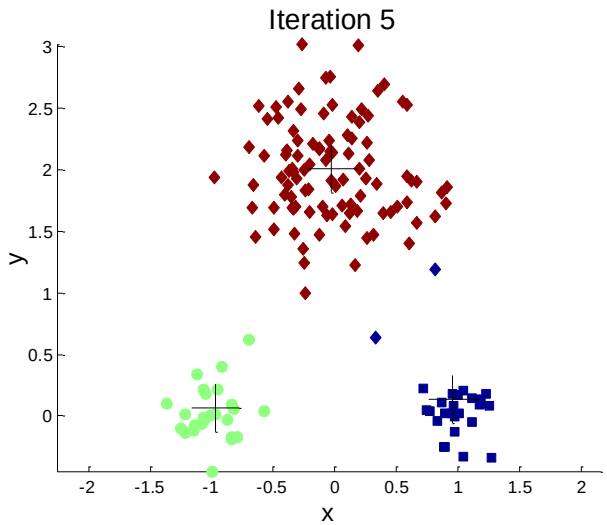


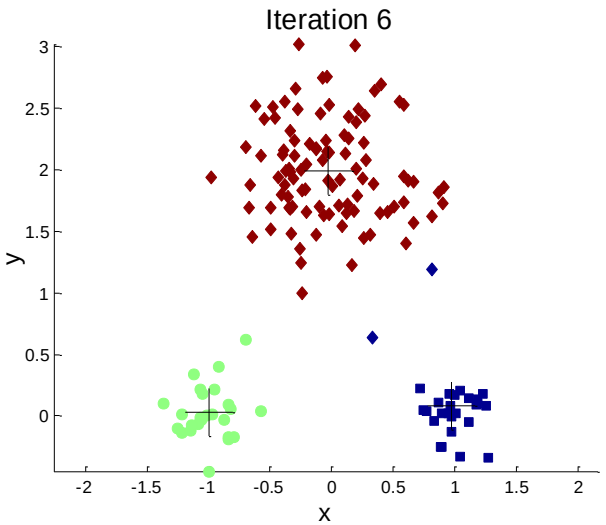


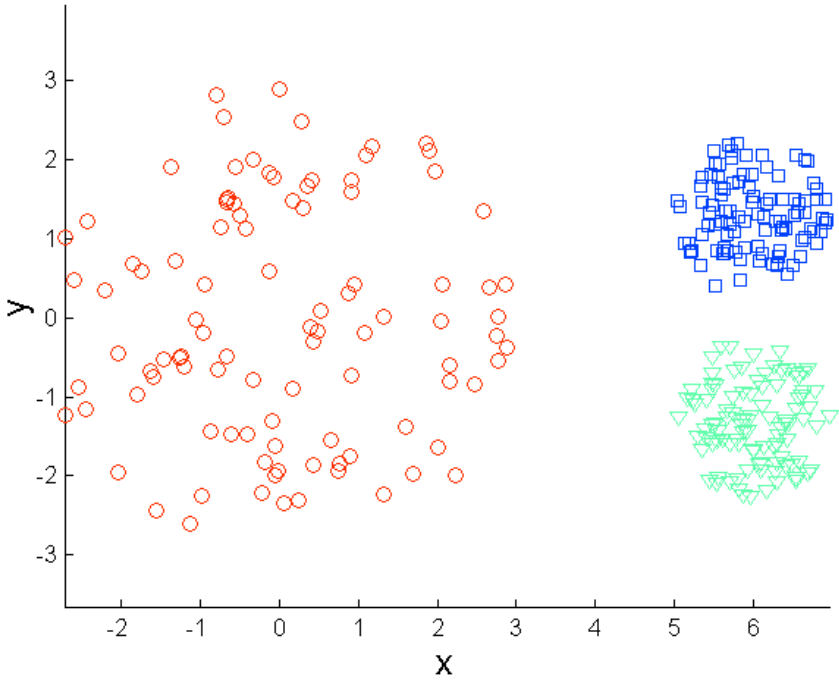
# Solutions to Initial Centroids Problem



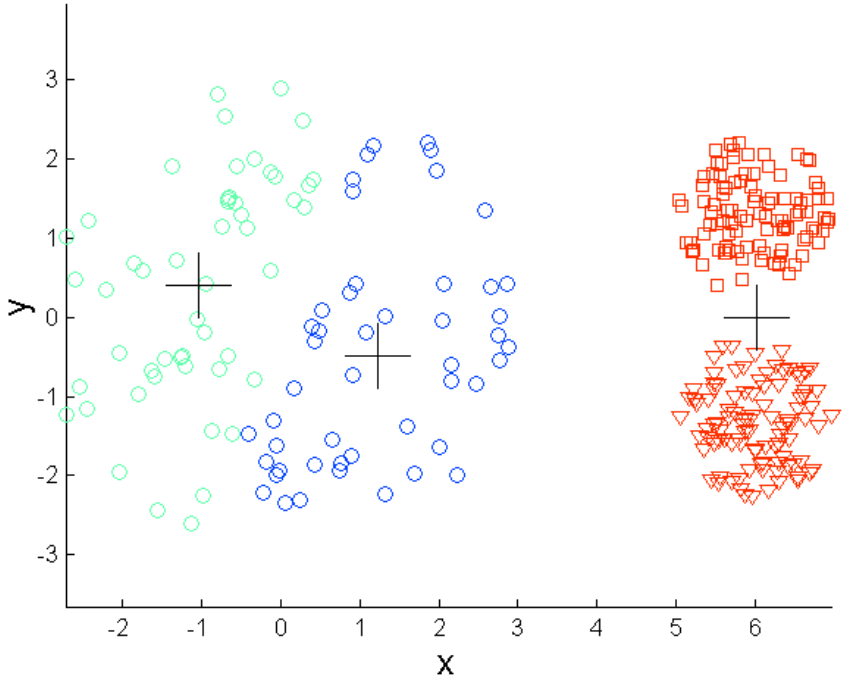
- Multiple runs
  - Helps, but probability is not on your side
- Sample and use hierarchical clustering to determine initial centroids
- Select more than k initial centroids and then select among these initial centroids
  - Select most widely separated
- Post-processing
- Bisecting K-means
  - Not as susceptible to initialization issues

# Limitations of K-means: Differing Density





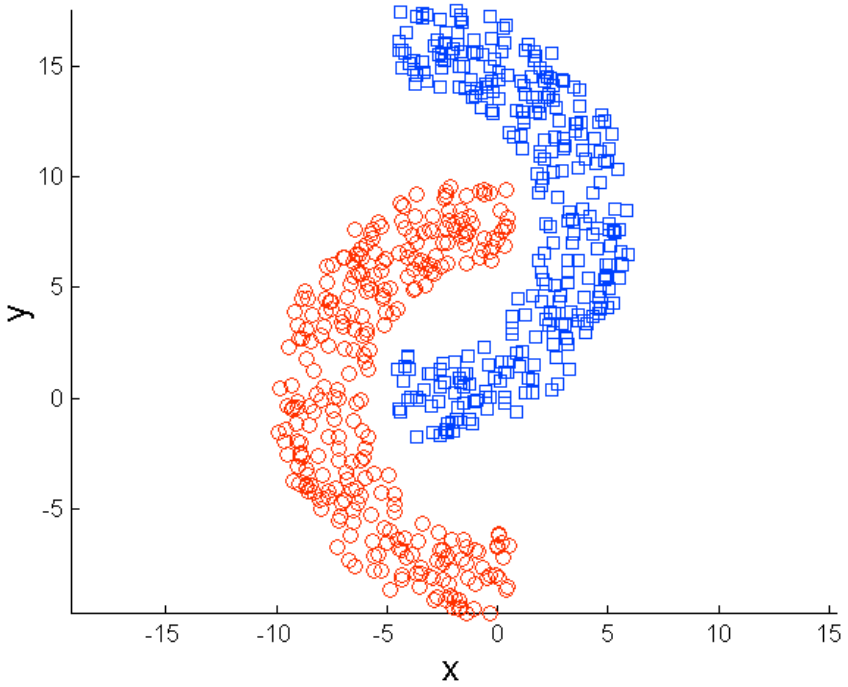

Original Points



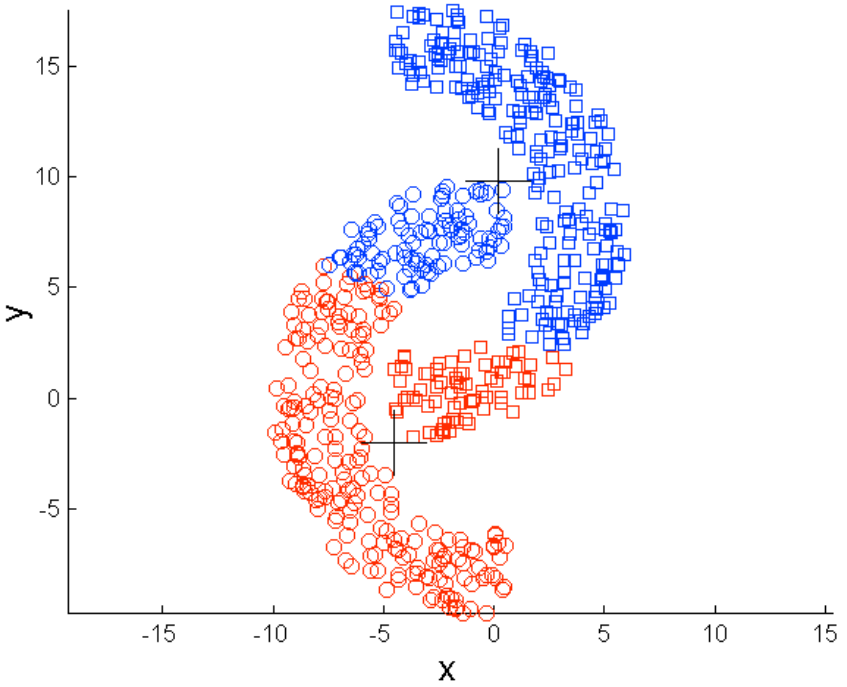

K-means (3 Clusters)

# Limitations of K-means: Non-globular Shapes






Original Points




K-means (2 Clusters)

# The *K-Medoids* Clustering Method

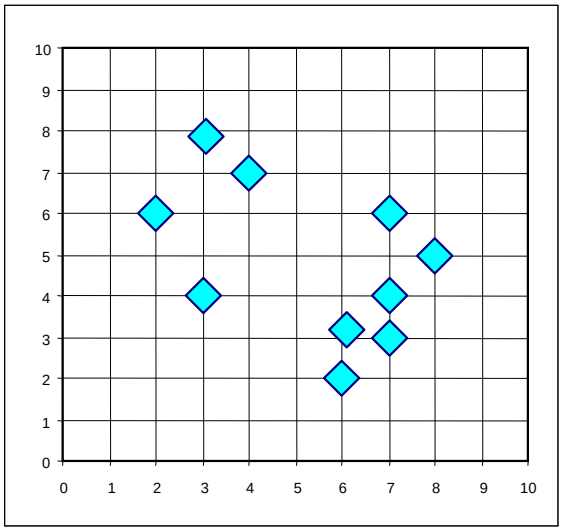


- Find *representative* objects, called medoids, in clusters
- *PAM* (Partitioning Around Medoids, 1987)
  - starts from an initial set of medoids and iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering
  - *PAM* works effectively for small data sets, but does not scale well for large data sets
- *CLARA* (Kaufmann & Rousseeuw, 1990)
- *CLARANS* (Ng & Han, 1994): Randomized sampling
- Focusing + spatial data structure (Ester et al., 1995)

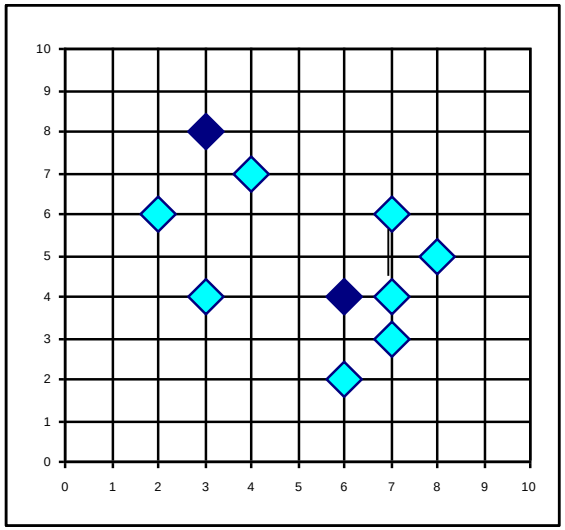
# Typical k-medoids algorithm (PAM)



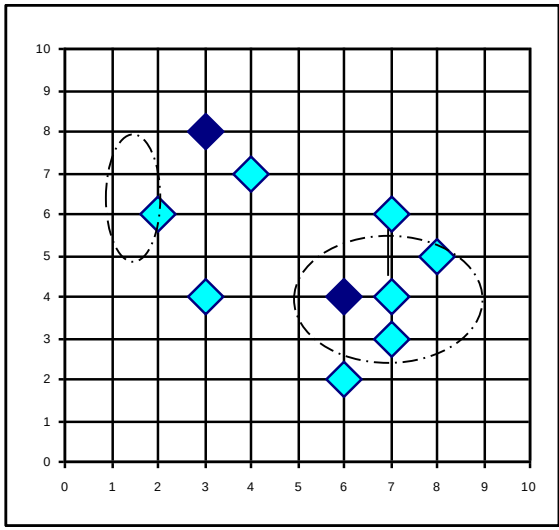
Total Cost = 20



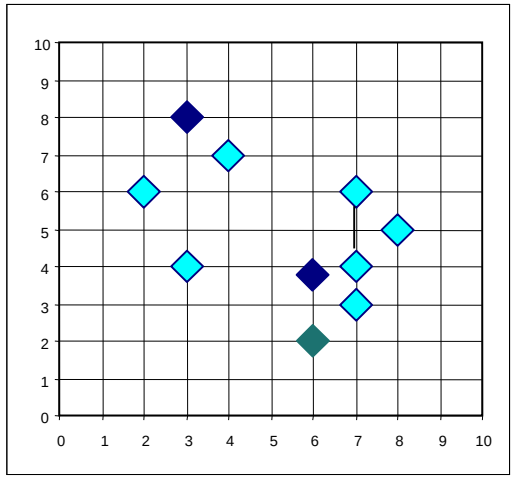
Arbitrary choose k object as initial medoids



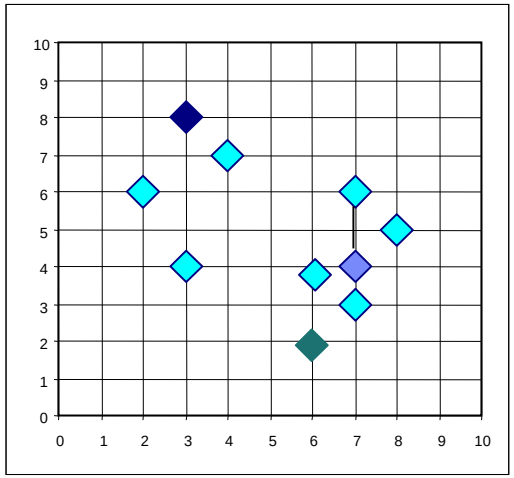
Assign each remaining object to nearest medoids



Randomly select a nonmedoid object,$O_{ramdom}$



Compute total cost of swapping



Total Cost = 26

Swapping O and $O_{ramdom}$

If quality is improved.

K=2

**Do loop**

**Until no change**

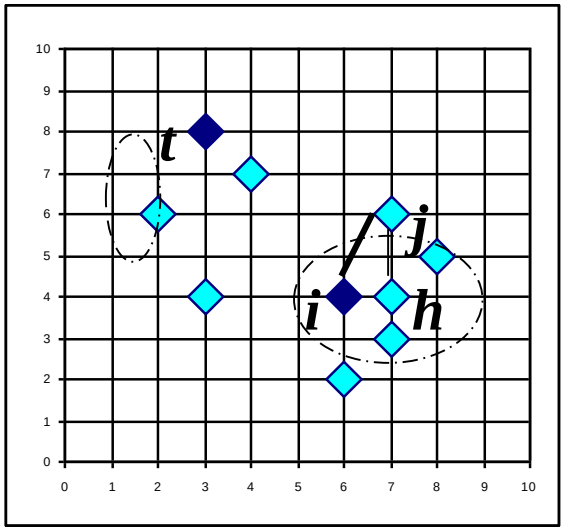
# PAM (Partitioning Around Medoids) (1987)



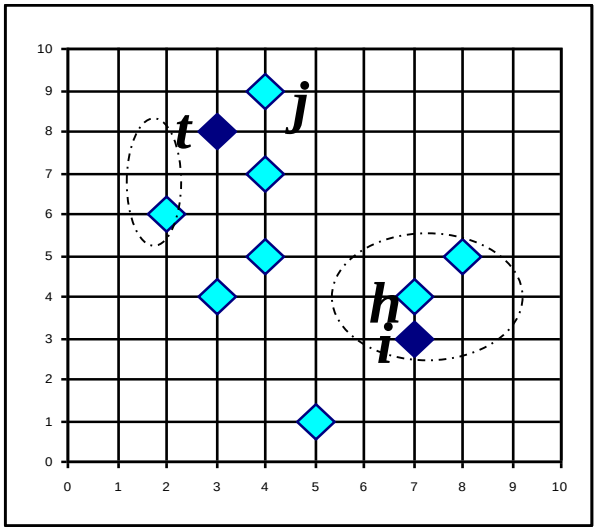
- PAM (Kaufman and Rousseeuw, 1987), built in Splus
- Use real object to represent the cluster
  - Select ***k*** representative objects arbitrarily
  - For each pair of non-selected object ***h*** and selected object ***i***, calculate the total swapping cost $TC_{ih}$
  - For each pair of ***i*** and ***h***,
    - If $TC_{ih} < 0$, ***i*** is replaced by ***h***
    - Then assign each non-selected object to the most similar representative object
  - repeat steps 2-3 until there is no change

# PAM Clustering: Total swapping cost $TC_{ih}=\sum_j C_{jih}$

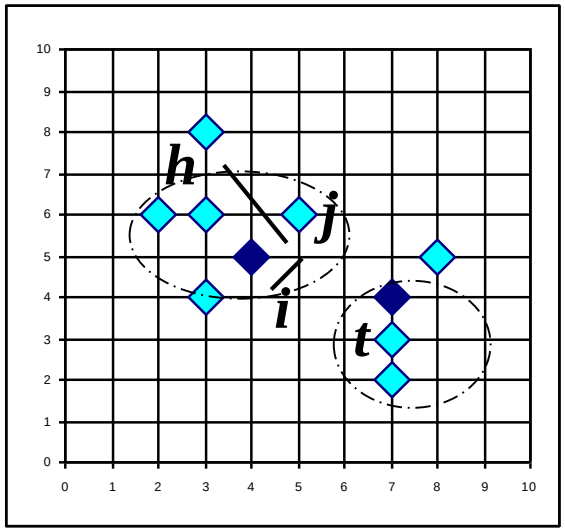




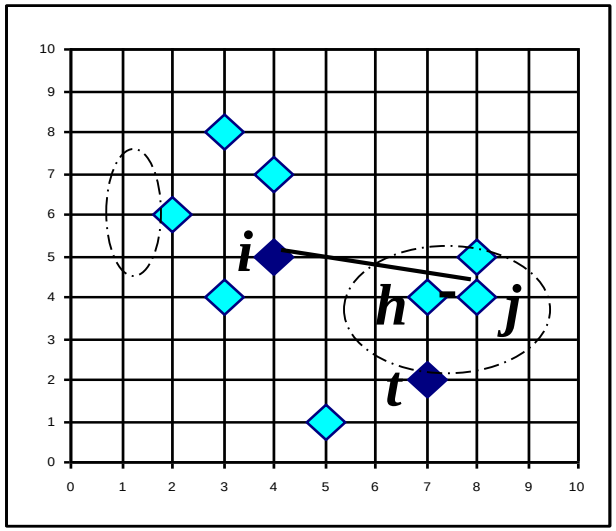

$C_{jih} = d(j, h) - d(j, i)$




$C_{jih} = 0$




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




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

# What is the problem with PAM?



- Pam is more robust than k-means in the presence of noise and outliers because a medoid is less influenced by outliers or other extreme values than a mean
- Pam works efficiently for small data sets but does not **scale well** for large data sets.
  - $O(k(n-k)^2)$ for each iteration

where n is # of data,k is # of clusters

➔Sampling based method,

CLARA(Clustering LARge Applications)

# *CLARA* (Clustering Large Applications) (1990)



- *CLARA* (Kaufmann and Rousseeuw in 1990)
  - Built in statistical analysis packages, such as S+
- 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)
- CLARANS draws sample of neighbors dynamically
- The clustering process can be presented as searching a graph where every node is a potential solution, that is, a set of *k* medoids
- If the local optimum is found, *CLARANS* starts with new randomly selected node in search for a new local optimum
- It is more efficient and scalable than both *PAM* and *CLARA*
- Focusing techniques and spatial access structures may further improve its performance (Ester et al.'95)

# Density-Based Clustering Methods



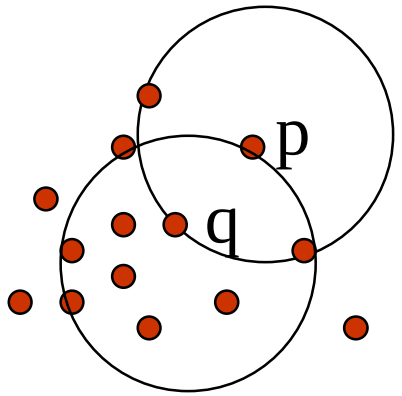
- Clustering based on density (local cluster criterion), such as density-connected points
- Major features:
  - Discover clusters of arbitrary shape
  - Handle noise
  - One scan
  - Need density parameters as termination condition
- Several interesting studies:
  - DBSCAN: Ester, et al. (KDD'96)
  - OPTICS: Ankerst, et al (SIGMOD'99).
  - DENCLUE: Hinneburg & D. Keim (KDD'98)
  - CLIQUE: Agrawal, et al. (SIGMOD'98)

# Density-Based Clustering: Background



- Two parameters***:***
  - ***Eps***: Maximum radius of the neighbourhood
  - ***MinPts***: Minimum number of points in an Eps-neighbourhood of that point
- $N_{Eps}(p)$: ***{q belongs to D | dist(p,q) <= Eps}***
- Directly density-reachable**:** A point ***p*** is directly density-reachable from a point ***q*** wrt. ***Eps***, ***MinPts*** if
  - 1) ***p*** belongs to $N_{Eps}(q)$
  - 2) core point condition:

$$|N_{Eps}(q)| >= MinPts$$



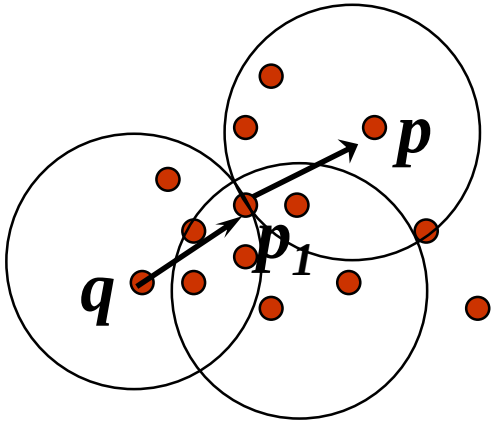

MinPts = 5

Eps = 1 cm

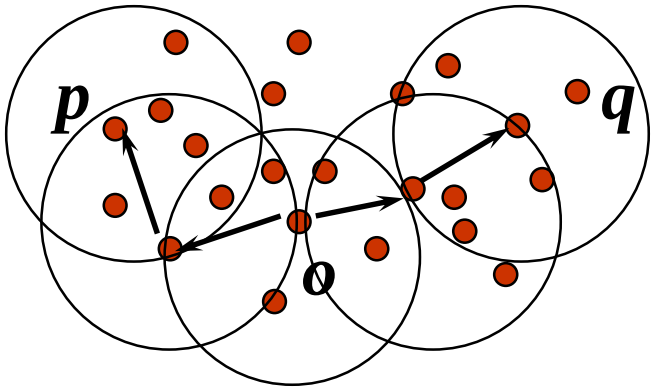
# Density-Based Clustering: Background (II)



- Density-reachable:
  - A point *p* is density-reachable from a point *q* wrt. *Eps*, *MinPts* if there is a chain of points $p_1$, …, $p_n$, $p_1 = q$, $p_n = p$ such that $p_{i+1}$ is directly density-reachable from $p_i$



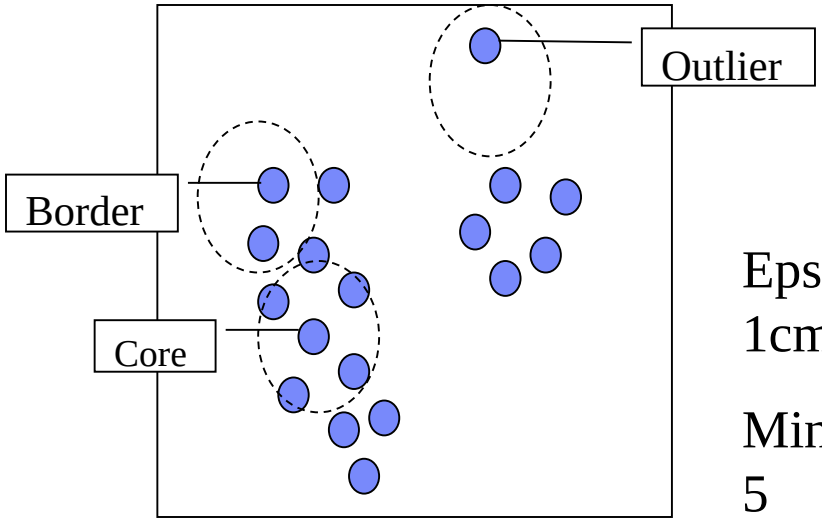

- Density-connected
  - A point *p* is density-connected to a point *q* wrt. *Eps*, *MinPts* if there is a point *o* such that both, *p* and *q* are density-reachable from *o* wrt. *Eps* and *MinPts*.



# DBSCAN: Density Based Spatial Clustering of Applications with Noise



- Relies on a *density-based* notion of cluster: A *cluster* is defined as a maximal set of density-connected points
- Discovers clusters of arbitrary shape in spatial databases with noise
- DBSCAN is a density-based algorithm.
  - Density = number of points within a specified radius (Eps)
  - A point is a core point if it has more than a specified number of points (MinPts) within Eps
    - These are points that are at the interior of a cluster
  - A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point
  - A noise point is any point that is not a core point or a border point.




Eps = 1cm

MinPts = 5

# DBSCAN Algorithm



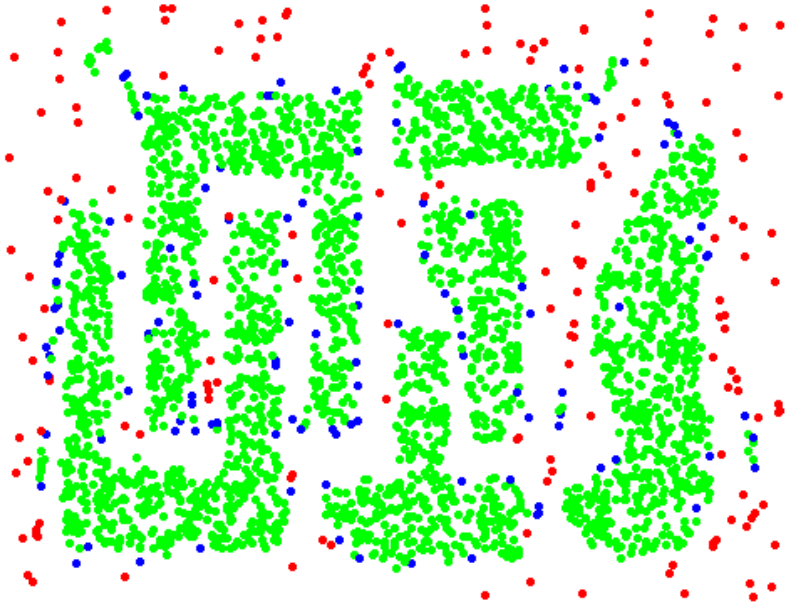
- Eliminate noise points
- Perform clustering on the remaining points

$current\_cluster\_label \leftarrow 1$
**for** all core points **do**
  **if** the core point has no cluster label **then**
    $current\_cluster\_label \leftarrow current\_cluster\_label + 1$
    Label the current core point with cluster label $current\_cluster\_label$
  **end if**
  **for** all points in the $Eps$-neighborhood, except $i^{th}$ the point itself **do**
    **if** the point does not have a cluster label **then**
      Label the point with cluster label $current\_cluster\_label$
    **end if**
  **end for**
**end for**

# DBSCAN: Core, Border and Noise Points



Original Points



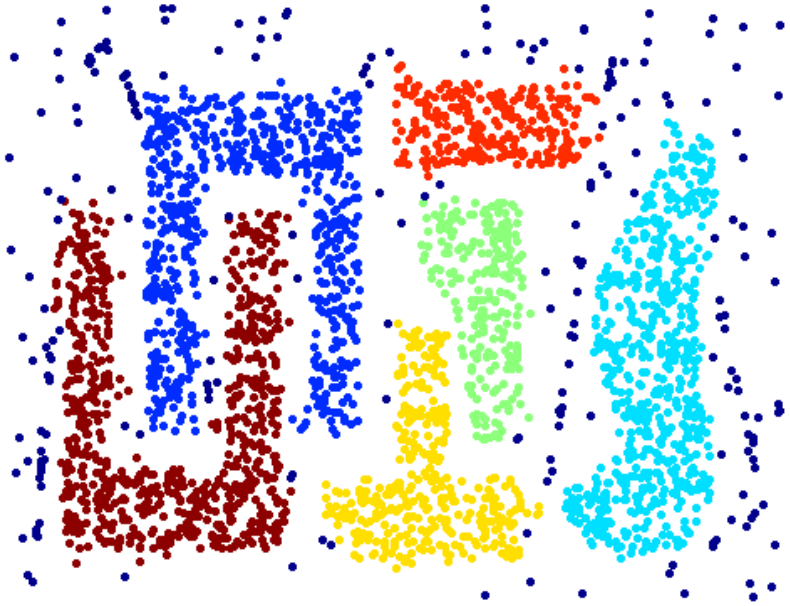
Point types: core, border and noise

Eps = 10, MinPts = 4

# When DBSCAN Works Well



Original Points

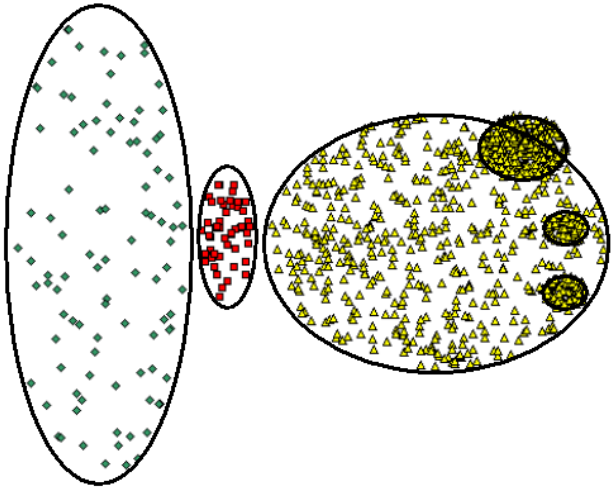


Clusters

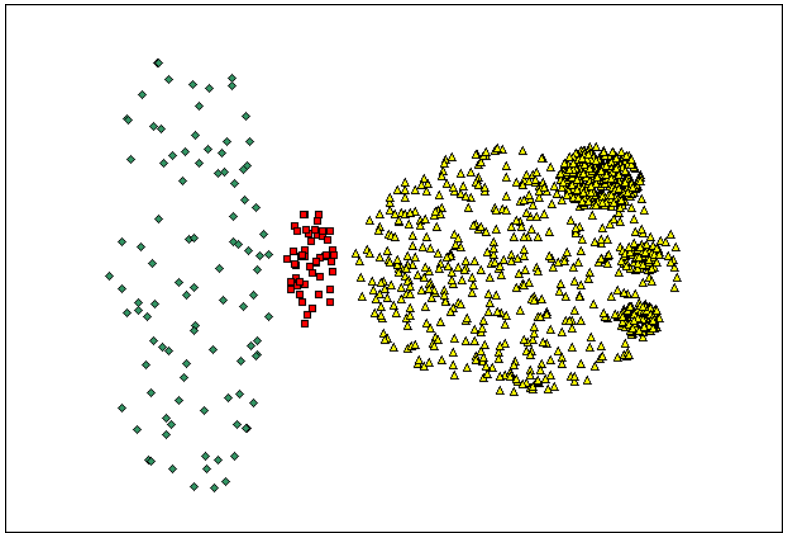
- Resistant to Noise
- Can handle clusters of different shapes and sizes

# When DBSCAN Does NOT Work Well



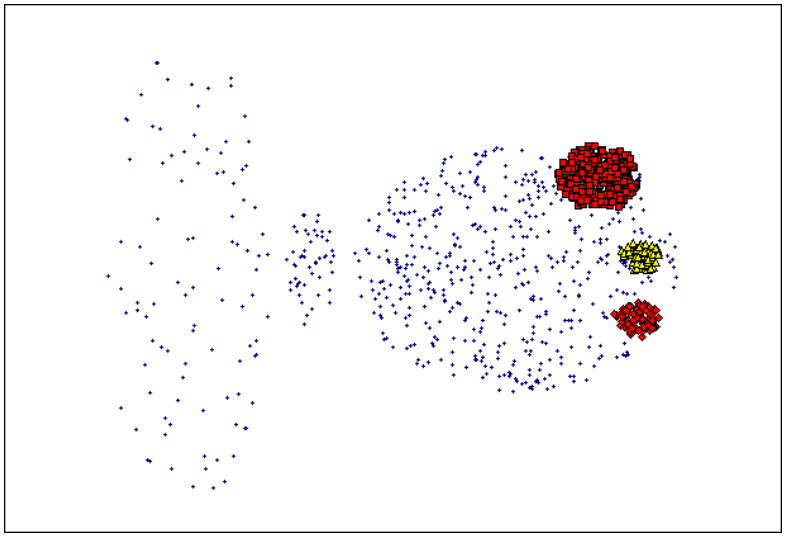


**Original Points**



(MinPts=4, Eps=9.75).

- **Varying densities**
- **High-dimensional data**

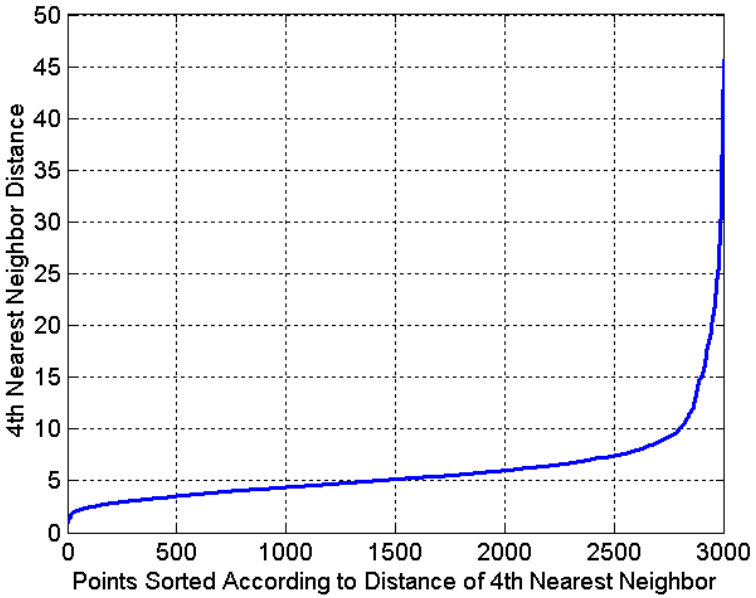


(MinPts=4, Eps=9.92)

# DBSCAN: Determining EPS and MinPts



- Idea is that for points in a cluster, their $k^{th}$ nearest neighbors are at roughly the same distance
- Noise points have the $k^{th}$ nearest neighbor at farther distance
- So, plot sorted distance of every point to its $k^{th}$ nearest neighbor



# External criteria for clustering quality



- Quality measured by its ability to discover some or all of the hidden patterns or latent classes in gold standard data
- Assesses a clustering with respect to ground truth requires *labeled data*
- Assume documents with *C* gold standard classes, while our clustering algorithms produce *K* clusters, $\omega_1$, $\omega_2$, …, $\omega_K$ with $n_i$ members.
- Simple measure: purity, the ratio between the dominant class in the cluster $\pi_i$ and the size of cluster $\omega_i$

$$Purity(\omega_i) = \frac{1}{n_i} \max_j (n_{ij}) \quad j \in C$$

- Biased because having *n* clusters maximizes purity
- Others are entropy of classes in clusters (or mutual information between classes and clusters)

# Rand index and Cluster F-measure



$$RI = \frac{A + B}{A + B + C + D}$$

Compare with standard Precision and Recall:

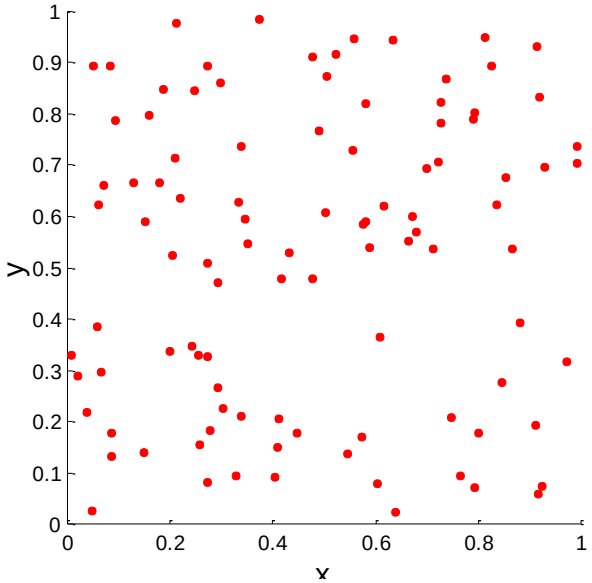
$$P = \frac{A}{A + B} \qquad R = \frac{A}{A + C}$$

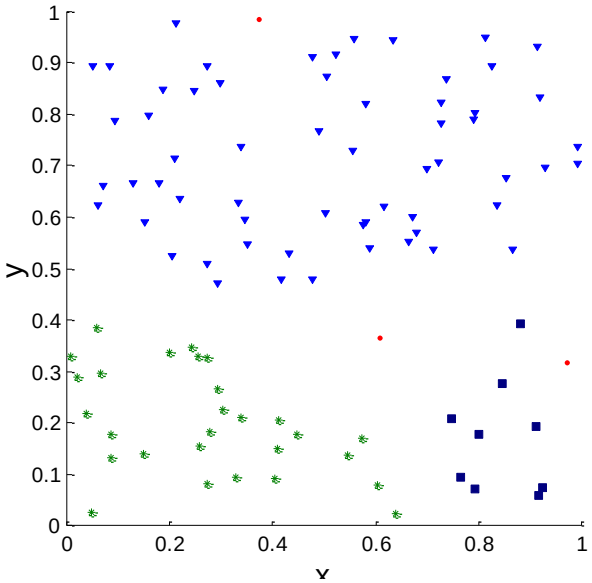
People also define and use a cluster F-measure, which is probably a better measure.

# Clusters found in Random Data



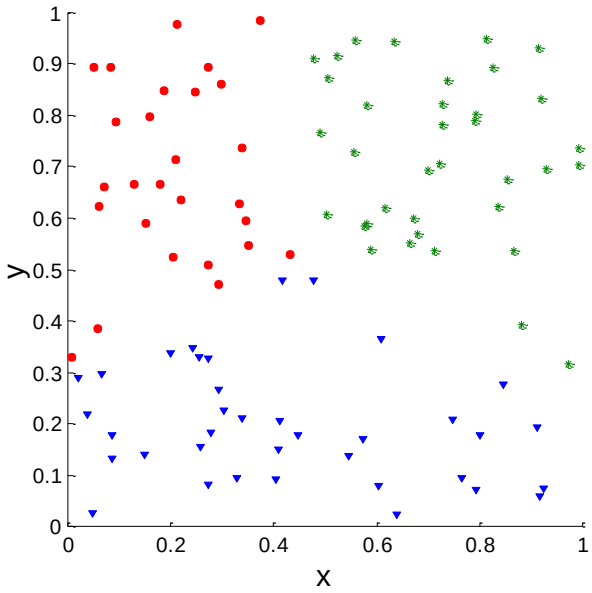
**Random Points**

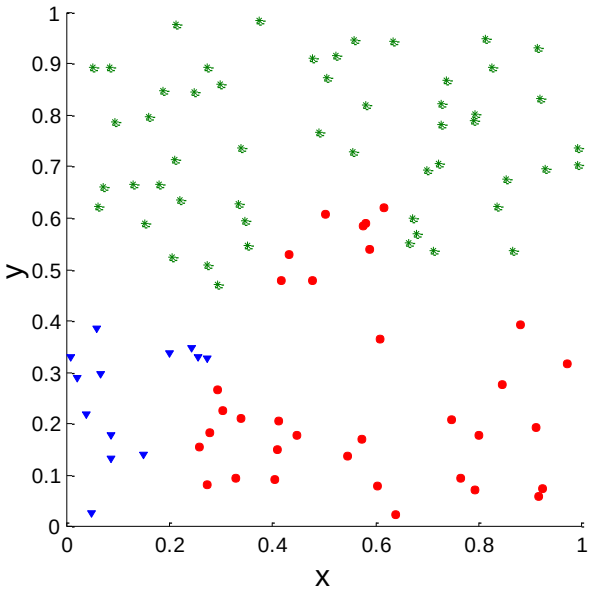



**DBSCAN**

**K-means**




**Complete Link**

# Different Aspects of Cluster Validation



- Determining the clustering tendency of a set of data, i.e., distinguishing whether non-random structure actually exists in the data.
- Comparing the results of a cluster analysis to externally known results, e.g., to externally given class labels.
- Evaluating how well the results of a cluster analysis fit the data *without* reference to external information.
  - Use only the data
- Comparing the results of two different sets of cluster analyses to determine which is better.
- Determining the ‘correct’ number of clusters.
- For 2, 3, and 4, we can further distinguish whether we want to evaluate the entire clustering or just individual clusters.

# Measures of Cluster Validity



- Numerical measures that are applied to judge various aspects of cluster validity, are classified into the following three types.
  - External Index: Used to measure the extent to which cluster labels match externally supplied class labels.
    - Entropy
  - Internal Index: Used to measure the goodness of a clustering structure *without* respect to external information.
    - Sum of Squared Error (SSE)
  - Relative Index: Used to compare two different clusterings or clusters.
    - Often an external or internal index is used for this function, e.g., SSE or entropy
- Sometimes these are referred to as criteria instead of indices
  - However, sometimes criterion is the general strategy and index is the numerical measure that implements the criterion.

# Measuring Cluster Validity Via Correlation

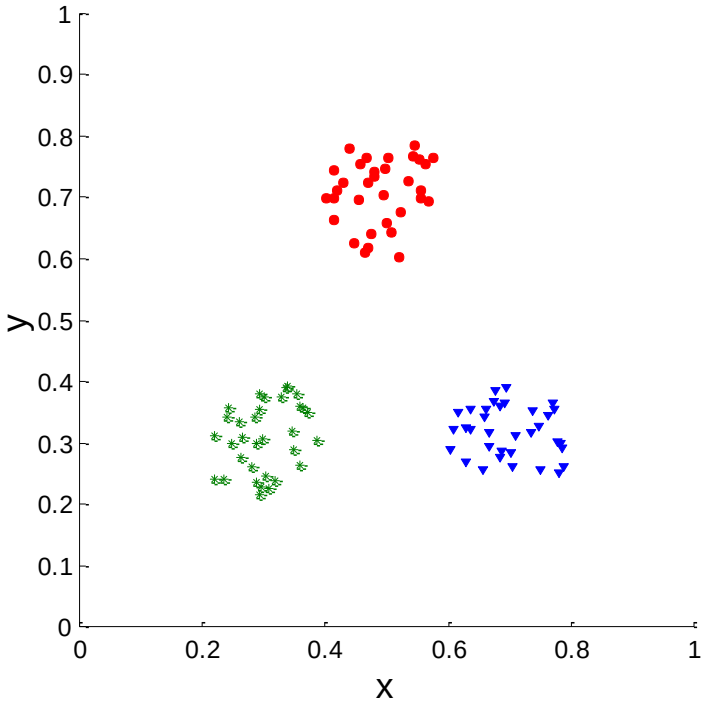


- Two matrices
  - Proximity Matrix
  - "Incidence" Matrix
    - One row and one column for each data point
    - An entry is 1 if the associated pair of points belong to the same cluster
    - An entry is 0 if the associated pair of points belongs to different clusters
- Compute the correlation between the two matrices
  - Since the matrices are symmetric, only the correlation between n(n-1) / 2 entries needs to be calculated.
- High correlation indicates that points that belong to the same cluster are close to each other.
- Not a good measure for some density or contiguity based clusters.

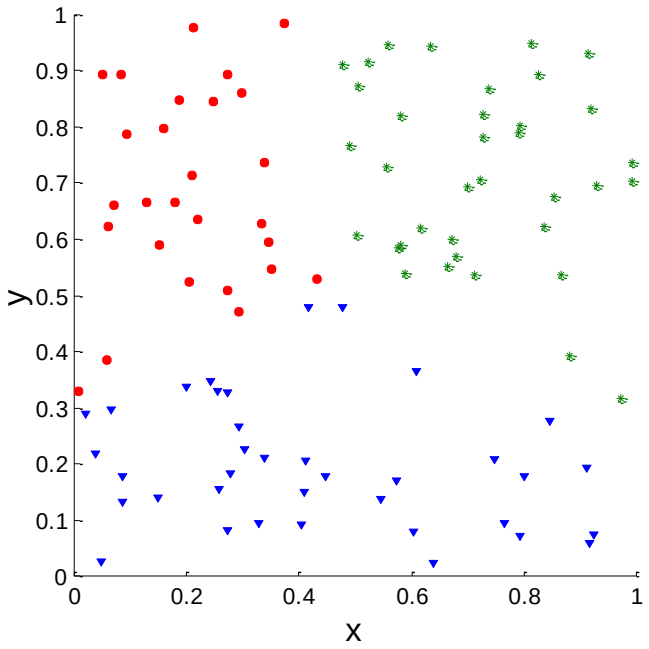
# Measuring Cluster Validity Via Correlation



- Correlation of incidence and proximity matrices for the K-means clustering's of the following two data sets.




**Corr = -0.9235**

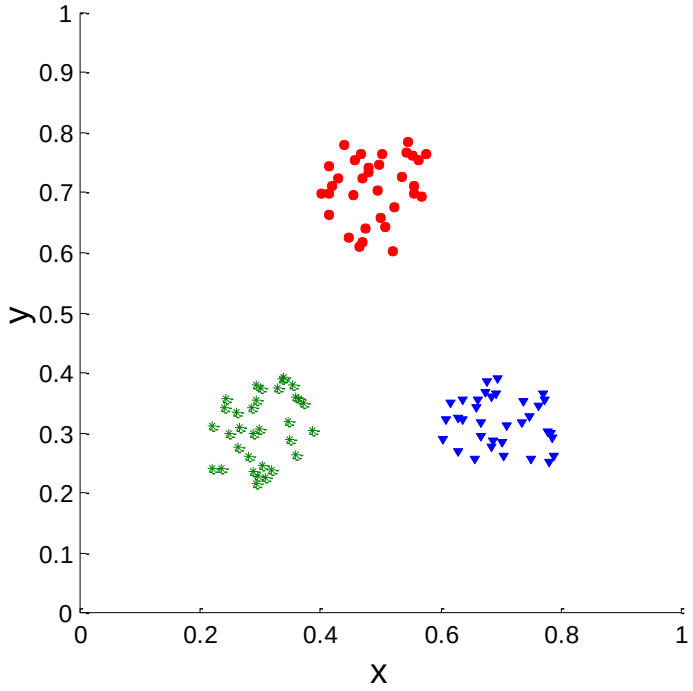


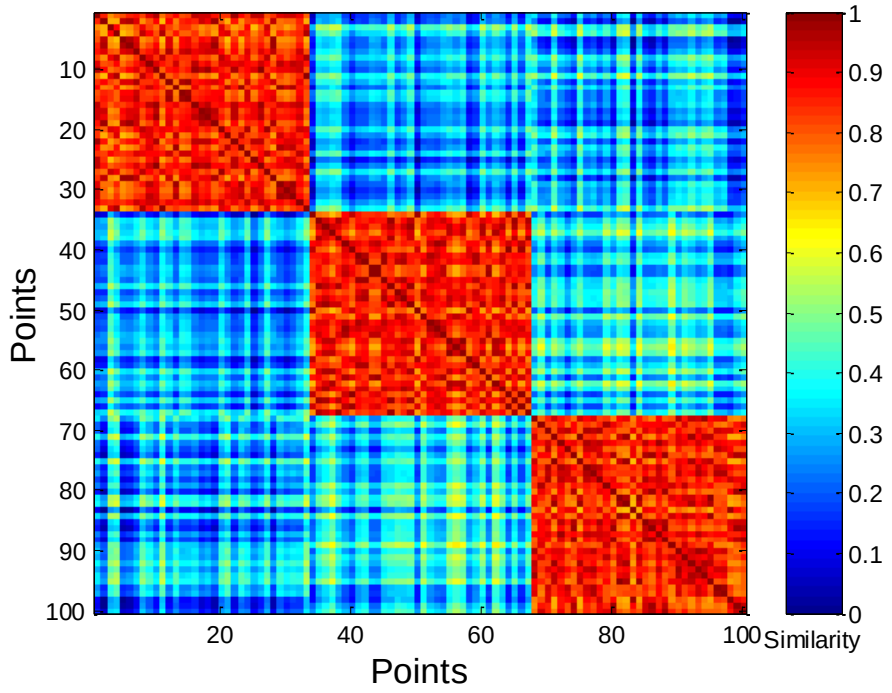

**Corr = -0.5810**

# Using Similarity Matrix for Cluster Validation



- Order the similarity matrix with respect to cluster labels and inspect visually.

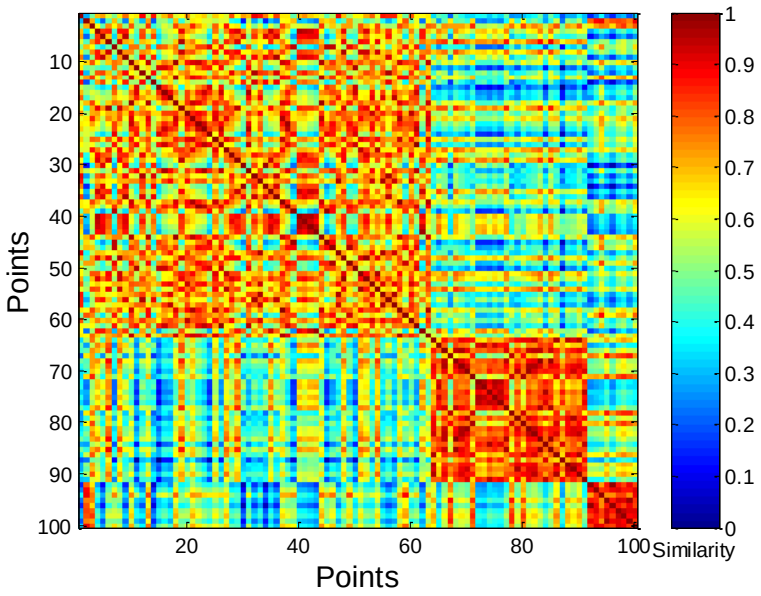


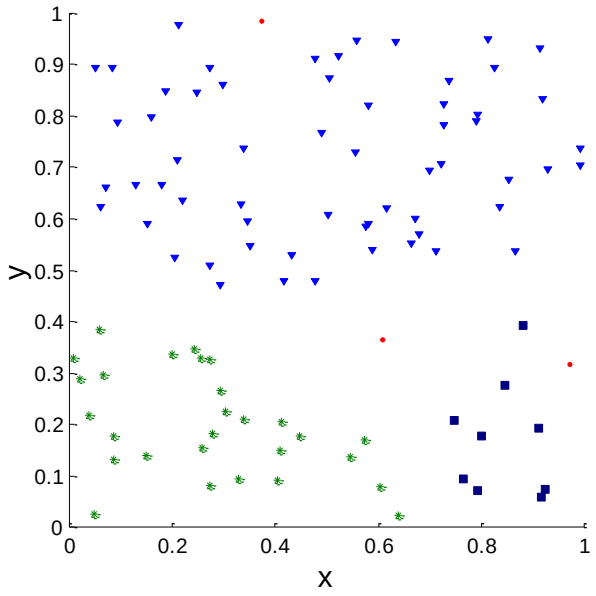



# Using Similarity Matrix for Cluster Validation



- Clusters in random data are not so crisp



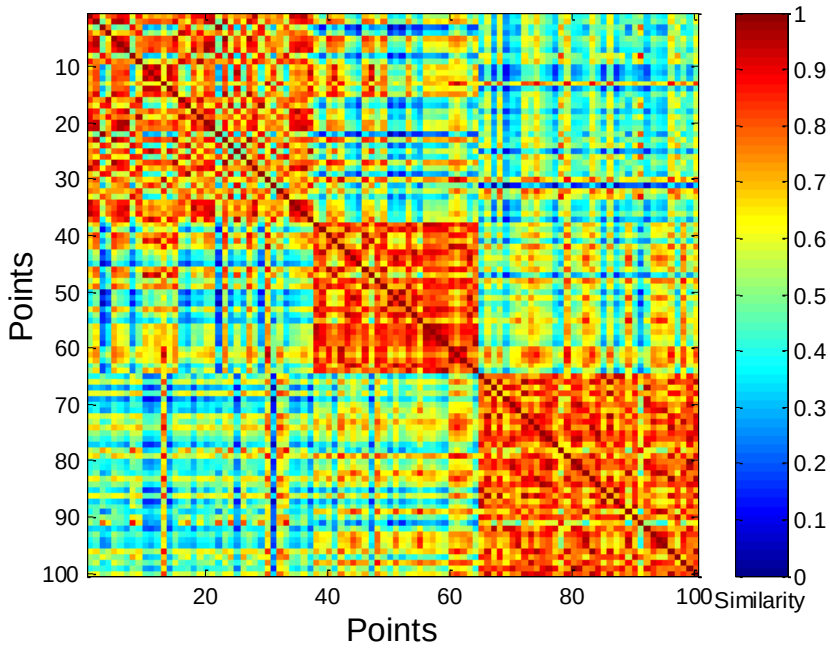



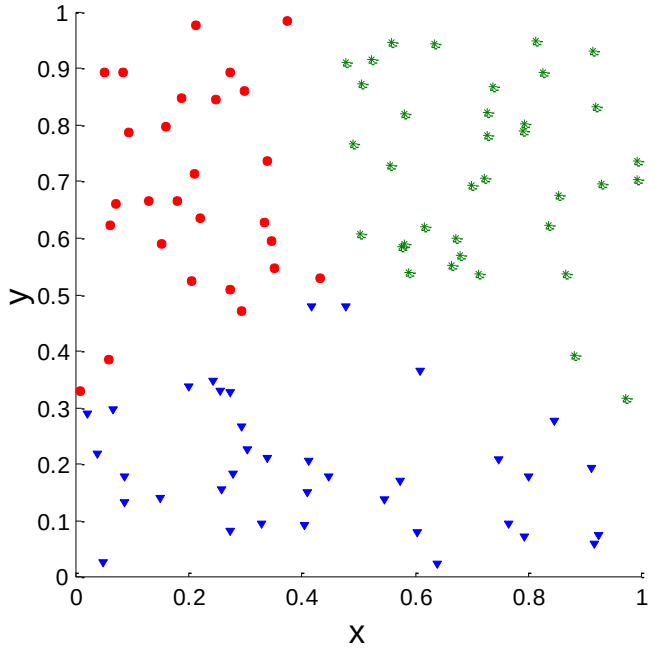

**DBSCAN**

# Using Similarity Matrix for Cluster Validation



- Clusters in random data are not so crisp

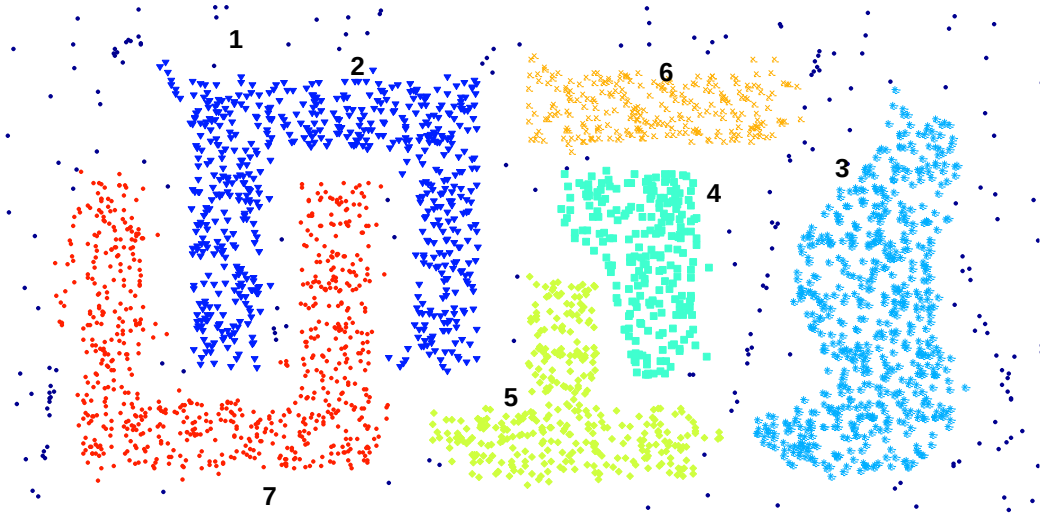


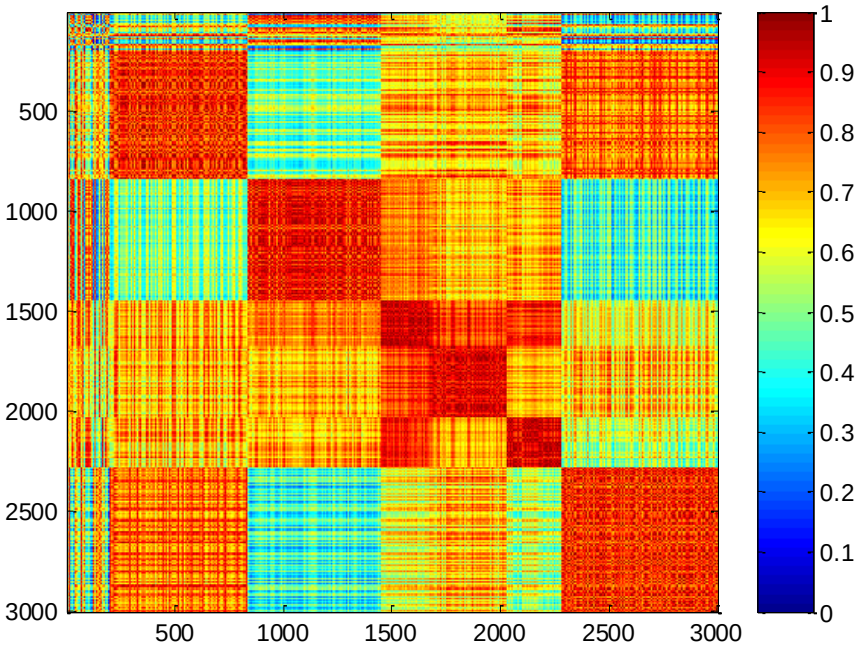




**K-means**

# Using Similarity Matrix for Cluster Validation





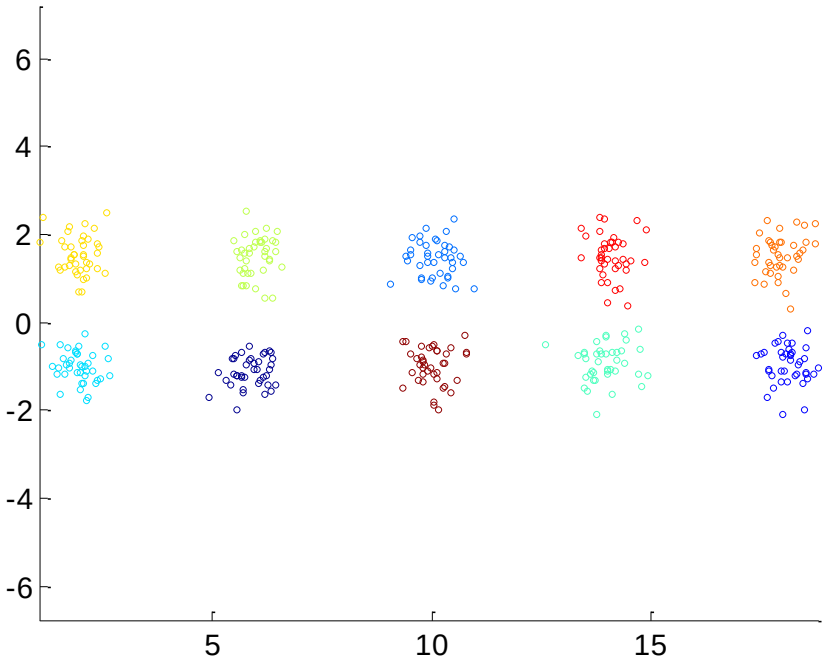



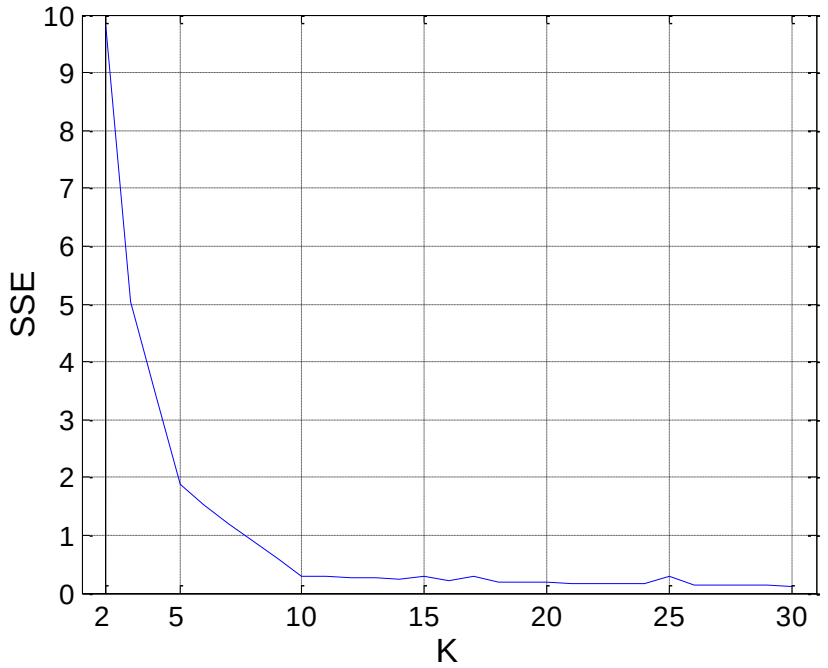

**DBSCAN**

# Internal Measures: SSE



- Clusters in more complicated figures aren't well separated
- Internal Index: Used to measure the goodness of a clustering structure without respect to external information
  - SSE
- SSE is good for comparing two clusterings or two clusters (average SSE).
- Can also be used to estimate the number of clusters

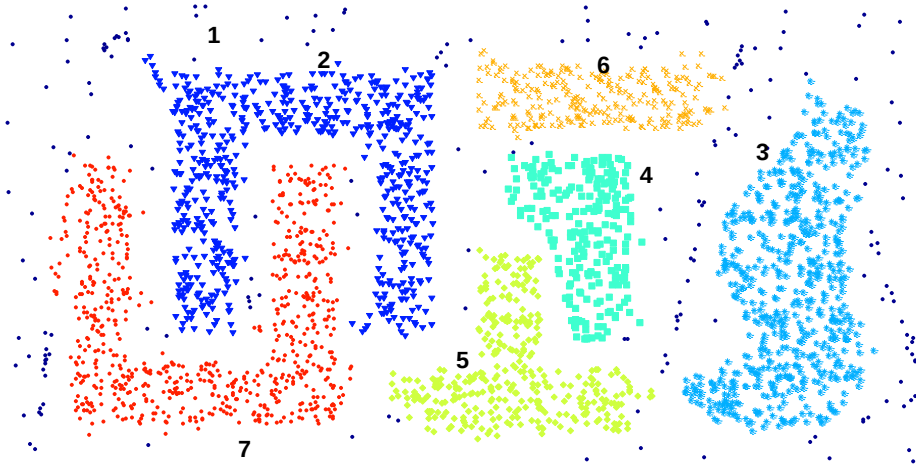


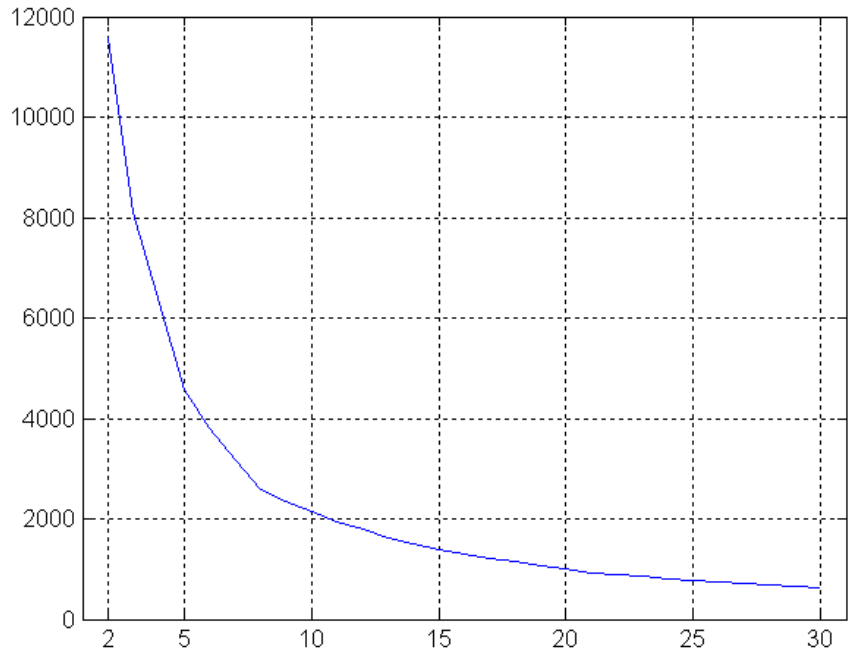



# Internal Measures: SSE



- SSE curve for a more complicated data set




**SSE of clusters found using K-means**

# Framework for Cluster Validity

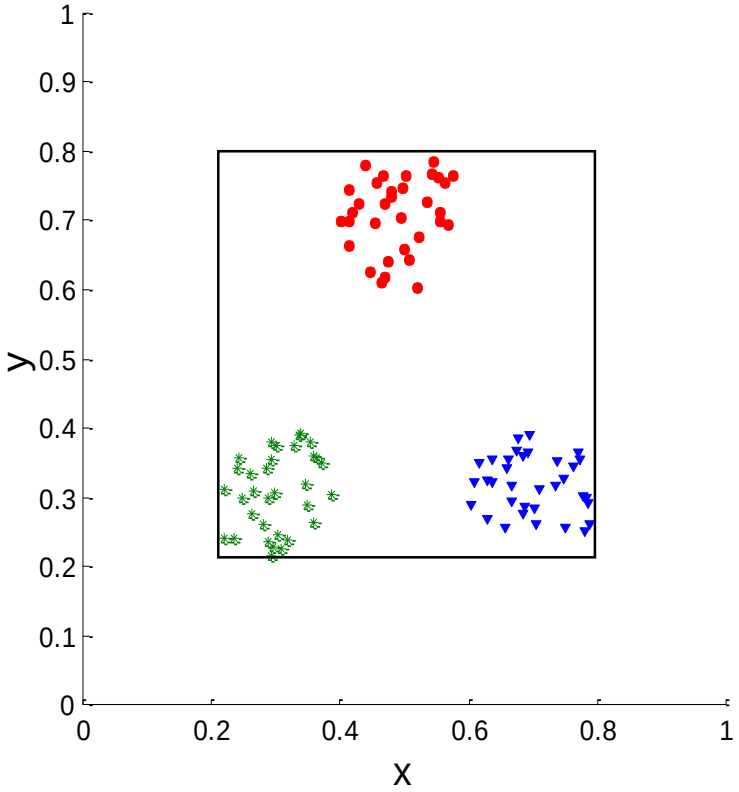


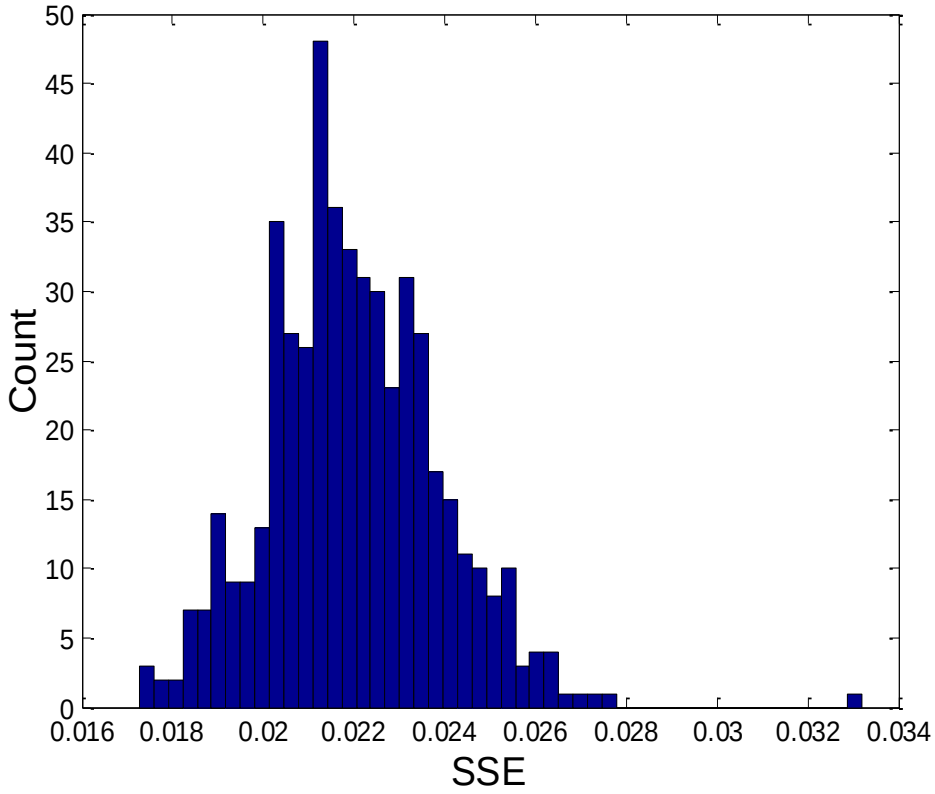
- Need a framework to interpret any measure.
  - For example, if our measure of evaluation has the value, 10, is that good, fair, or poor?
- Statistics provide a framework for cluster validity
  - The more “atypical” a clustering result is, the more likely it represents valid structure in the data
  - Can compare the values of an index that result from random data or clusterings to those of a clustering result.
    - If the value of the index is unlikely, then the cluster results are valid
  - These approaches are more complicated and harder to understand.
- For comparing the results of two different sets of cluster analyses, a framework is less necessary.
  - However, there is the question of whether the difference between two index values is significant

# Statistical Framework for SSE



- Example
  - Compare SSE of 0.005 against three clusters in random data
  - Histogram shows SSE of three clusters in 500 sets of random data points of size 100 distributed over the range 0.2 – 0.8 for x and y values

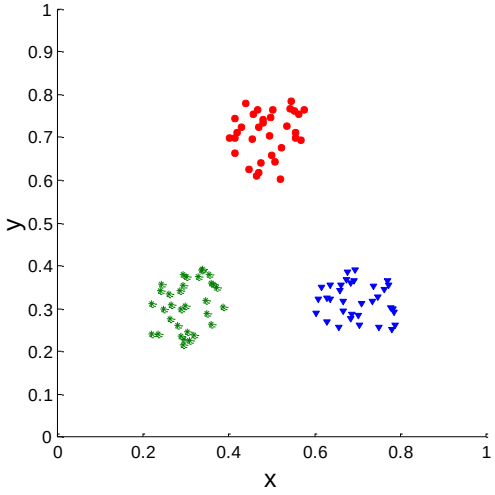


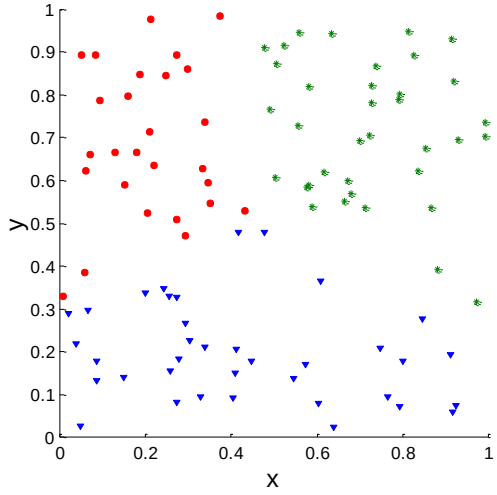



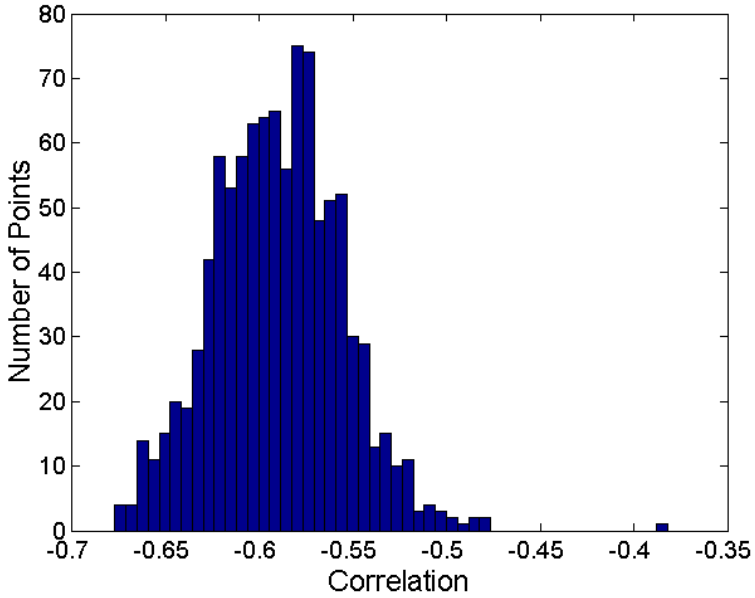
# Statistical Framework for Correlation



- Correlation of incidence and proximity matrices for the K-means clusterings of the following two data sets.




**Corr = -0.9235**

**Corr = -0.5810**

# Internal Measures: Cohesion and Separation



- Cluster Cohesion: Measures how closely related are objects in a cluster
  - Example: SSE
- Cluster Separation: Measure how distinct or well-separated a cluster is from other clusters
- Example: Squared Error
  - Cohesion is measured by the within cluster sum of squares (SSE)

$$WSS = \sum_{i} \sum_{x \in C_i} (x - m_i)^2$$

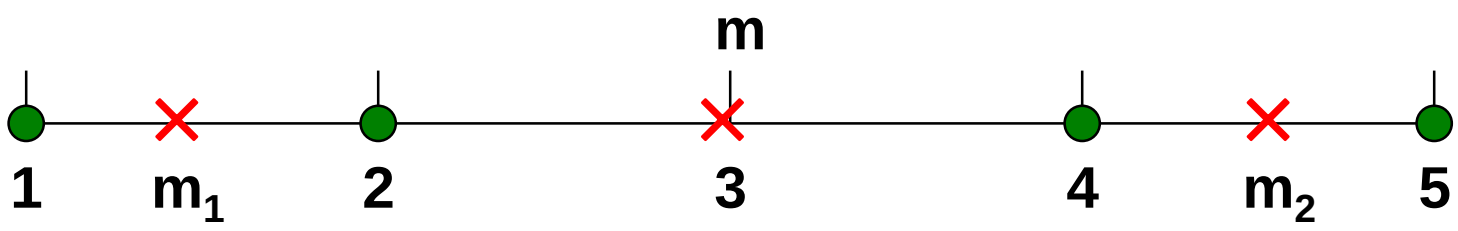
  - Separation is measured by the between cluster sum of squares

$$BSS = \sum_{i} |C_i| (m - m_i)^2$$

    - Where $|C_i|$ is the size of cluster i

# Internal Measures: Cohesion and Separation

- Example: SSE
  - BSS + WSS = constant




**K=1 cluster:**

$$WSS = (1-1.5)^2 + (2-1.5)^2 + (4-4.5)^2 + (5-4.5)^2 = 1$$

$$BSS = 2 \times (3-1.5)^2 + 2 \times (4.5-3)^2 = 9$$

$$Total = 1 + 9 = 10$$

**K=2 clusters:**

$$WSS = (1-3)^2 + (2-3)^2 + (4-3)^2 + (5-3)^2 = 10$$

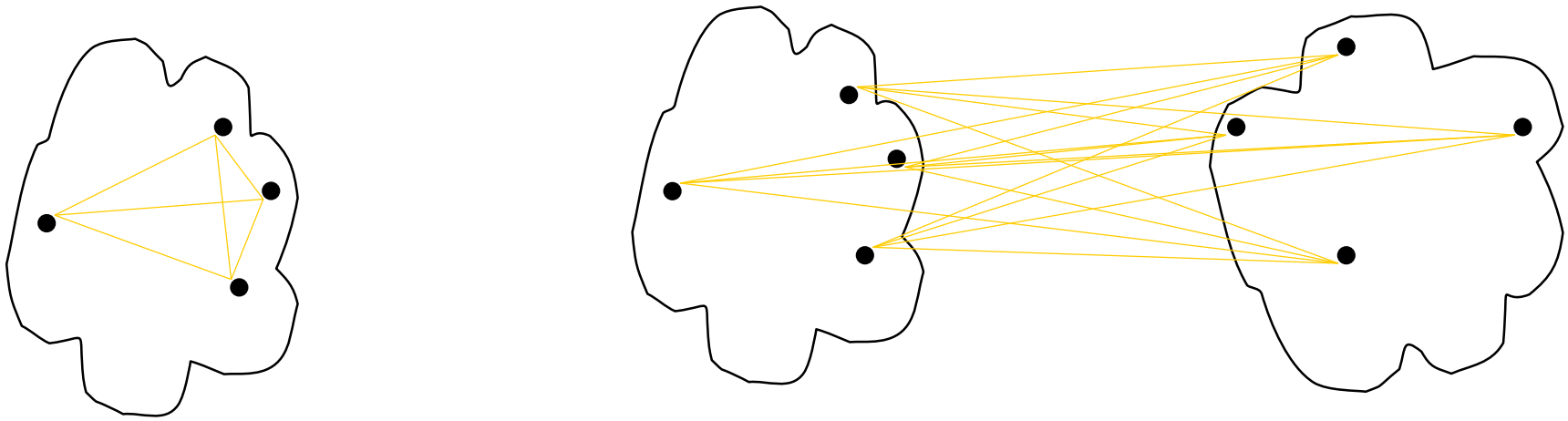
$$BSS = 4 \times (3-3)^2 = 0$$

$$Total = 10 + 0 = 10$$

# Internal Measures: Cohesion and Separation



- A proximity graph based approach can also be used for cohesion and separation.
  - Cluster cohesion is the sum of the weight of all links within a cluster.
  - Cluster separation is the sum of the weights between nodes in the cluster and nodes outside the cluster.



cohesion

separation

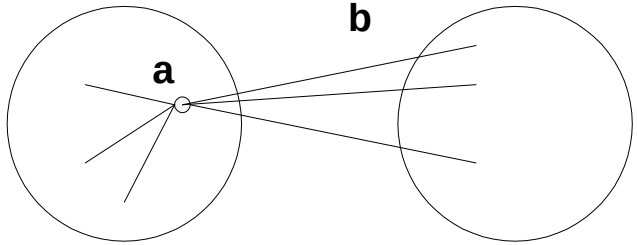
# Internal Measures: Silhouette Coefficient



- Silhouette Coefficient combine ideas of both cohesion and separation, but for individual points, as well as clusters and clusterings
- For an individual point, *i*
  - Calculate ***a*** = average distance of *i* to the points in its cluster
  - Calculate ***b*** = min (average distance of *i* to points in another cluster)
  - The silhouette coefficient for a point is then given by

    $s = 1 - a/b$ if $a < b$, (or $s = b/a - 1$ if $a \geq b$, not the usual case)

  - Typically between 0 and 1.
  - The closer to 1 the better.




- Can calculate the Average Silhouette width for a cluster or a clustering

# Final Comment on Cluster Validity



"The validation of clustering structures is the most difficult and frustrating part of cluster analysis.

Without a strong effort in this direction, cluster analysis will remain a black art accessible only to those true believers who have experience and great courage."

*Algorithms for Clustering Data*, Jain and Dubes

# Summary



- Clustering analysis groups objects based on their (dis)similarity and has a broad range of applications.
- Distance (or similarity) metric underlying data types plays a critical role in all clustering analysis and distance-based learning (e.g., k-NN)
- Clustering algorithms can be categorized into partitioning, hierarchical, density-based, model-based, spectral clustering and clustering ensemble
- There are still lots of research issues on cluster analysis;
  - finding the number of “natural” clusters with arbitrary “shapes”
  - dealing with mixed types of features (correlated features but different types)
  - handling massive amount of data – Big Data (efficiency vs. performance)
  - coping with data of high dimensionality (many features for an object)
  - performance evaluation (especially when no ground-truth available)

# Checkpoints (1 of 2)



**Fill in the blanks:**

1. __is the minimum no. of variables/ features required to perform clustering.
2. _____ method is used for finding optimal of cluster in K-means algorithm.
3. _____ is a technique for analyzing data when the criterion or dependent variable is categorical and the independent variables are interval in nature.

**State True or False:**

1. A dendrogram is not possible for K-Means clustering analysis.
2. Unsupervised learning is a learning from unlabeled data using factor and cluster analysis models.
3. Automated vehicle is an example of supervised learning.

# Checkpoint Solutions (1 of 2)



**Fill in the blanks:**

1. **1** is the minimum no. of variables/ features required to perform clustering.
2. **Elbow** method is used for finding optimal of cluster in K-Mean algorithm.
3. **Cluster analysis** is a technique for analyzing data when the criterion or dependent variable is categorical and the independent variables are interval in nature.

**State True or False:**

1. A dendrogram is not possible for K-Means clustering analysis **– True.**
2. Unsupervised learning is a learning from unlabeled data using factor and cluster analysis models **– True.**
3. Automated vehicle is an example of un-supervised learning **– False.**

# Checkpoints (2 of 2)



## Multiple Choice Questions

1. Which of the following algorithm is most sensitive to outliers?
   a. K-means clustering algorithm
   b. K-medians clustering algorithm
   c. K-modes clustering algorithm
   d. K-medoids clustering algorithm

2. _____ is a clustering procedure characterized by the development of a tree-like structure.
   a. Non-hierarchical clustering
   b. Hierarchical clustering
   c. Divisive clustering
   d. Agglomerative clustering

3. The _____ method uses information on all pairs of distances, not merely the minimum or maximum distances.
   a. single linkage
   b. medium linkage
   c. complete linkage
   d. average linkage

# Checkpoint Solutions (2 of 2)



## Multiple Choice Questions

1. Which of the following algorithm is most sensitive to outliers?
   - **a. K-means clustering algorithm**
   - b. K-medians clustering algorithm
   - c. K-modes clustering algorithm
   - d. K-medoids clustering algorithm

2. _____ is a clustering procedure characterized by the development of a tree-like structure.
   - a. Non-hierarchical clustering
   - **b. Hierarchical clustering**
   - c. Divisive clustering
   - d. Agglomerative clustering

3. The _____ method uses information on all pairs of distances, not merely the minimum or maximum distances.
   - a. single linkage
   - b. medium linkage
   - c. complete linkage
   - **d. average linkage**

# Two Marks Questions



1. Define unsupervised learning.
2. Define clustering. Give an example.
3. Compare hard clustering to soft clustering.
4. Define cluster ensemble process.
5. Define Manhattan distance measure.
6. Define cosine similarity measure. Illustrate.
7. Define Dendrogram. Illustrate.
8. Define purity and rand index. Illustrate.
9. Define precision, recall and f-measure.

# Four Marks Questions



1. What are the goals of clustering?
2. Justify ‘Clustering is a pre-processing tool’.
3. Write a note on the requirements of clustering.
4. Discuss the type of data in clustering analysis.
5. Write about similarity and dissimilarity measures.
6. Explain the principle of partitioning methodology.
7. Explain the principle of hierarchical methodology.
8. Explain the principle of density based methodology
9. Explain the principle of spectral clustering methodology
10. Discuss agglomerative clustering algorithm.
11. Explain Ward’s clustering procedure.
12. Describe k-medoid clustering method.
13. Write about CLARA method.
14. Write a note on framework on cluster validation process.

# Eight Marks Questions



1. Discuss the process of unsupervised learning.
2. Write about the statistics associated with cluster analysis.
3. Describe the procedure of conducting cluster analysis.
4. Explain the classification of clustering procedures.
5. Discuss the applications of clustering.
6. Discuss the factors to be considered for cluster analysis.
7. Discuss the different type of clusters with an example to each.
8. Write about cluster distance measures.
9. Explain k-means clustering technique. Analyze its computational complexity.
10. Discuss DBSCAN algorithm. Comment on its time complexity.
11. Discuss about external criteria to measure cluster quality.