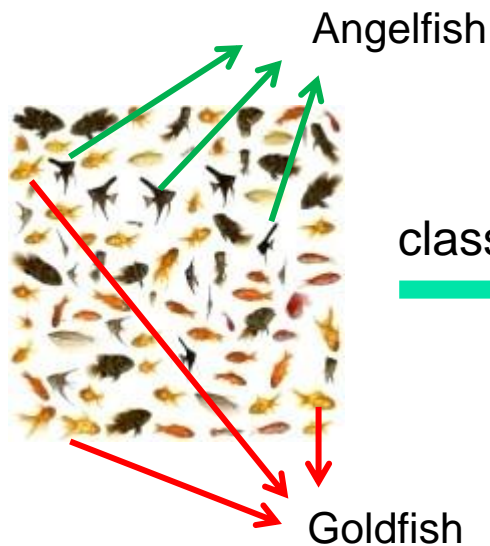
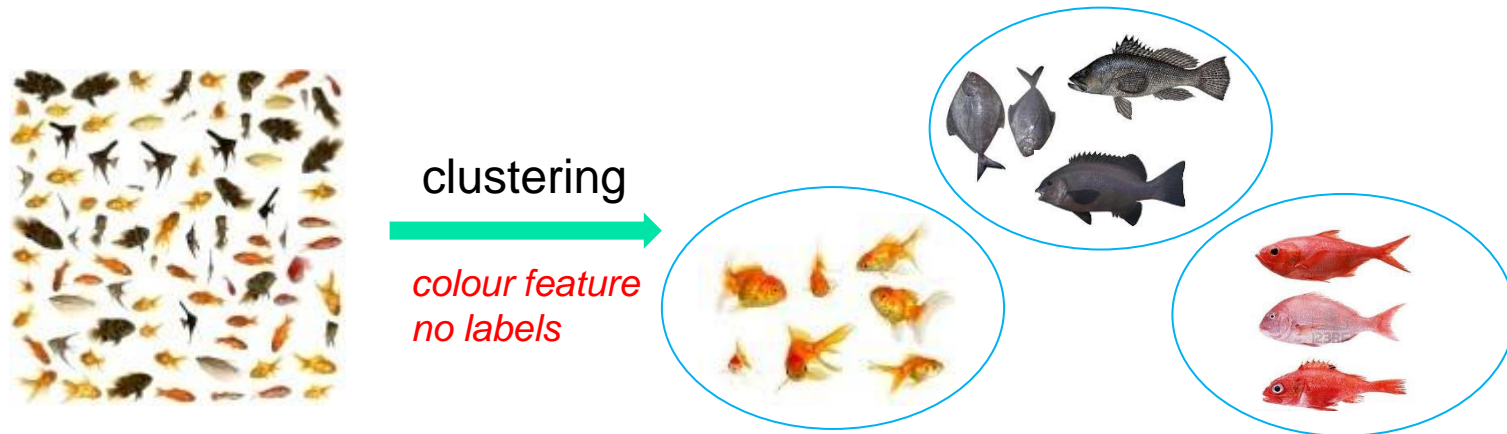




Data Mining

- Clustering (I)

Clustering vs. Classification



classification

Angelfish:

Up to 6 inches or 15cm. Their bodies are very thin, yet tall, their profile rounded, almost disc-shaped.

Salmon:

pinkish in color and have spots on their fins and back

Tuna:

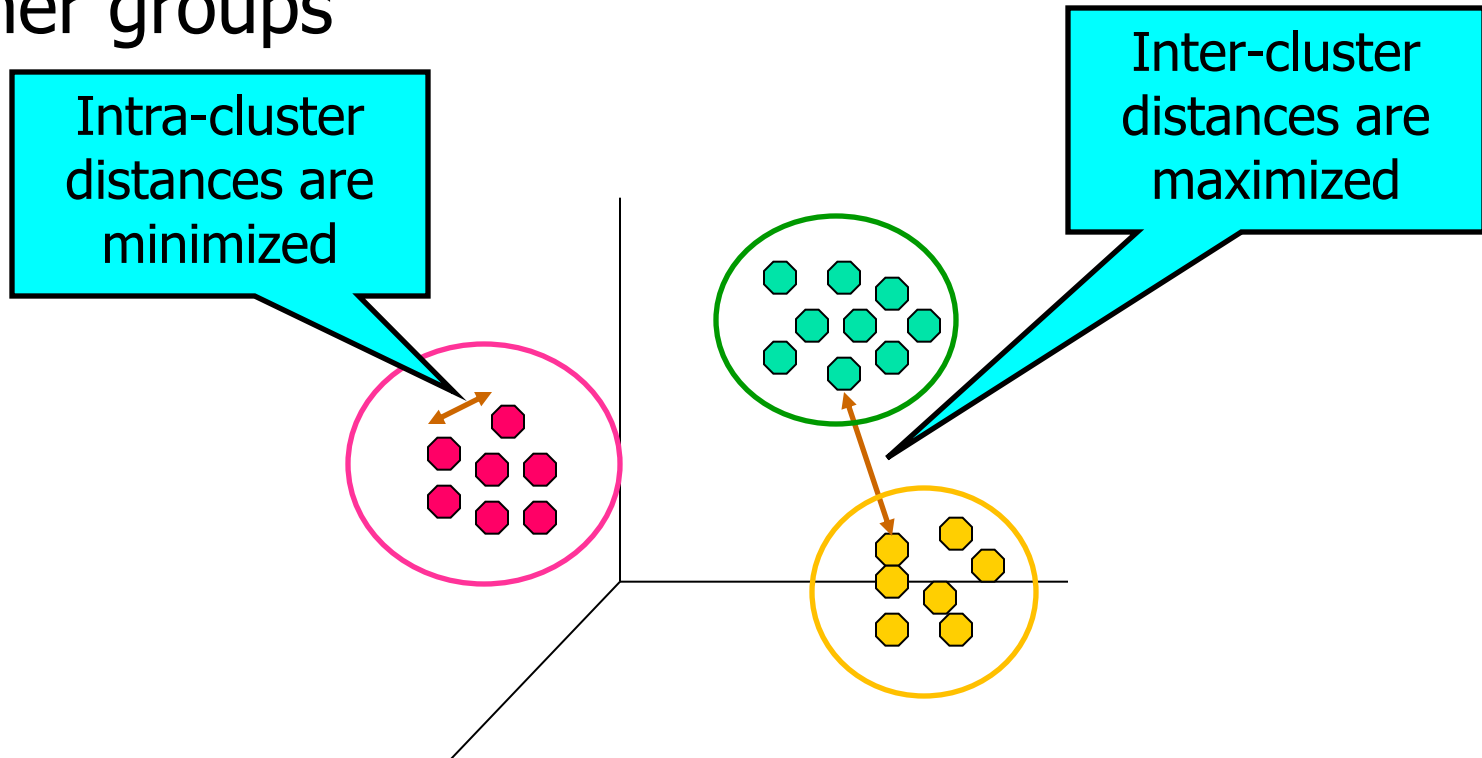
The tuna is a streamlined fish, stout in the middle

Outline

- Introduction
 - Definition
 - Application of clustering
- Clustering Models
 - Partitional Clustering
 - Hierarchical Clustering

What is Clustering?

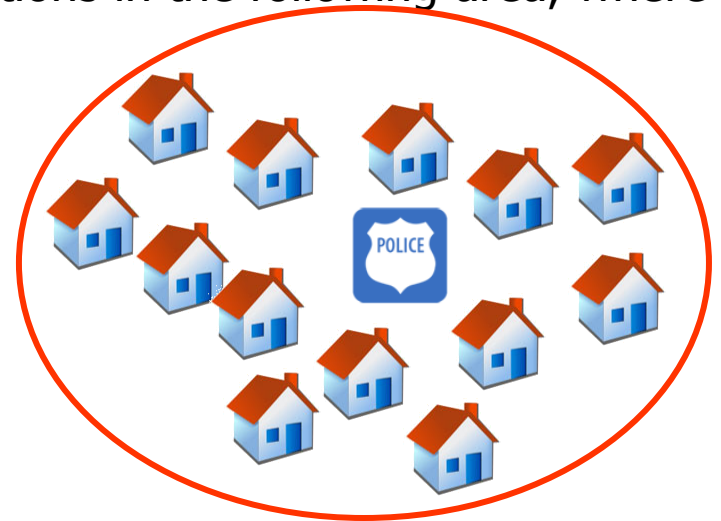
- Finding **groups** of objects such that the objects in a group will be **similar** (or related) to one another and **different** from (or unrelated to) the objects in other groups



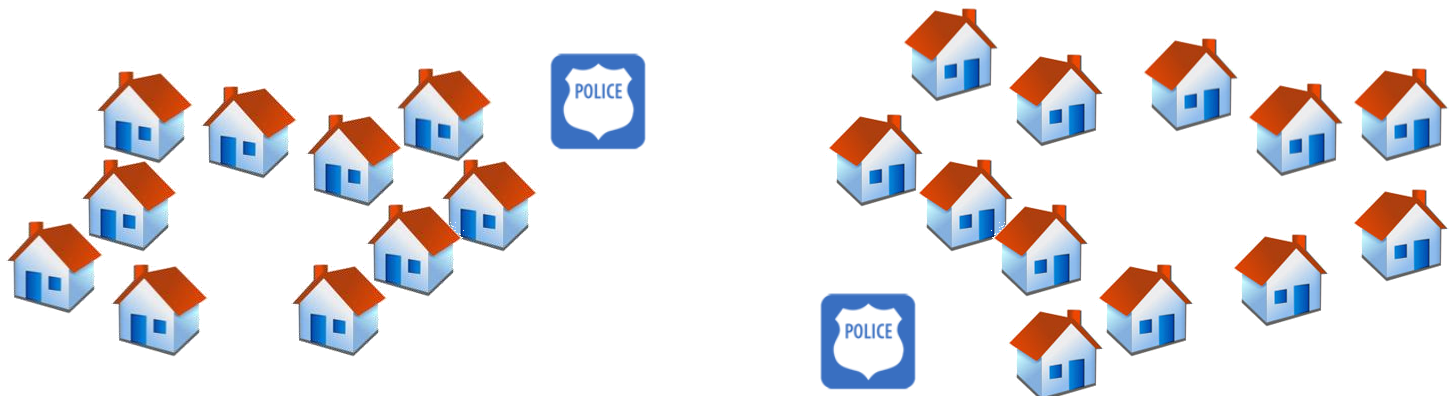
Some Applications of Clustering

- City planning
 - Suppose you want to build two police stations in the following area, where should you build?

Plan A



Plan B

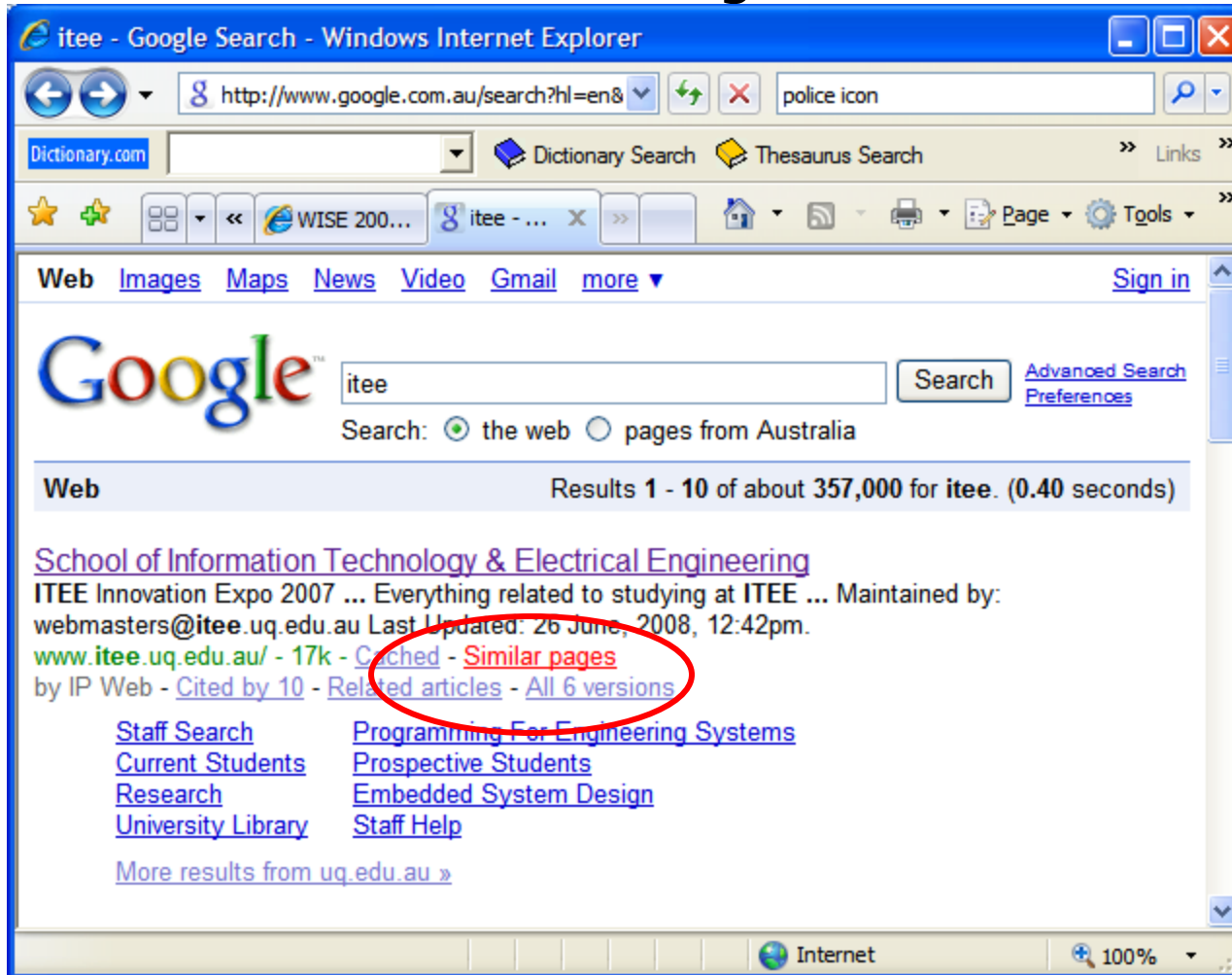


Some Applications of Clustering

- Market segmentation
 - Suppose you are working as a branch manager in an insurance company.
 - You have 10 sales teams.
 - You want to segment the market into 10 different segments, such that each team can concentrate on a specific market.
 - Criteria for segmentation may include:
 - Sex
 - Age
 - Income
 - Expenses
 - Career
 - ...

Some Applications of Clustering

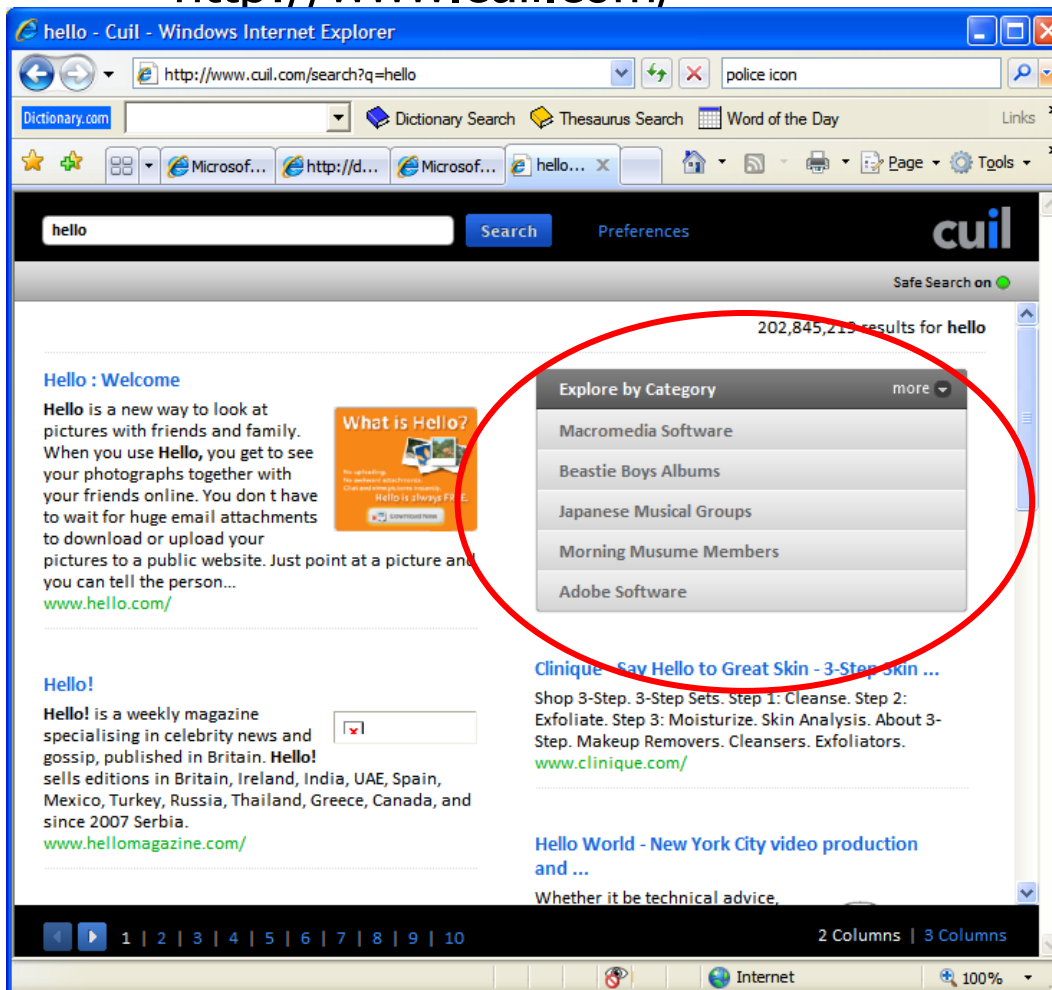
- Reduce information overloading



Some Applications of Clustering

- Easy navigation

- <http://www.cuil.com/>



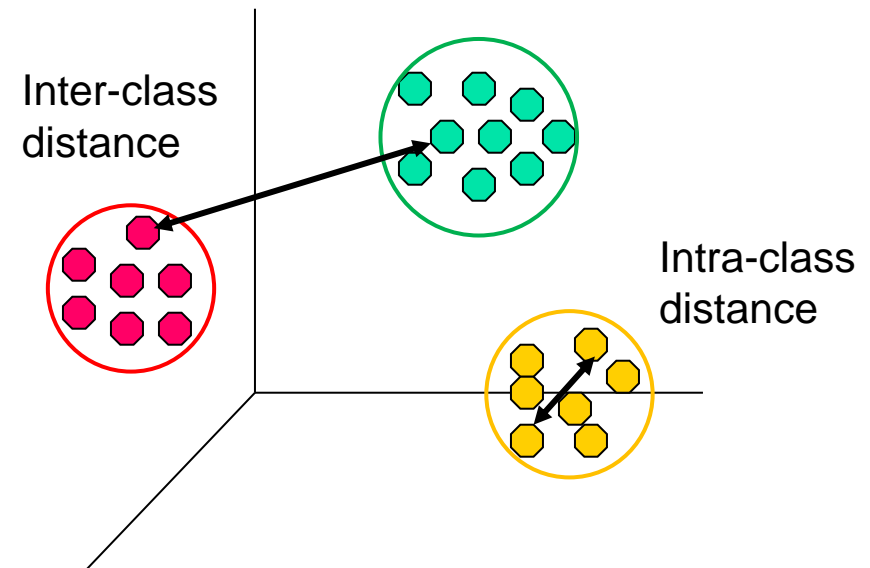
Cuil was a search engine that organized web pages by content. It launched in July 2008 with an index of 121,617,892,992 web pages.

It shut down on Sep 17, 2010.

- slow response time
- irrelevant or wrong search results

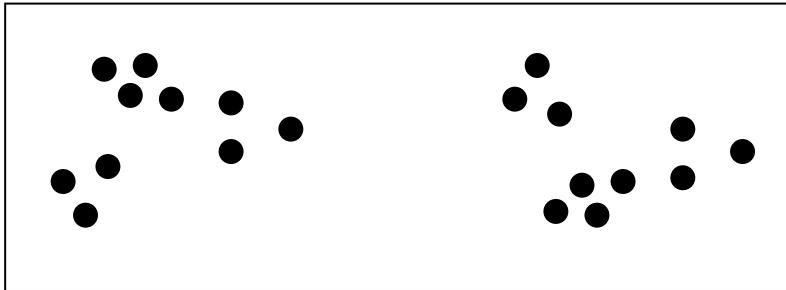
What is a Good Clustering?

- A good clustering method should produce high quality clusters with
 - High intra-class similarity
 - Low inter-class similarity
 - Ability to discover hidden patterns

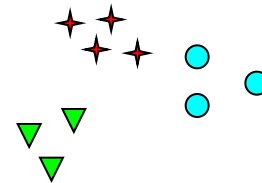


How Many Clusters?

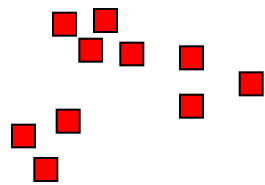
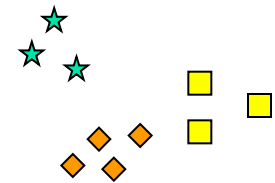
- How many clusters?



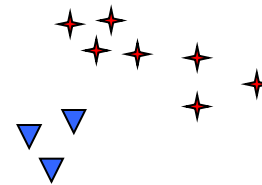
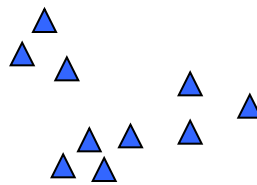
How many clusters?



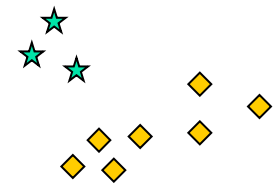
Six Clusters



Two Clusters



Four Clusters



Notion of a Cluster can be Ambiguous

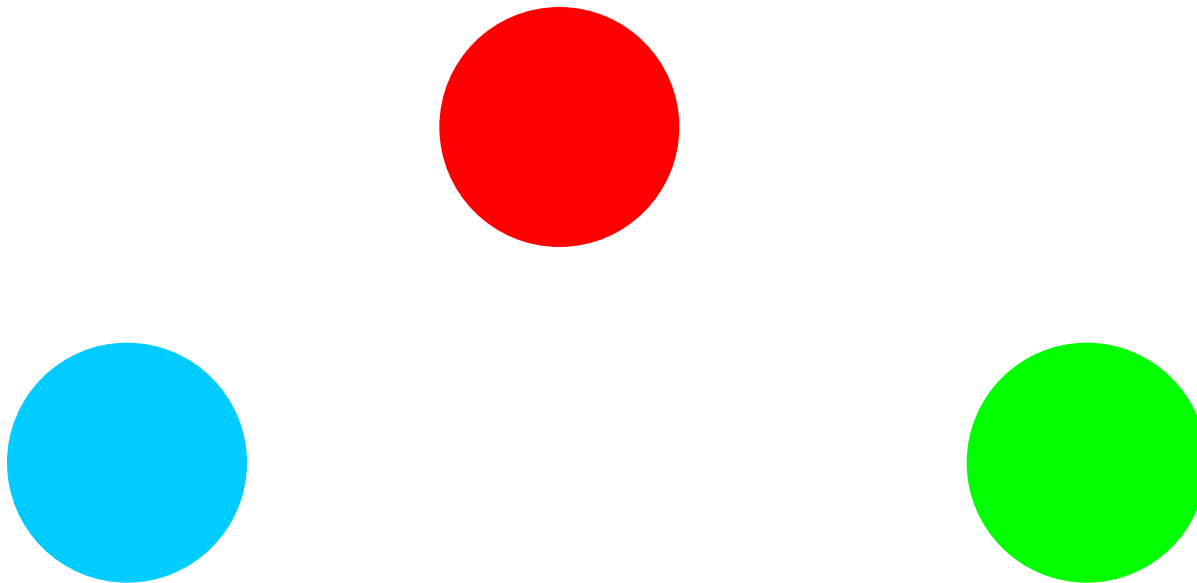
Types of Clusters

- Well-separated clusters
- Center-based clusters
- Contiguous clusters
- Density-based clusters
- Property or Conceptual
- 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.

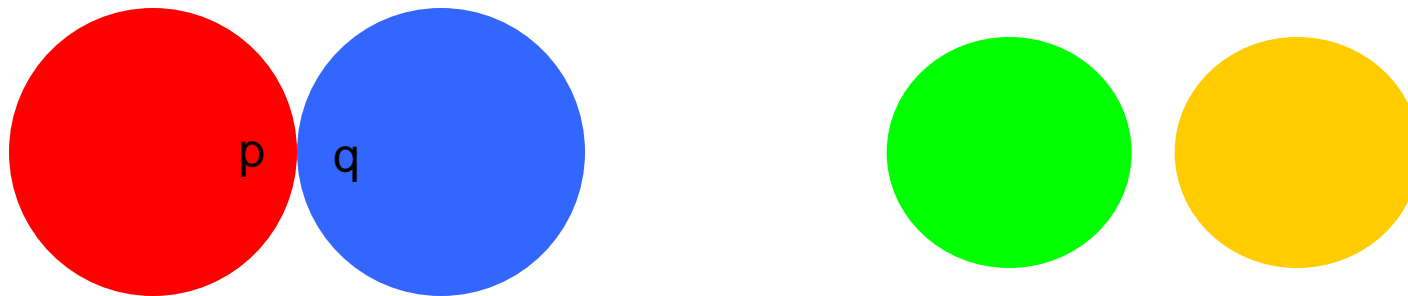


3 well-separated clusters

Types of Clusters: Center-Based

■ Center-based

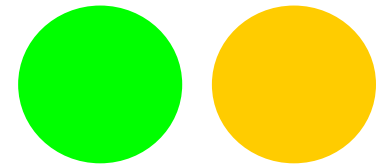
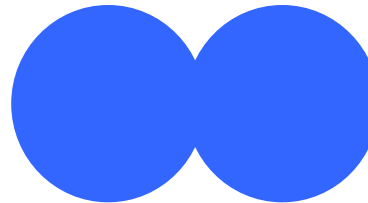
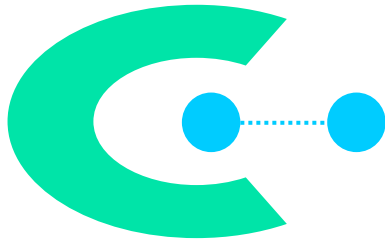
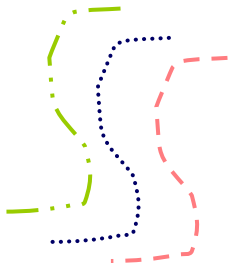
- A cluster is a set of objects such that an object in a cluster is closer (more similar) to the “center” of a cluster, than to the center of any other cluster
- The center of a cluster is often a **centroid**, the average of all the points in the cluster, or a **medoid**, the most “representative” point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

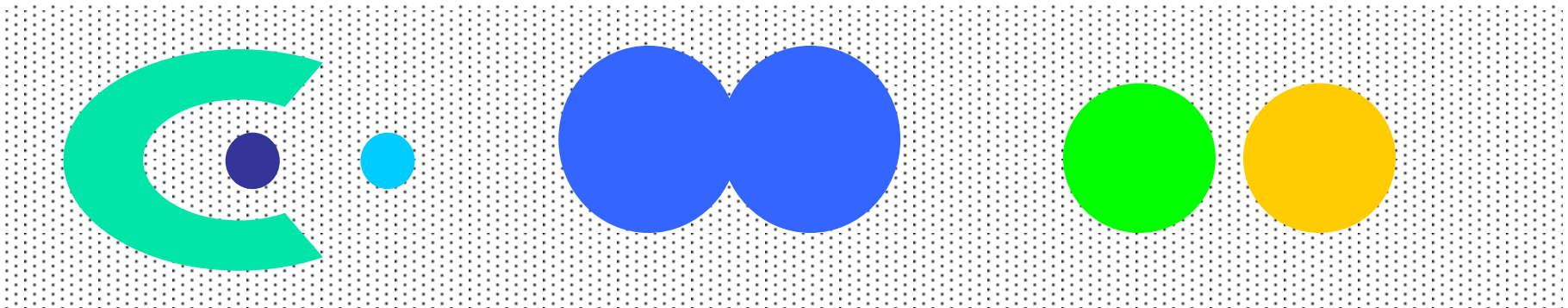


8 contiguous clusters

Types of Clusters: Density-Based

■ Density-based

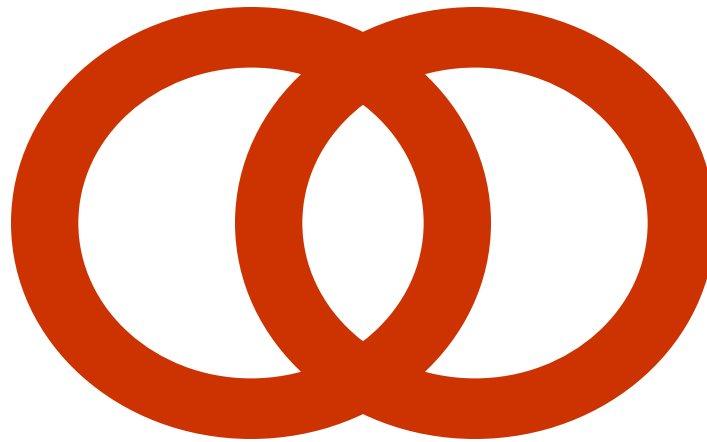
- A cluster is a dense region of points, which is separated by low-density regions, from other regions of high density.
- Used when the clusters are irregular or intertwined, and when noise and outliers are present.



6 density-based clusters

Types of Clusters: Conceptual Clusters

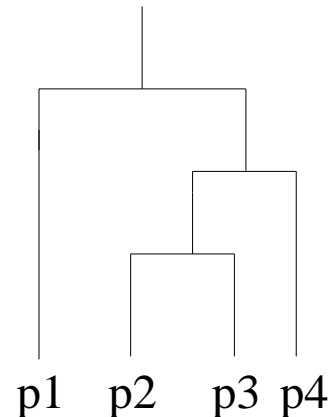
- Shared Property or Conceptual Clusters
 - Finds clusters that share some common property or represent a particular concept.



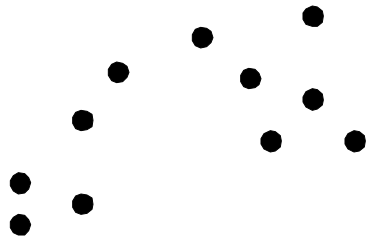
2 Overlapping Circles

Types of Clustering

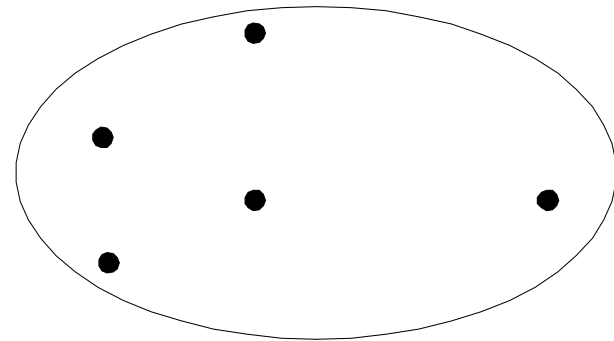
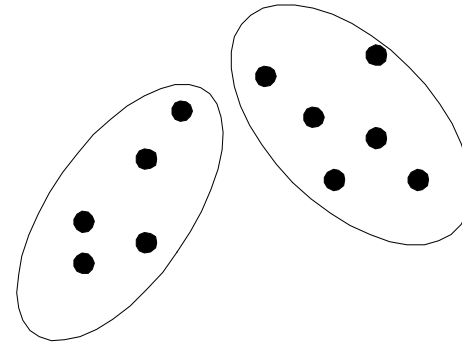
- A **clustering** is a set of clusters
- Broadly speaking, two types of clustering:
 - Partitional Clustering
 - A division data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset
 - Hierarchical clustering
 - A set of nested clusters organized as a hierarchical tree
 - Called dendrogram



Partitional Clustering

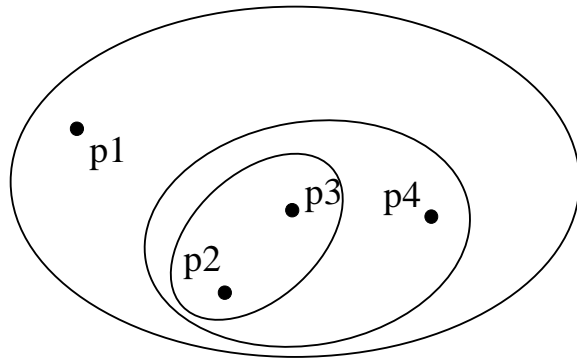


Original Points

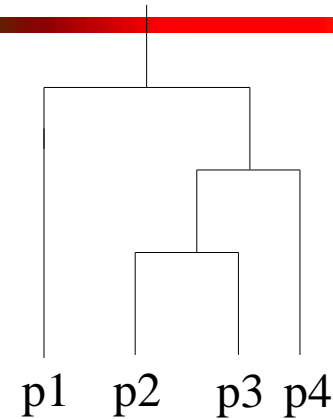


A Partitional Clustering

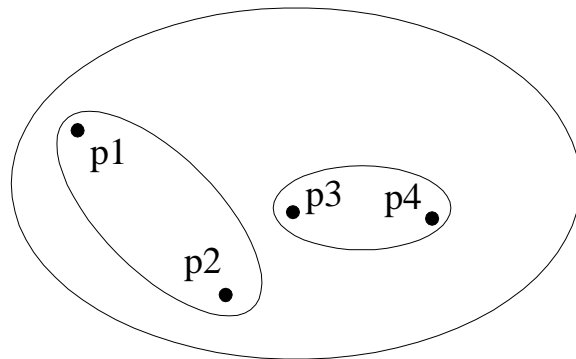
Hierarchical Clustering



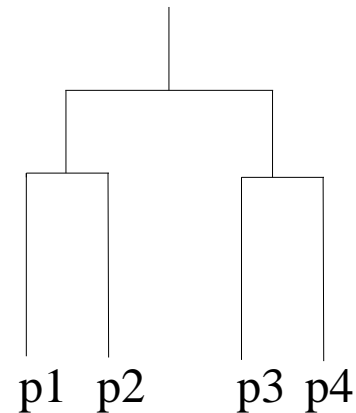
Traditional Hierarchical Clustering



Traditional Dendrogram



Non-traditional Hierarchical Clustering



Non-traditional Dendrogram



Clustering Models

– Partitional Clustering

K-Means

- Steps:

1. Select K points as the initial centroids
2. Repeat
3. Assign all points to the nearest centroid
4. Re-compute the centroid
5. Until all the centroids do not change

- Note:

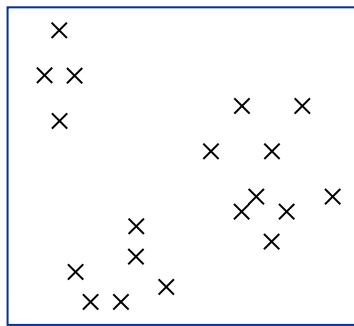
- Step (4) may not necessarily be a real point.

K-means Clustering – Details

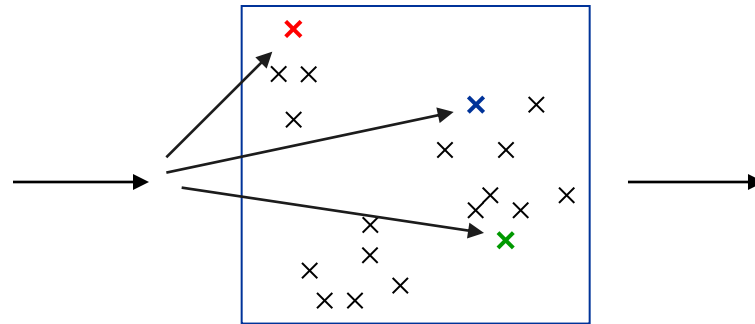
- Initial centroids are often chosen randomly
 - Clusters produced vary from one run to another.
- The centroid is (typically) the mean of the points in the cluster.
- ‘Closeness’ is measured by Euclidean distance, cosine similarity, correlation, etc.
- K-means will converge for common similarity measures mentioned above.
- Most of the convergence happens in the first few iterations.
 - Often the stopping condition is changed to ‘Until relatively few points change clusters’

K-Means

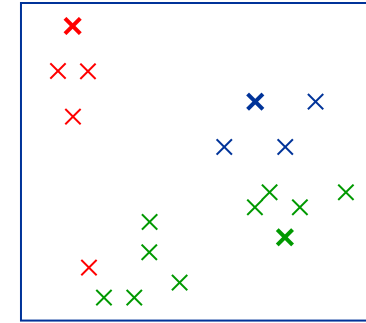
■ A simple example



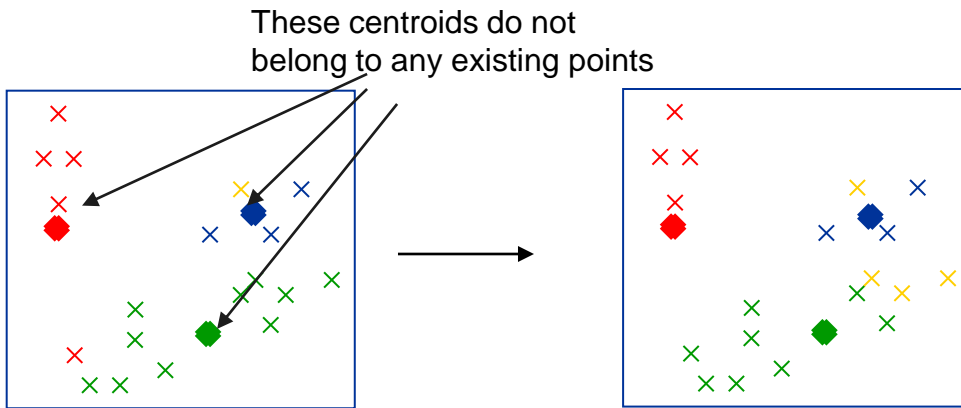
Initial data points



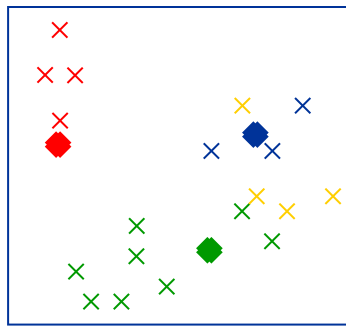
Randomly pick 3 data points as the centroids



Assign the data points to the nearest centroids



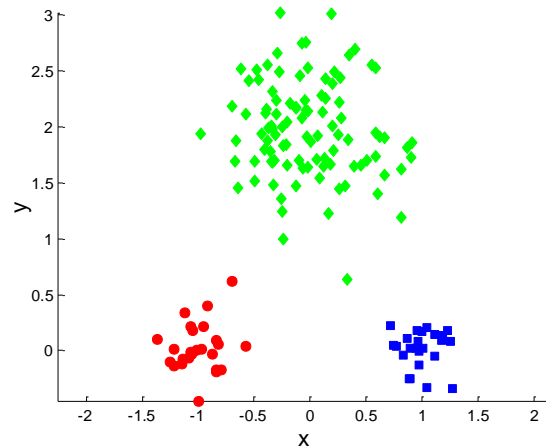
Re-compute the centroids of each cluster



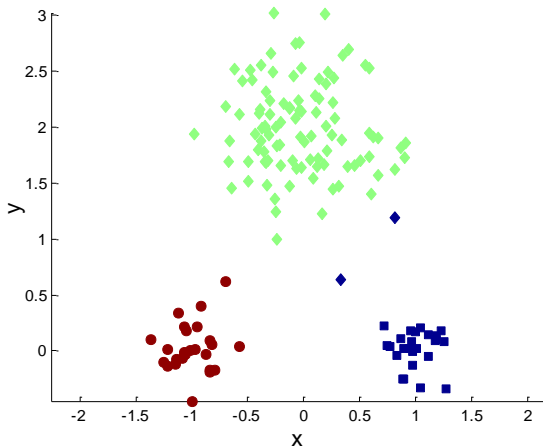
Assign the data points to the nearest centroids

Continue to:
1. Re-compute the centroids of each cluster;
2. Assign the data points to the nearest centroids.
Until the centroids do not change

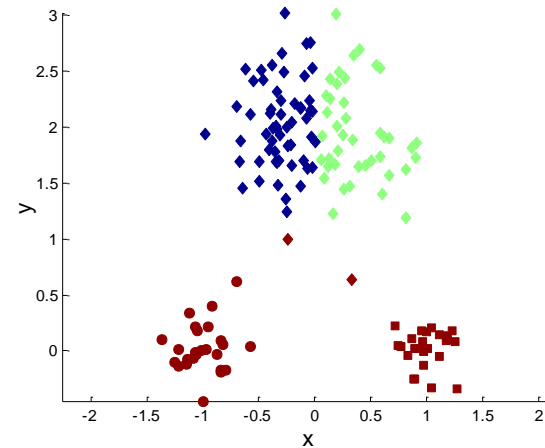
Two different K-means Clusterings



Original Points

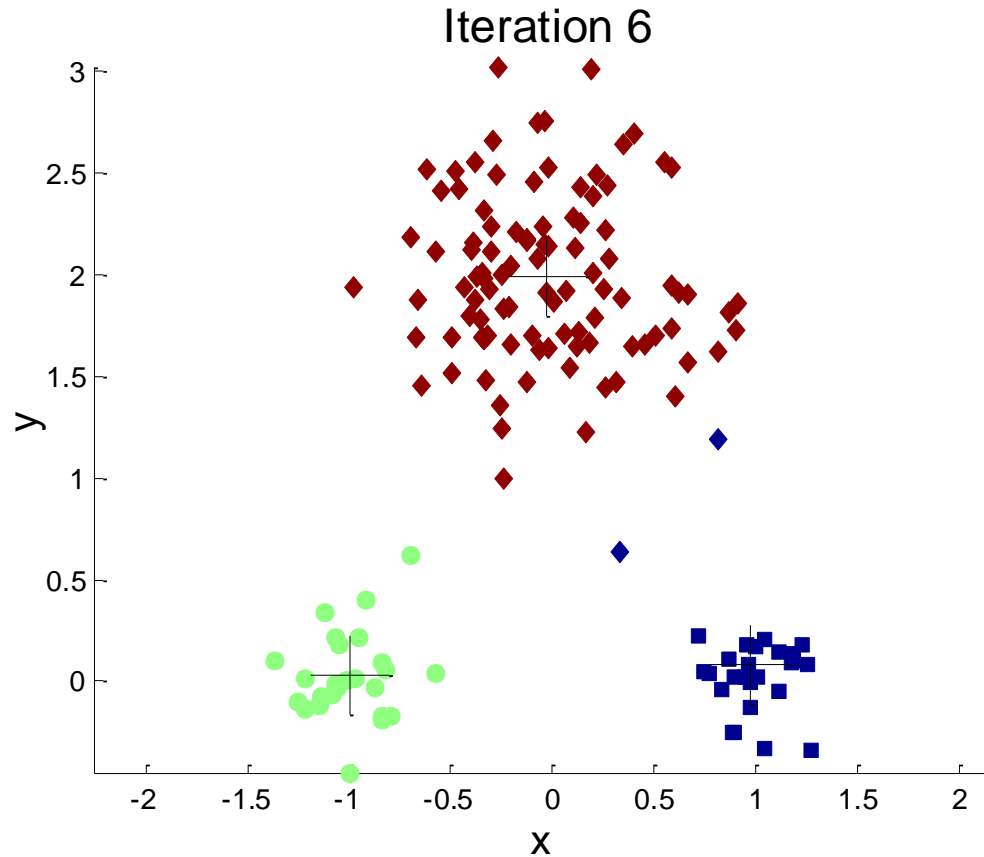


Optimal Clustering

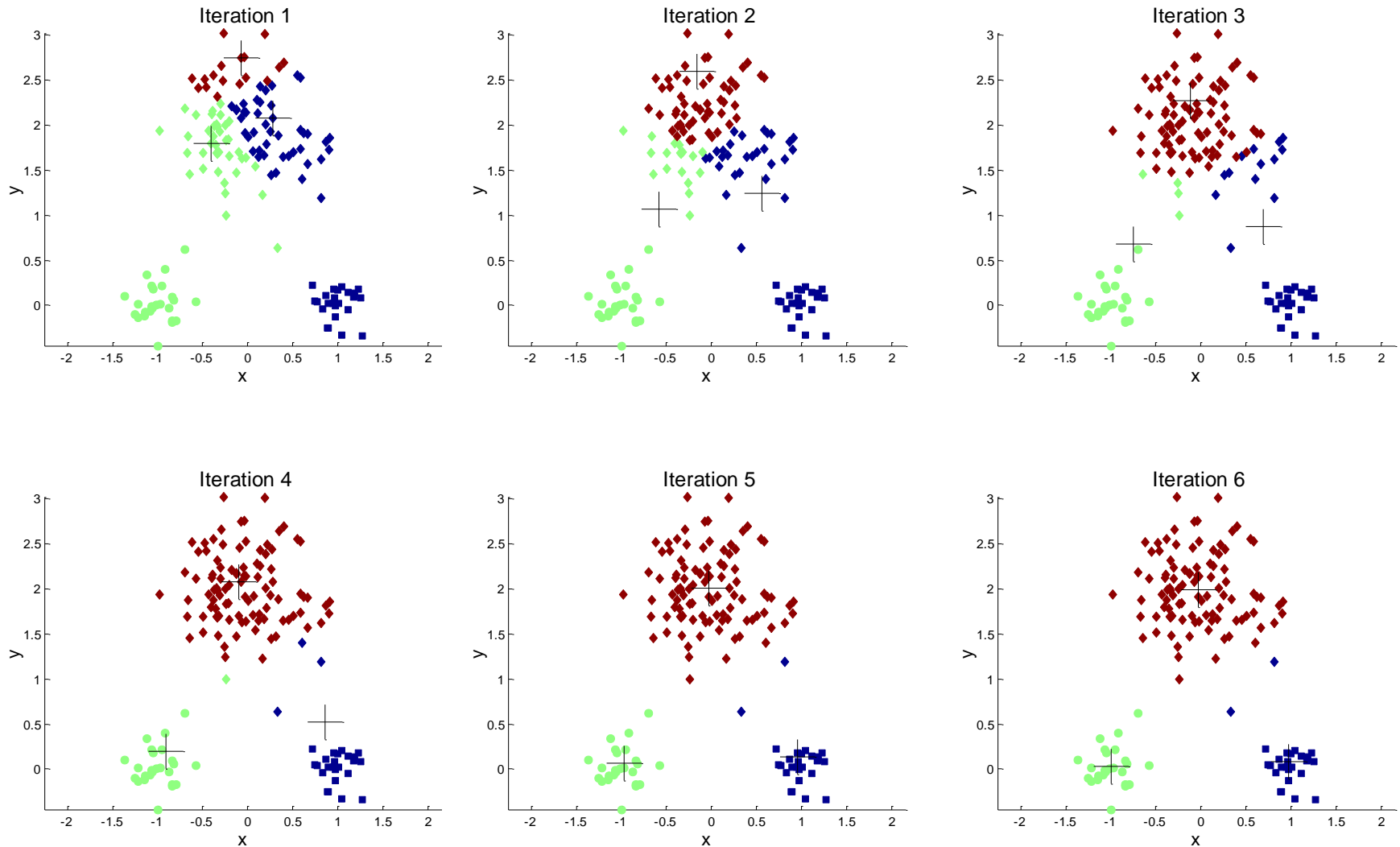


Sub-optimal Clustering

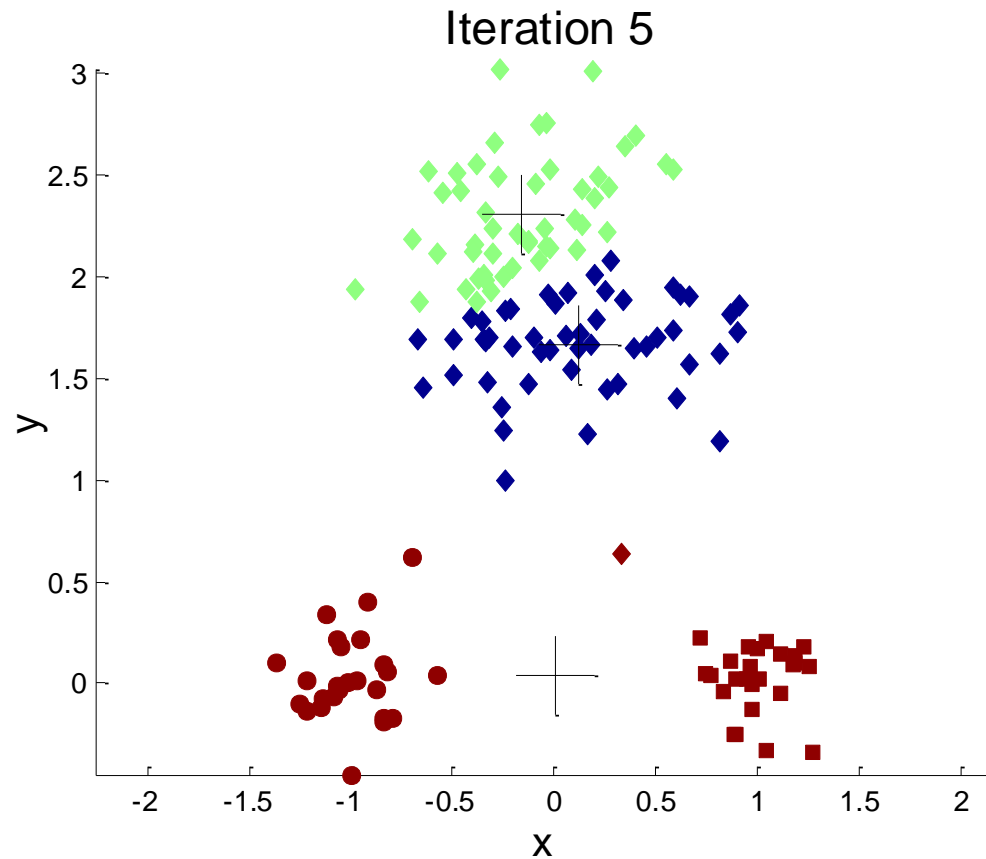
Importance of Choosing Initial Centroids



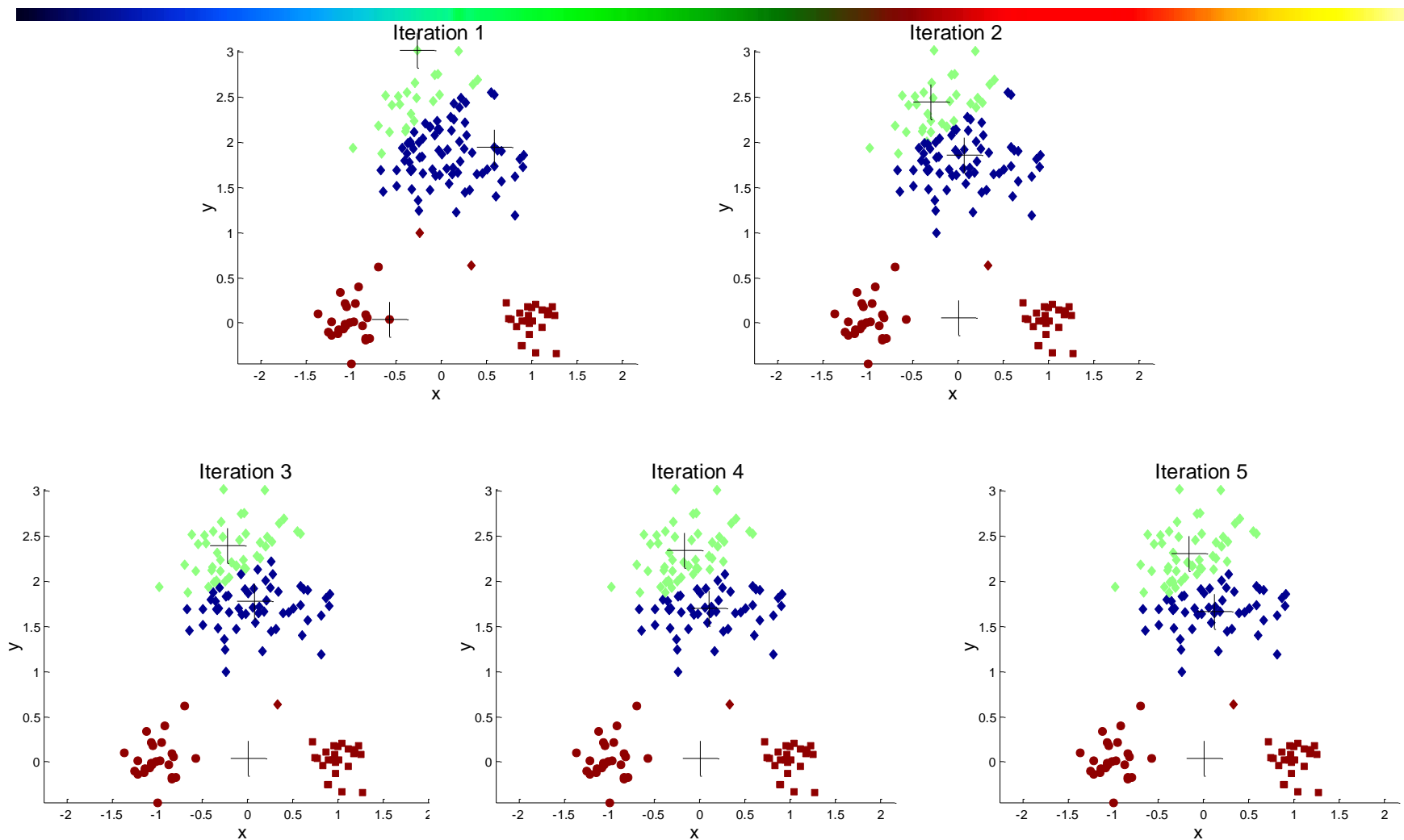
Importance of Choosing Initial Centroids



Importance of Choosing Initial Centroids



Importance of Choosing Initial Centroids



Evaluating K-means Clusters

- Most common measure is Sum of Squared Error (SSE)
 - For each point, the error is the distance to the nearest cluster center
 - To get SSE, we square these errors and sum them.

$$SSE = \sum_{i=1}^K \sum_{x \in C_i} dist^2(m_i, x)$$

- x is a data point in cluster C_i and m_i is the representative point for cluster C_i
 - can show that m_i corresponds to the center (mean) of the cluster
- Given two clusters, we can choose the one with the smallest error
- One easy way to reduce SSE is to increase K , the number of clusters
 - A good clustering with smaller K can have a lower SSE than a poor clustering with higher K

Quiz Time...





Suppose the data mining task is to cluster the following measurements of the variable *age* into three groups:

18, 22, 25, 42, 27, 43, 33, 35, 56, 28,

1. Use *k-means* algorithm to show the clustering procedures **step by step**; and
2. calculate corresponding SSE values.

Note that:

- a) Suppose the initial centroids are **22, 35** and **43**. Show the final three clusters.
- a) Suppose the initial centroids are **18, 27** and **35** . Show the final three clusters.

$$SSE = \sum_{i=1}^K \sum_{x \in C_i} dist^2(m_i, x)$$

Initial centroids: 22, 35, 43

| Cluster# | Old Centroid | Cluster Elements | new Centroid |
|----------|--------------|------------------|--------------|
| 1 | 22 | 18,22,25,27,28 | 24 |
| 2 | 35 | 33,35 | 34 |
| 3 | 43 | 42,43,56 | 47 |



| Cluster# | Old Centroid | Cluster Elements | new Centroid |
|----------|--------------|------------------|--------------|
| 1 | 24 | 18,22,25,27,28 | 24 |
| 2 | 34 | 33,35 | 34 |
| 3 | 47 | 42,43,56 | 47 |

SSE = 190

Initial centroids: 18, 27, 35

| Cluster# | Old Centroid | Cluster Elements | new Centroid | |
|----------|--------------|--------------------|--------------|--------|
| 1 | 18 | 18,22 | 20 | ROUND1 |
| 2 | 27 | 25,27,28 | 26.7 | |
| 3 | 35 | 33,35,42,43,56 | 41.8 | |
| 1 | 20 | 18, 22 | 20 | ROUND2 |
| 2 | 26.7 | 25, 27, 28, 33 | 28.25 | |
| 3 | 41.8 | 35, 42, 43, 56 | 44 | |
| 1 | 20 | 18, 22 | 20 | ROUND3 |
| 2 | 28.25 | 25, 27, 28, 33, 35 | 29.6 | |
| 3 | 44 | 42, 43, 56 | 47 | |
| 1 | 20 | 18, 22 | 20 | ROUND4 |
| 2 | 29.6 | 25, 27, 28, 33, 35 | 29.6 | |
| 3 | 47 | 42, 43, 56 | 47 | |

SSE = 201.2

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
- Bisecting K-means
 - Not as susceptible to initialization issues

Handling Empty Clusters

- Basic K-means algorithm can yield empty clusters
- Several strategies to redefine the centroids of empty-clusters
 - Choose the point that contributes most to SSE (from a non-empty cluster)
 - Choose a point from the cluster with the highest SSE
 - If there are several empty clusters, the above can be repeated several times.

Pre-processing and Post-processing

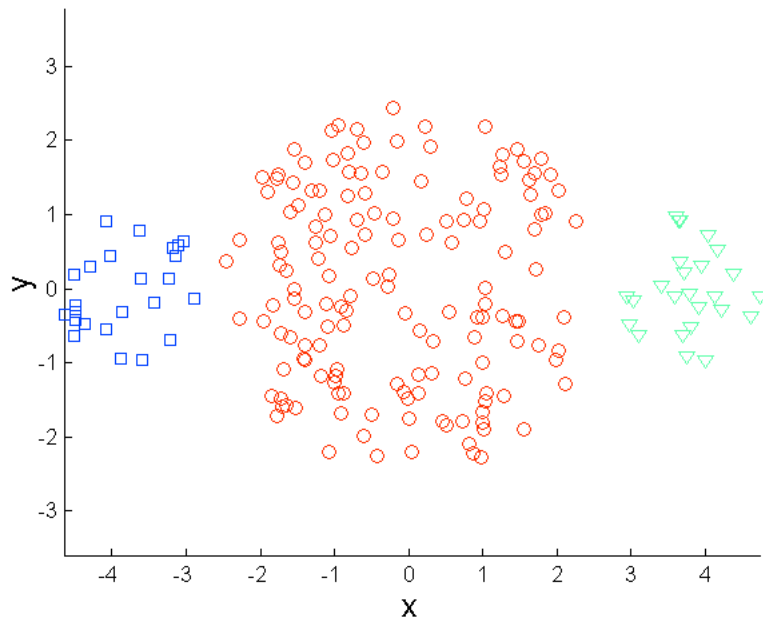
- Pre-processing
 - Normalize the data
 - Eliminate outliers
- Post-processing
 - Eliminate small clusters that may represent outliers
 - Split 'loose' clusters, i.e., clusters with relatively high SSE
 - Merge clusters that are 'close' and that have relatively low SSE
 - Can use these steps during the clustering process

Limitations of K-means

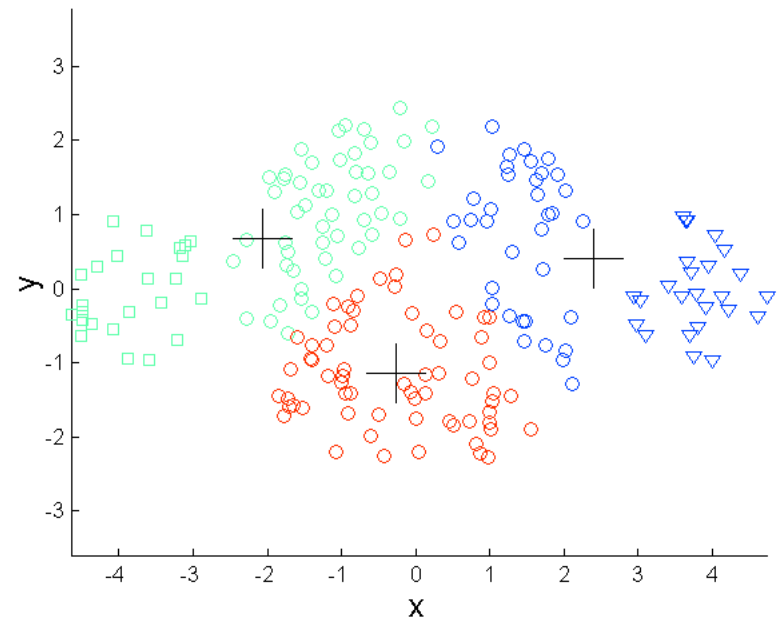
- K-means has problems when clusters are of differing
 - Sizes
 - Densities
 - Non-globular shapes
- K-means has problems when the data contains outliers.

Limitations of K-Means

- K-means has problems when clusters are:
 - Different sizes



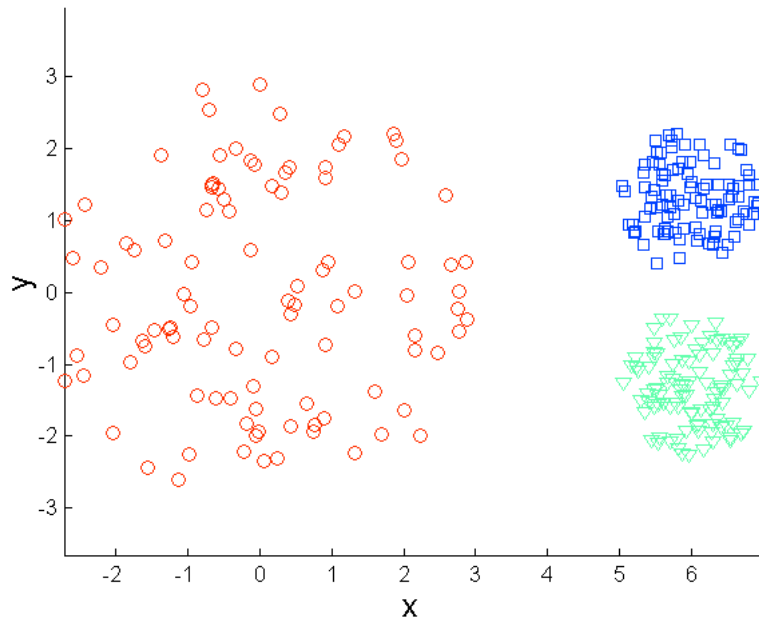
Original Points



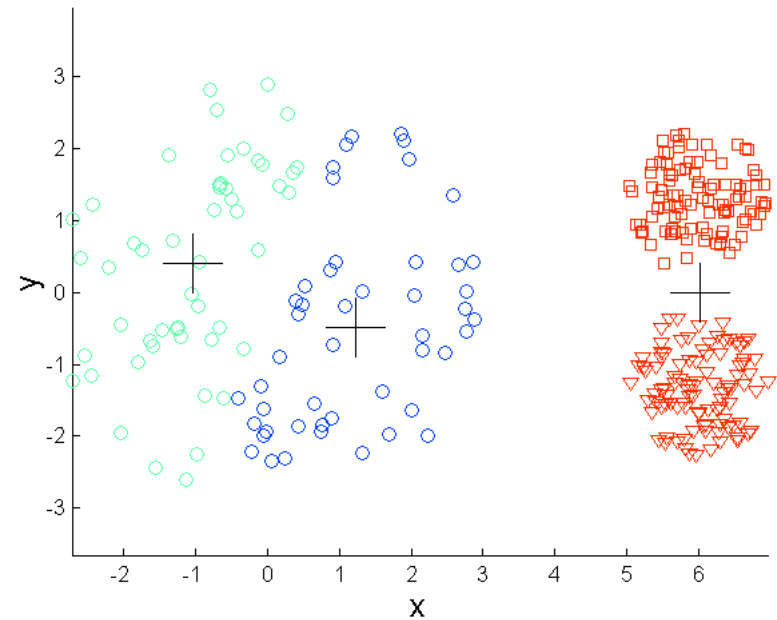
K-means (3 Clusters)

Limitations of K-means

- K-means has problems when clusters are:
 - Different densities



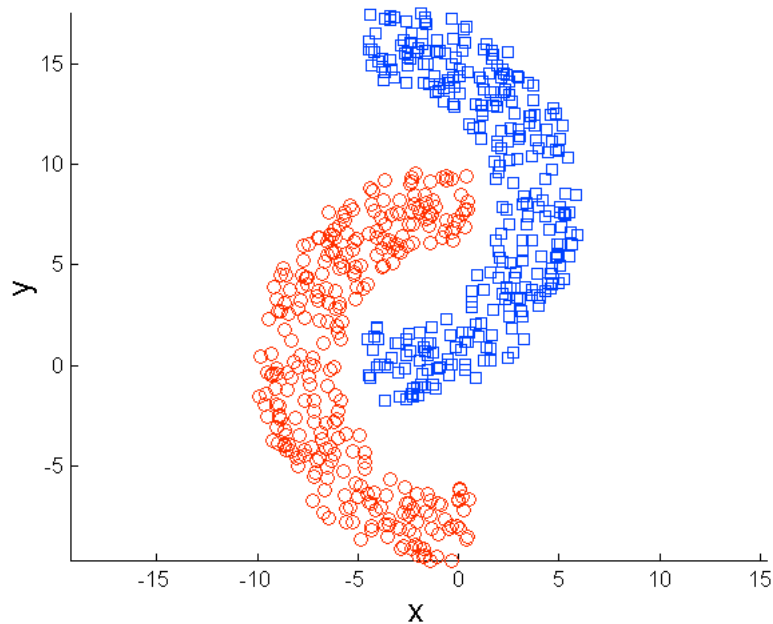
Original Points



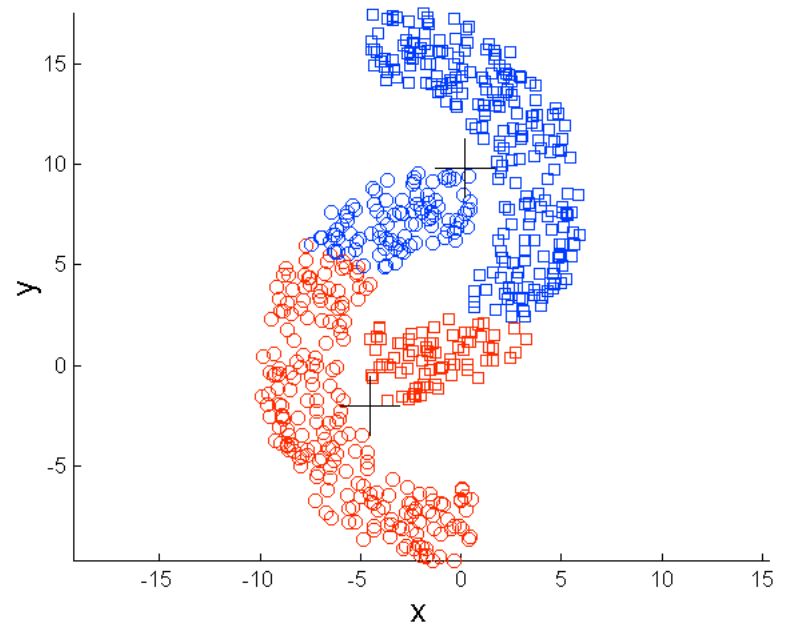
K-means (3 Clusters)

Limitations of K-means

- K-means has problems when clusters are:
 - Non-globular shapes



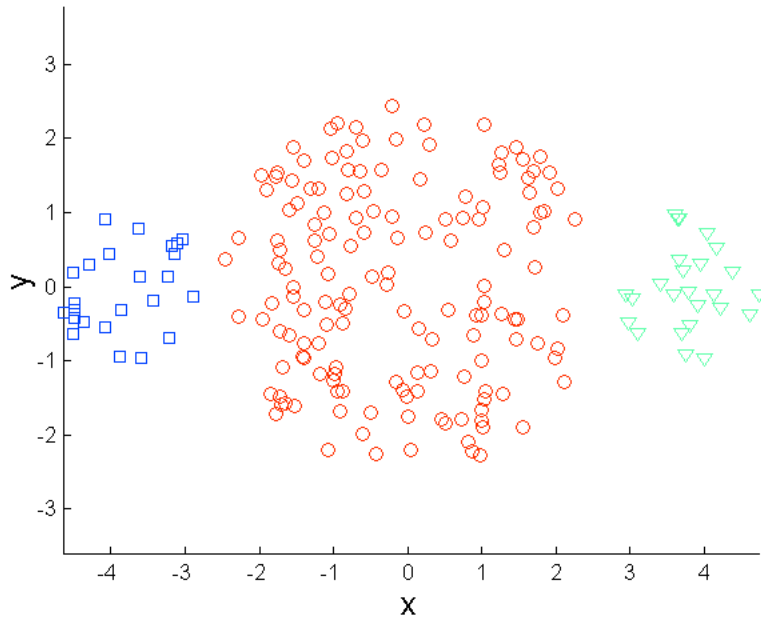
Original Points



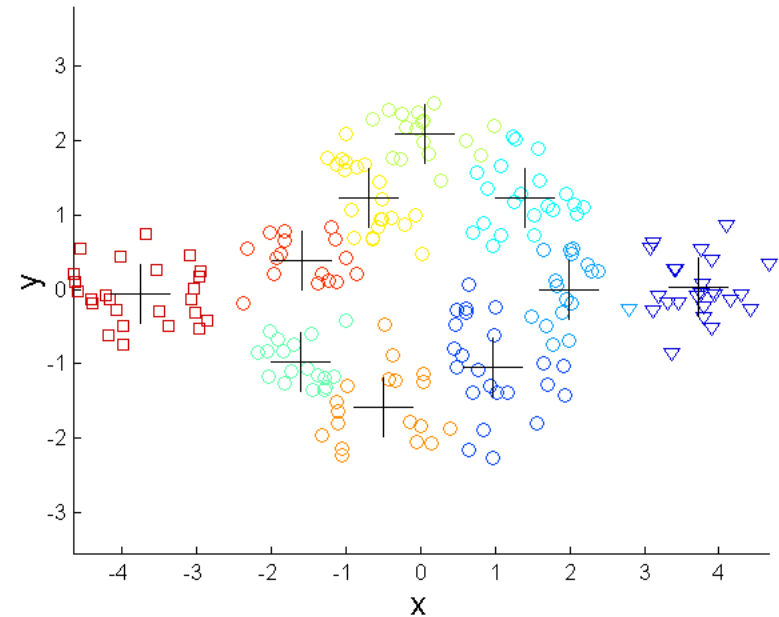
K-means (2 Clusters)

Overcoming Limitations of K-means

- One solution is to use many clusters
 - Find parts of clusters, but need to put together.



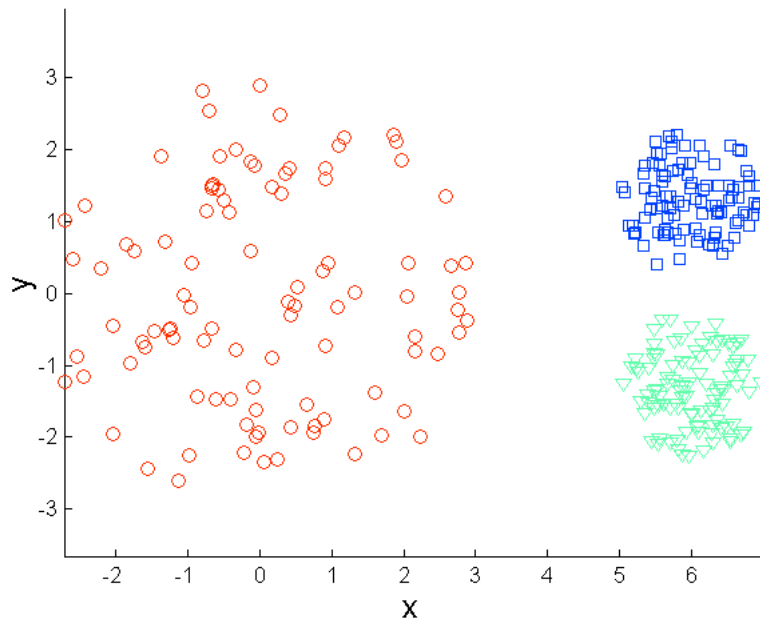
Original Points



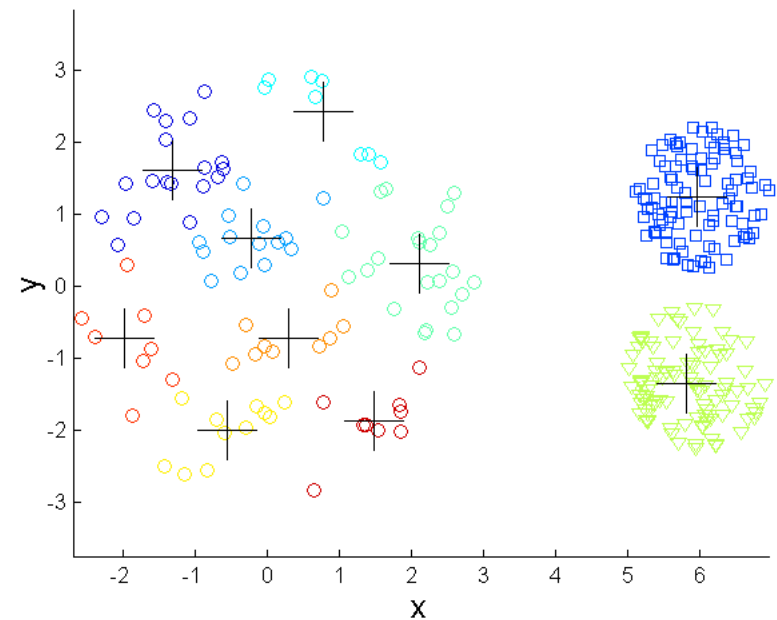
K-means Clusters

Overcoming Limitations of K-means

- One solution is to use many clusters
 - Find parts of clusters, but need to put together.



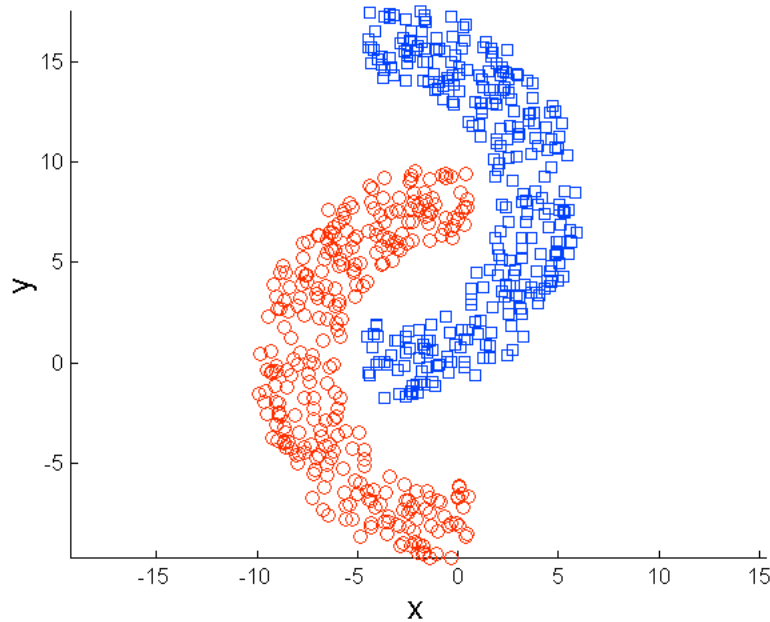
Original Points



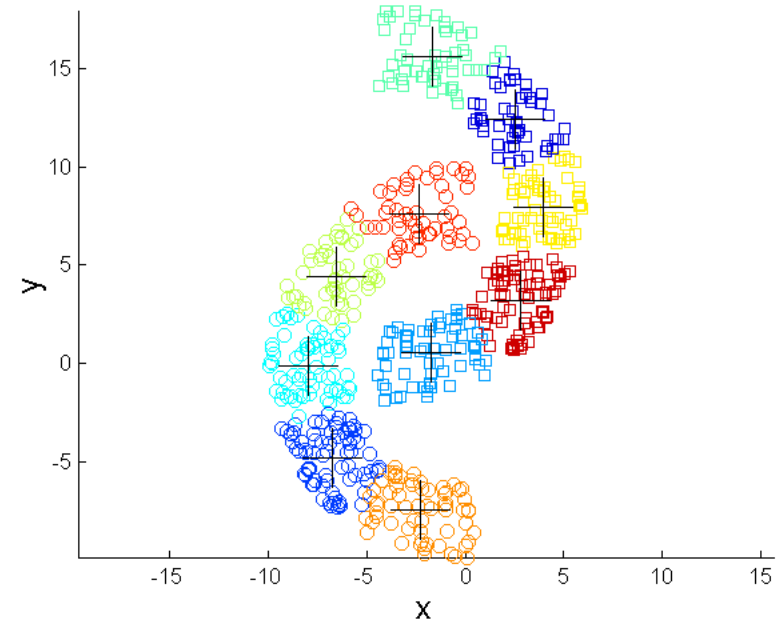
K-means Clusters

Overcoming Limitations of K-means

- One solution is to use many clusters
 - Find parts of clusters, but need to put together.



Original Points



K-means Clusters



Clustering Models

- Agglomerative Hierarchical Clustering

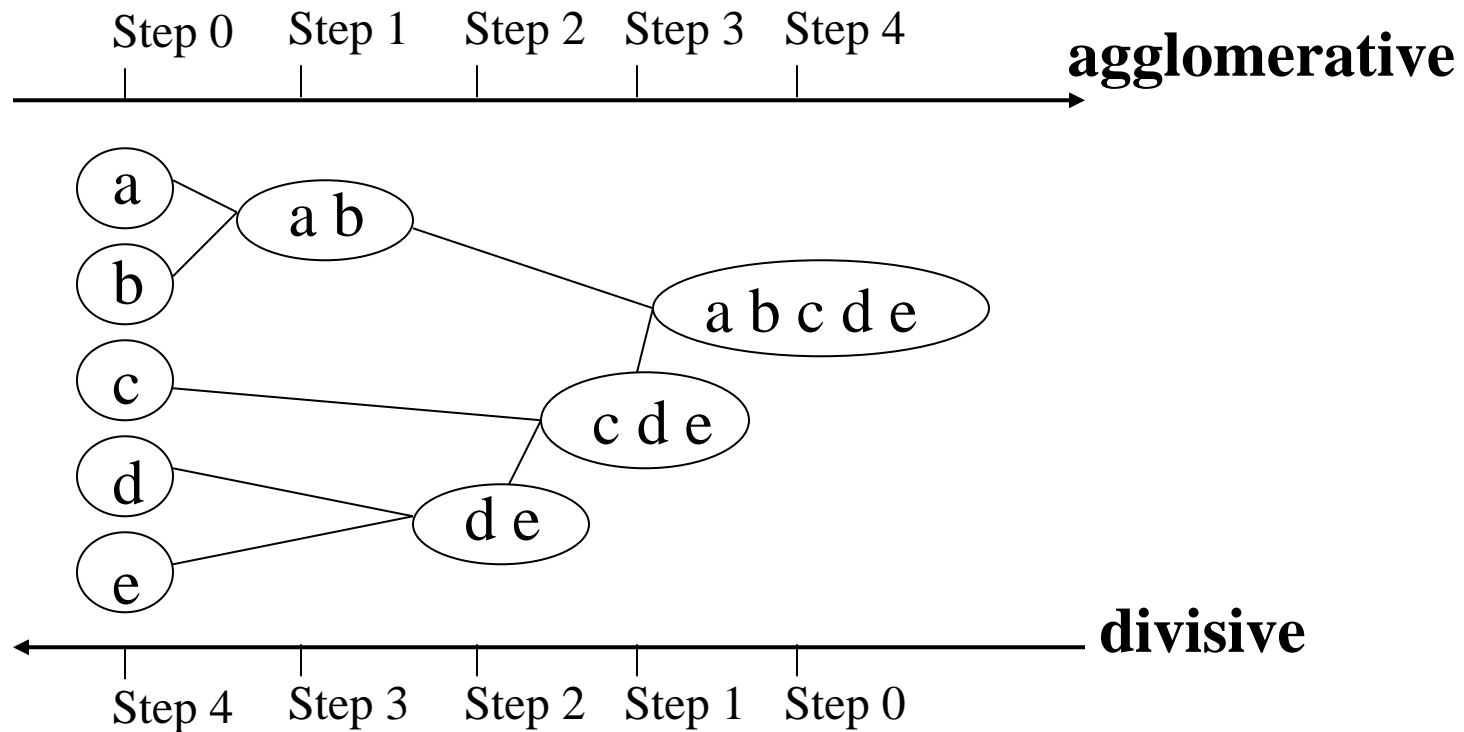
Strengths of Hierarchical Clustering

- Do not have to assume any particular number of clusters
 - Any desired number of clusters can be obtained by 'cutting' the dendrogram at the proper level
- They may correspond to meaningful taxonomies
 - Example in biological sciences (e.g., animal kingdom, phylogeny reconstruction, ...)

Hierarchical Clustering

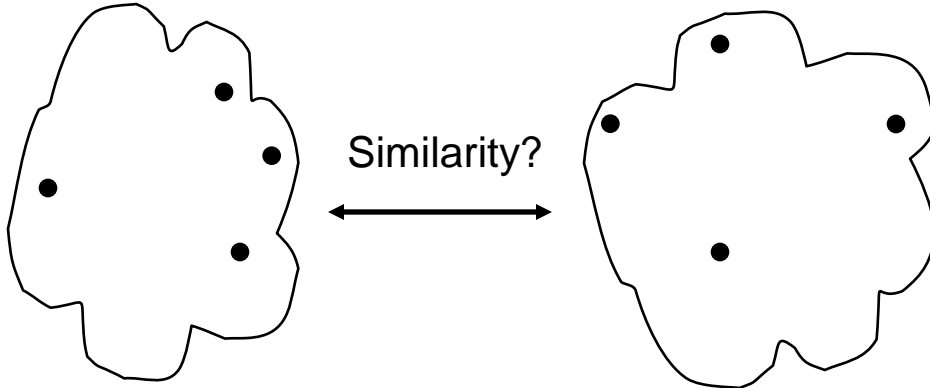
- Two main types of hierarchical clustering
 - Agglomerative:
 - Start with the points as individual clusters
 - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
 - Divisive:
 - Start with one, all-inclusive cluster
 - At each step, split a cluster until each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

An Example



Agglomerative Hierarchical Clustering

- Use distance matrix as clustering criteria.
 - This method does not require the number of clusters (K) as an input, but needs a termination condition



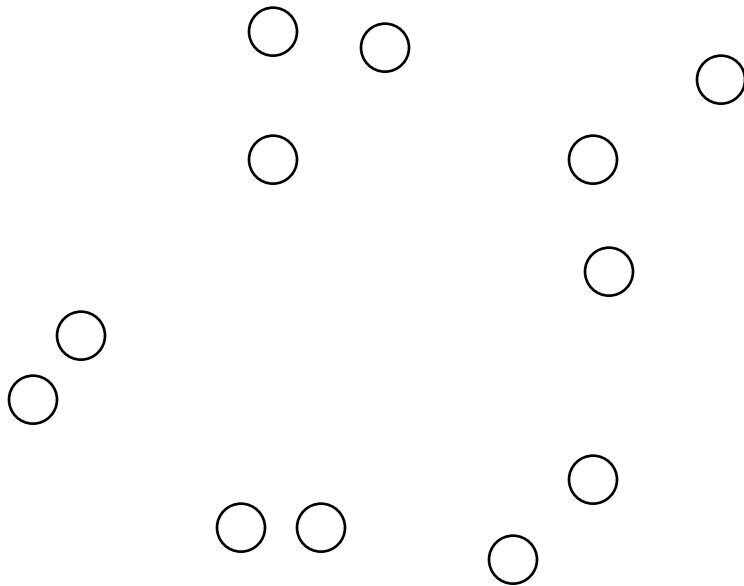
- Four methods:
 - Min (a.k.a. single linkage)
 - Max (a.k.a. complete linkage)
 - Group average
 - Distance between centroids

Agglomerative Clustering Algorithm

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

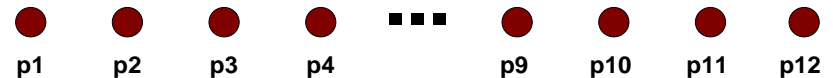
Starting Situation

- Start with clusters of individual points and a proximity matrix



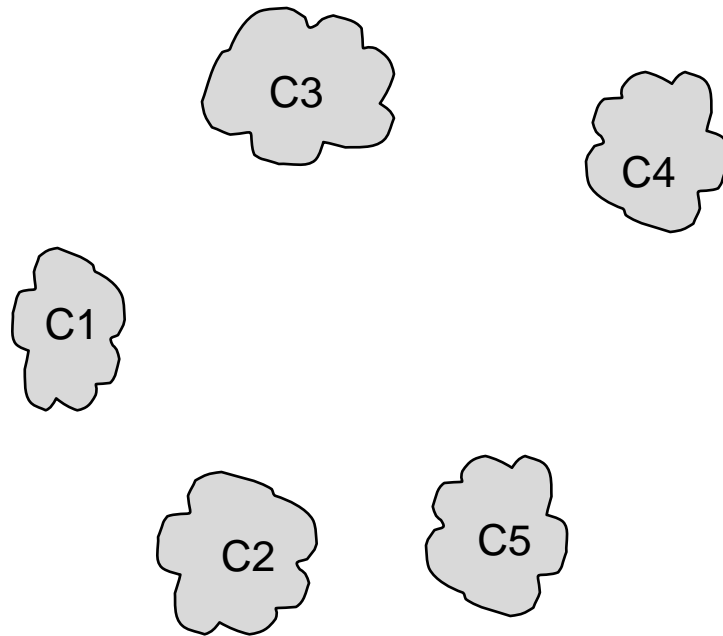
| | p1 | p2 | p3 | p4 | p5 | ... |
|----|----|----|----|----|----|-----|
| p1 | | | | | | |
| p2 | | | | | | |
| p3 | | | | | | |
| p4 | | | | | | |
| p5 | | | | | | |
| . | | | | | | |
| . | | | | | | |
| . | | | | | | |

Proximity Matrix



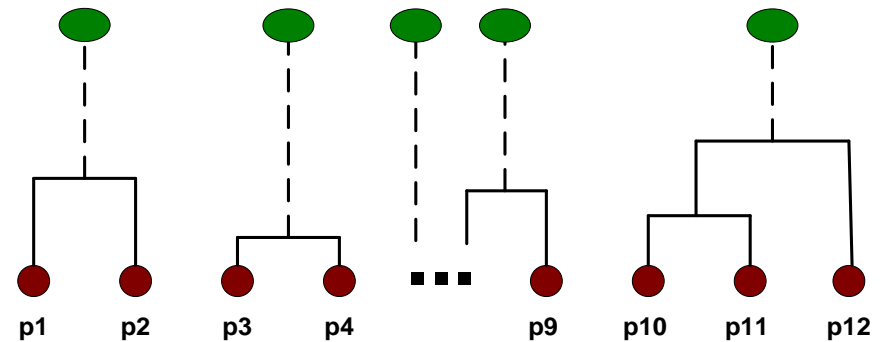
Intermediate Situation

- After some merging steps, we have some clusters



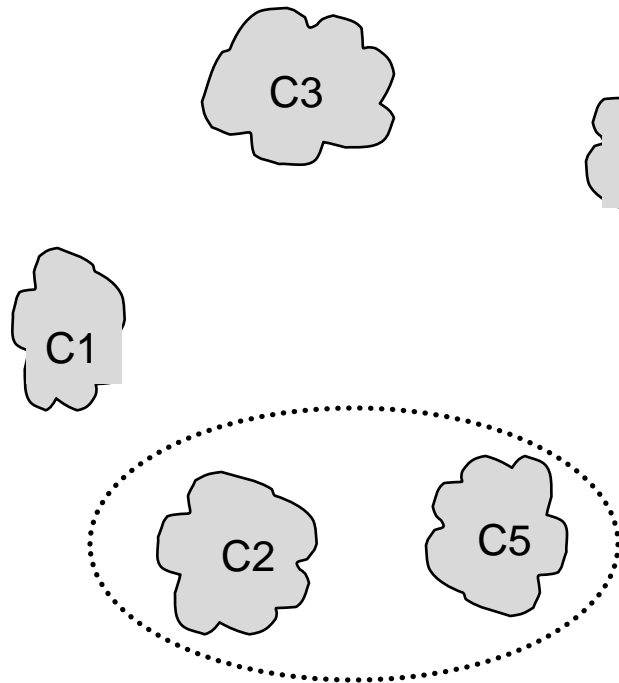
| | C1 | C2 | C3 | C4 | C5 |
|----|----|----|----|----|----|
| C1 | | | | | |
| C2 | | | | | |
| C3 | | | | | |
| C4 | | | | | |
| C5 | | | | | |

Proximity Matrix



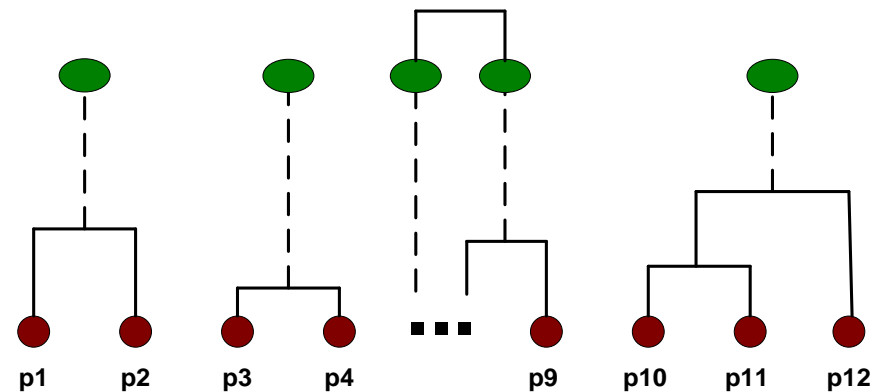
Intermediate Situation

- We want to merge the two closest clusters (C2 and C5) and update the proximity matrix.



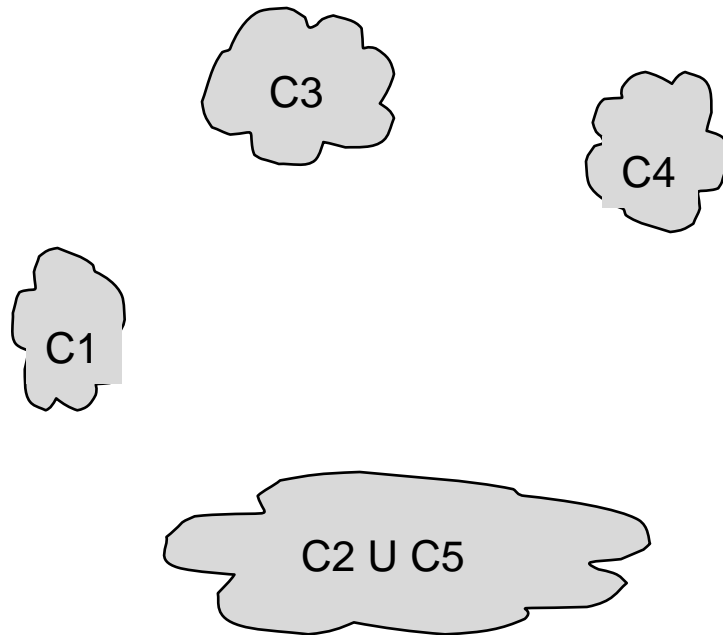
| | C1 | C2 | C3 | C4 | C5 |
|----|----|----|----|----|----|
| C1 | | | | | |
| C2 | | | | | |
| C3 | | | | | |
| C4 | | | | | |
| C5 | | | | | |

Proximity Matrix



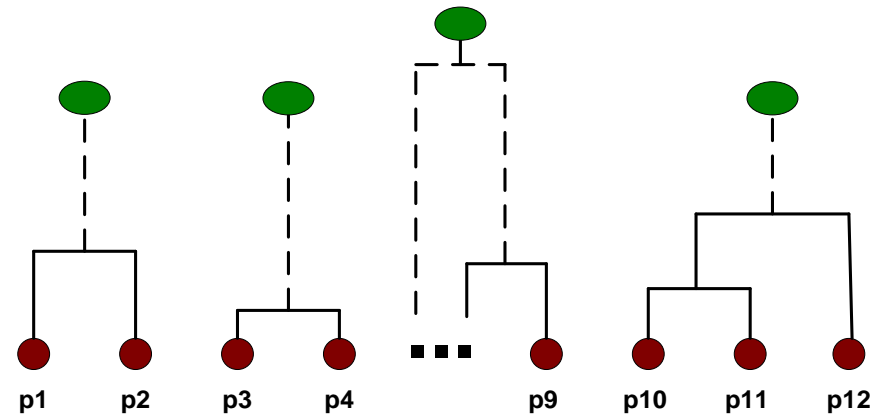
After Merging

- The question is "How do we update the proximity matrix?"

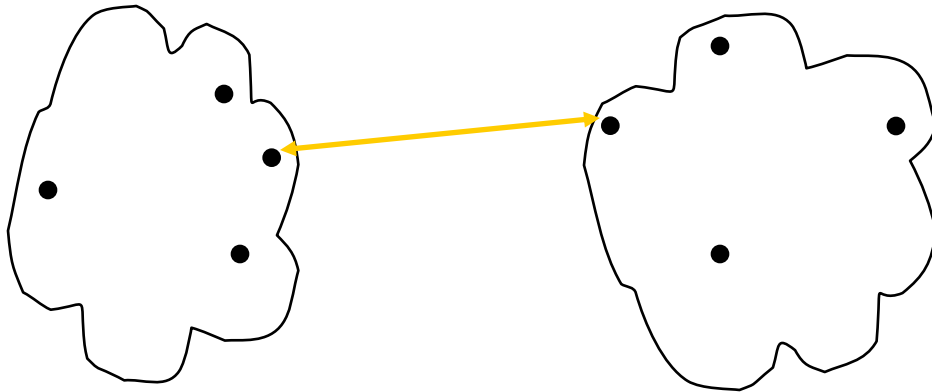


| | C1 | C2 U C5 | C3 | C4 |
|---------|----|---------------|----|----|
| C1 | | ? | | |
| C2 U C5 | ? | ? | ? | ? |
| C3 | | ? | | |
| C4 | | ? | | |

Proximity Matrix

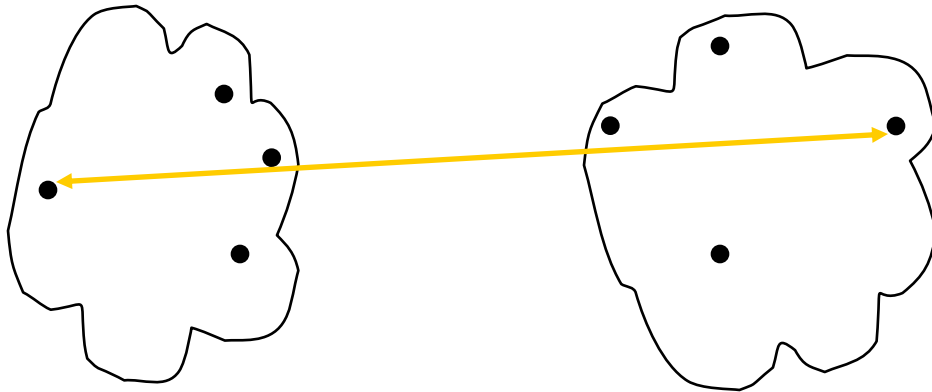


How to Define Inter-Cluster Similarity



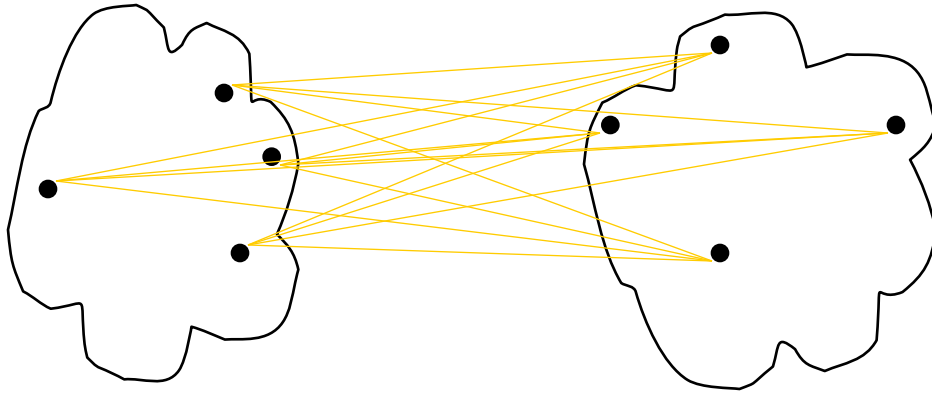
- **Min (a.k.a. single linkage)**
 - two most similar (closest) points in the different clusters
 - Determined by one pair of points, i.e., by one link in the proximity graph.

How to Define Inter-Cluster Similarity



- **Max (a.k.a. complete linkage)**
 - two least similar (most distant) points in the different clusters
 - Determined by all pairs of points in the two clusters

How to Define Inter-Cluster Similarity

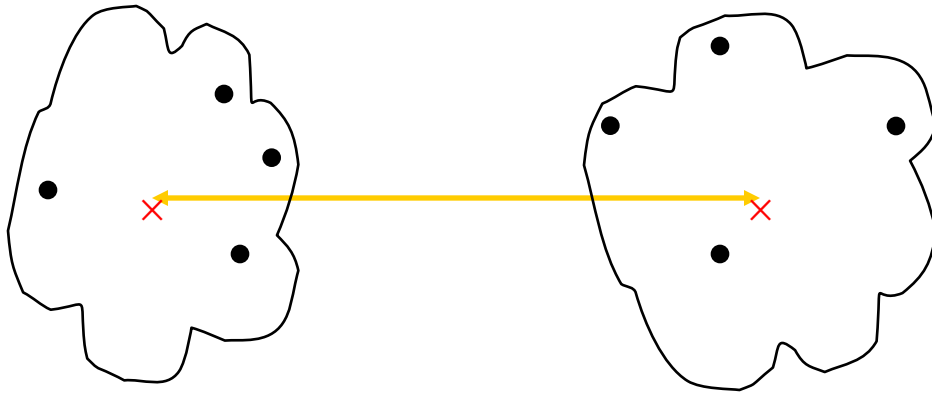


- **Group average**

- the average of pairwise proximity between points in the two clusters.

$$\text{proximity}(\text{Cluster}_i, \text{Cluster}_j) = \frac{\sum_{\substack{p_i \in \text{Cluster}_i \\ p_j \in \text{Cluster}_j}} \text{proximity}(p_i, p_j)}{|\text{Cluster}_i| * |\text{Cluster}_j|}$$

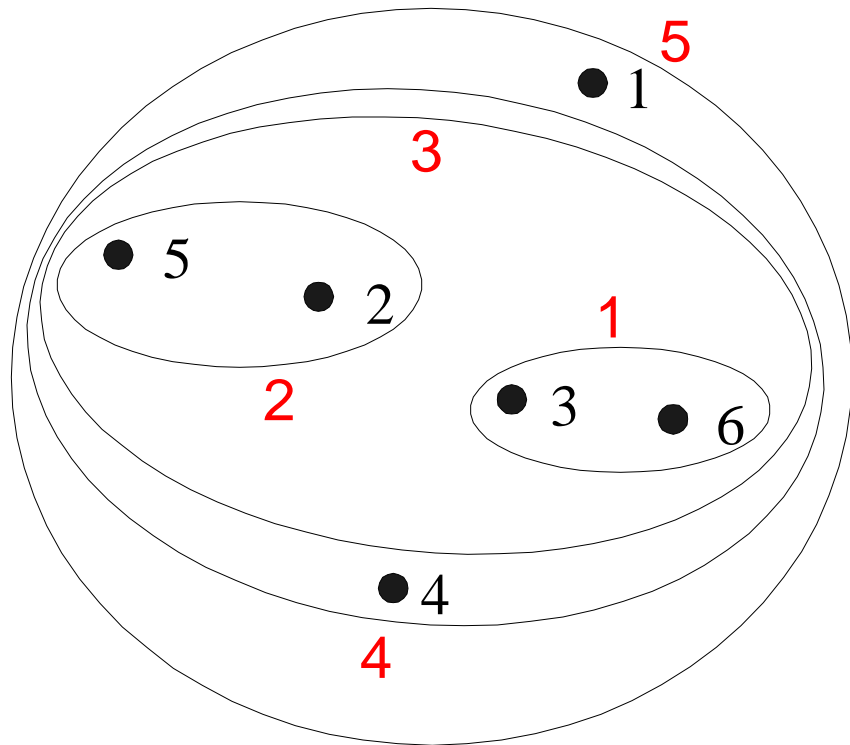
How to Define Inter-Cluster Similarity



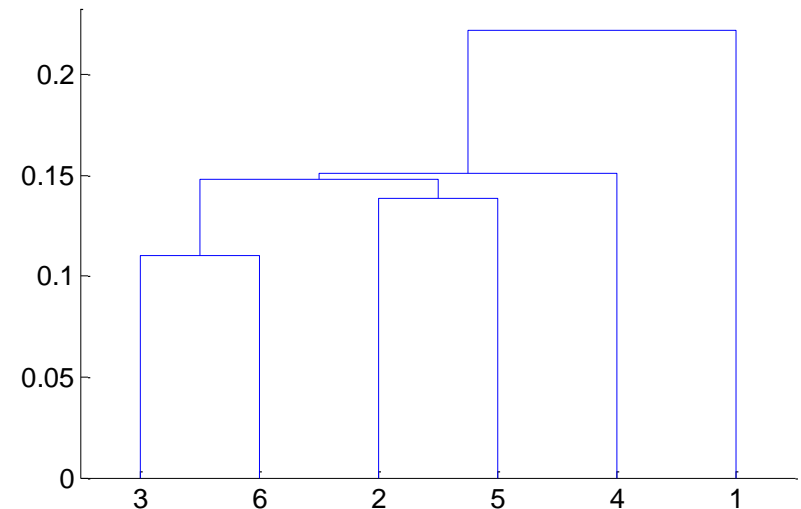
- Distance between centroids

Min (Single Linkage)

- Similarity of two clusters is based on the two most similar (closest) points in the different clusters



Nested Clusters

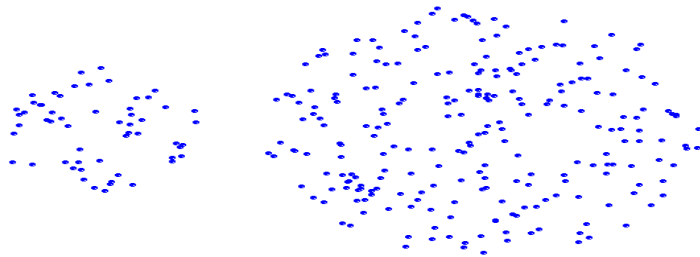


Dendrogram

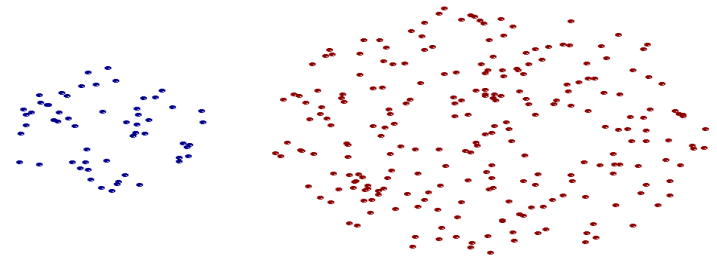
Strength and Limitation of Min

- Strength:

- Can handle non-elliptical shapes



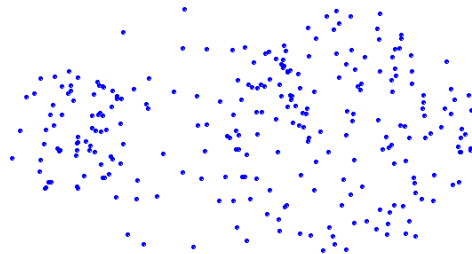
Original Points



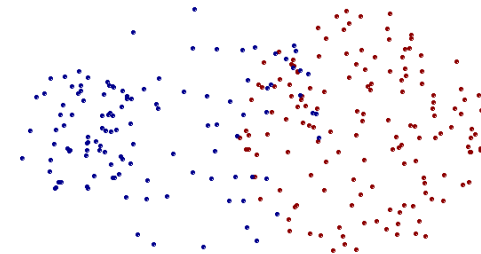
2 Clusters

- Limitation:

- Sensitive to noise



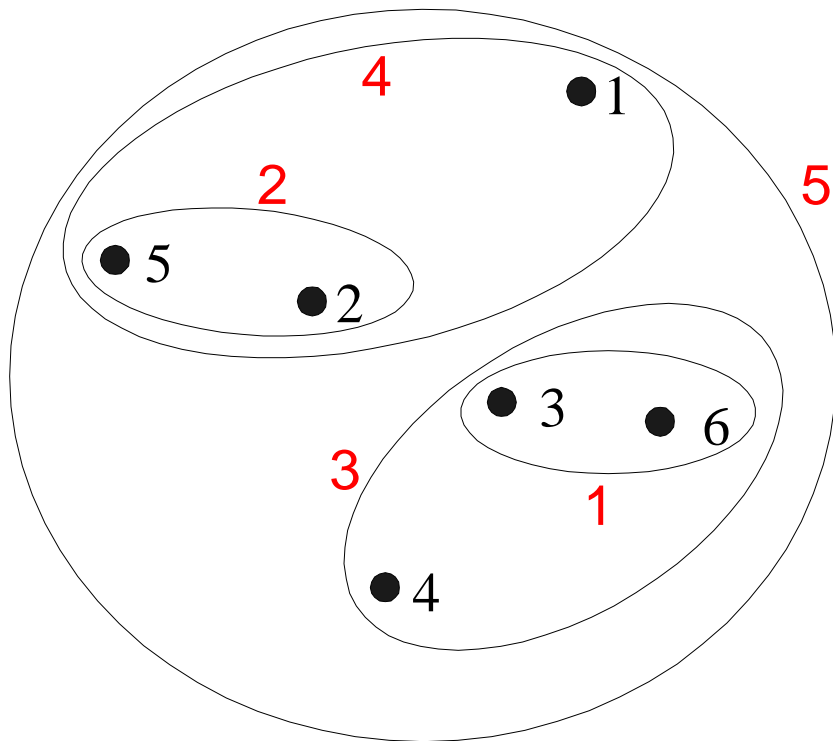
Original Points



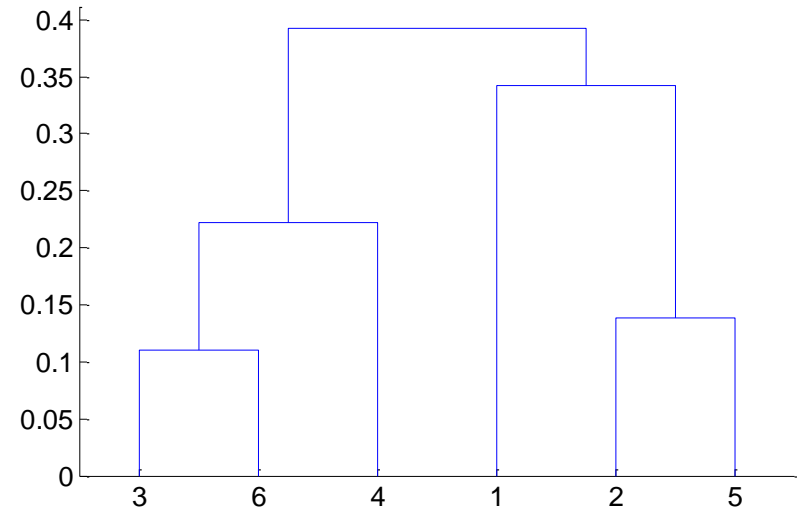
2 Clusters

Max (Complete Linkage)

- Similarity of two clusters is based on the two least similar (most distant) points in the different clusters



Nested Clusters

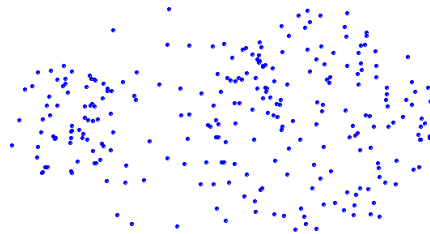


Dendrogram

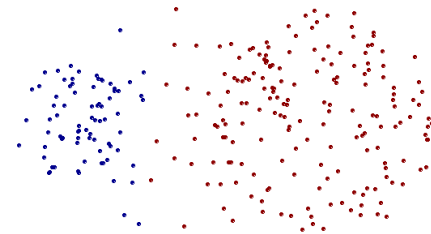
Strength and Limitation of Max (cont'd)

- Strength

- Less susceptible to noise



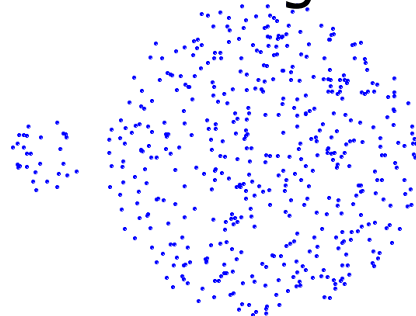
Original Points



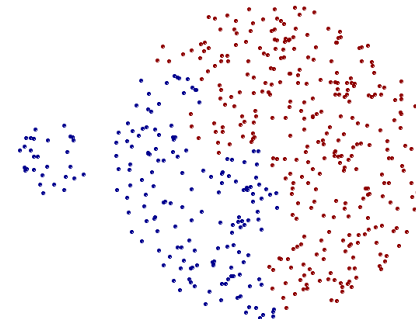
2 Clusters

- Limitation

- Tends to break large clusters



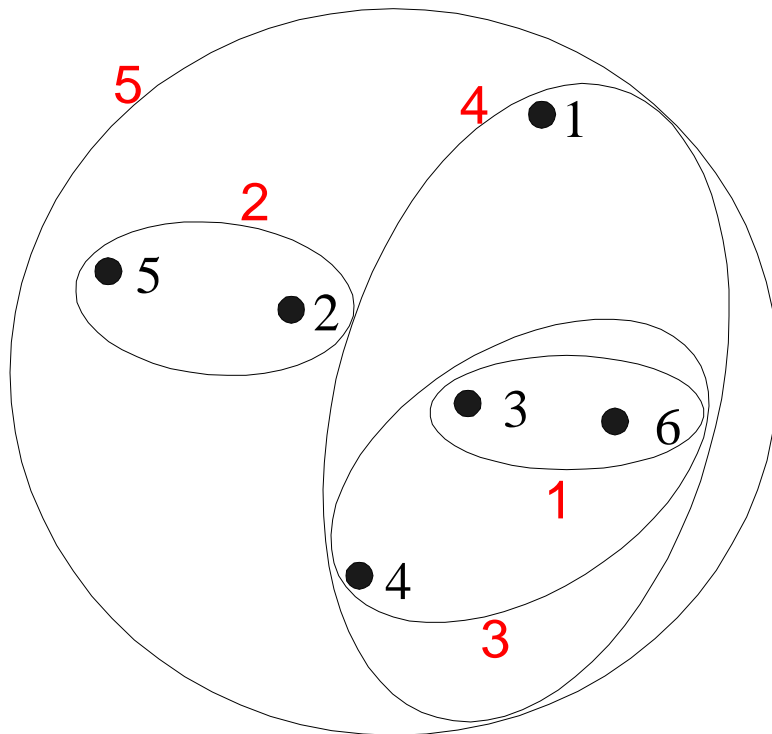
Original Points



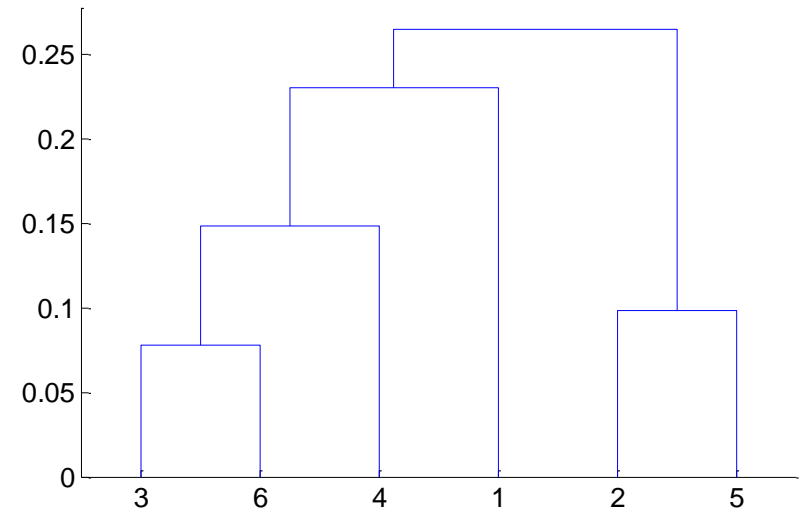
2 Clusters

Group Average

- Compromise between Single and Complete Link
- As the name implies, use the average pairwise distance between points in the two clusters

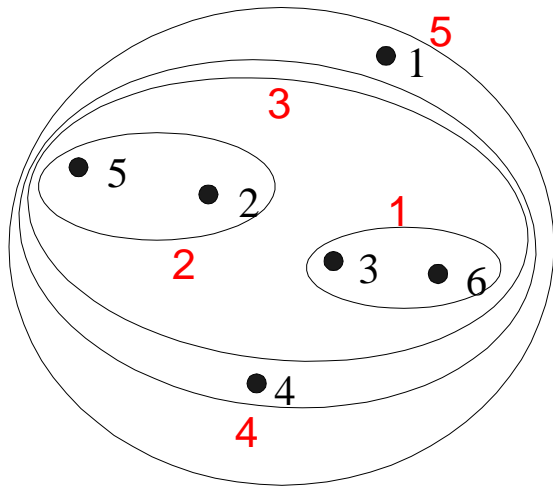


Nested Clusters

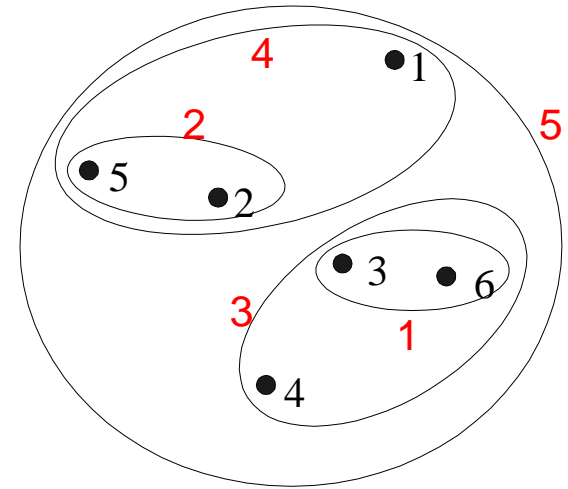


Dendrogram

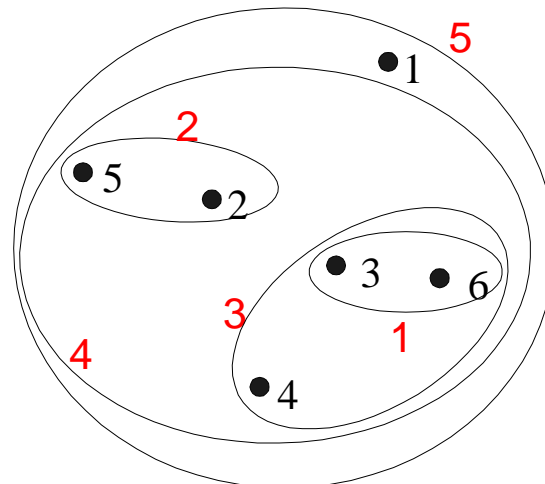
Putting All Together



MIN



MAX



Group Average

Quiz Time...





Given a set of numbers,

18, 22, 25, 42, 27, and 43,

use *Agglomerative Hierarchical Clustering* algorithm to group them step by step.

Use *min* to merge two closest clusters and update Proximity Matrix correspondingly.

| | | | | | | |
|----|----|----|----|----------|----|----------|
| | 18 | 22 | 25 | 42 | 27 | 43 |
| 18 | 0 | 4 | 7 | 24 | 9 | 25 |
| 22 | 4 | 0 | 3 | 20 | 5 | 21 |
| 25 | 7 | 3 | 0 | 17 | 2 | 18 |
| 42 | 24 | 20 | 17 | 0 | 15 | 1 |
| 27 | 9 | 5 | 2 | 15 | 0 | 16 |
| 43 | 25 | 21 | 18 | 1 | 16 | 0 |

| | | | | | | |
|-------|----|----|----------|----------|----|----|
| | 18 | 22 | 25 | 27 | 42 | 43 |
| 18 | 0 | 4 | 7 | 9 | 24 | |
| 22 | 4 | 0 | 3 | 5 | 20 | |
| 25 | 7 | 3 | 0 | 2 | 17 | |
| 27 | 9 | 5 | 2 | 0 | 15 | |
| 42 43 | 24 | 20 | 17 | 15 | 0 | |

25,27

| | | | | |
|-------|----|----------|----------|-------|
| | 18 | 22 | 25 27 | 42 43 |
| 18 | 0 | 4 | 7 | 24 |
| 22 | 4 | 0 | 3 | 20 |
| 25 27 | 7 | 3 | 0 | 15 |
| 42 43 | 24 | 20 | 15 | 0 |

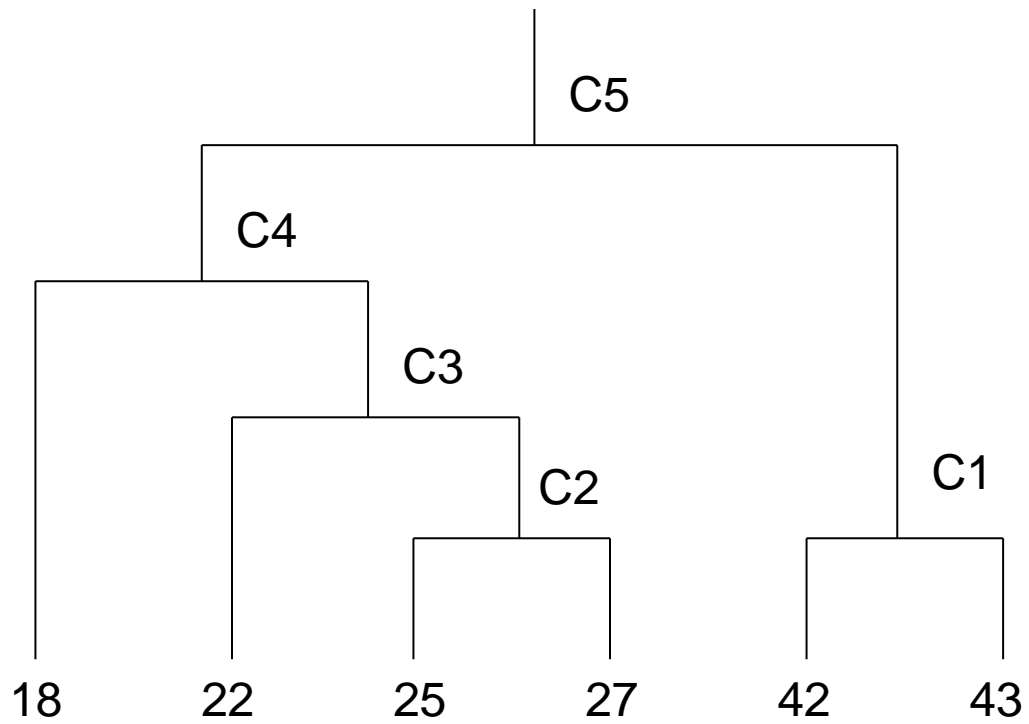
| | | | |
|----------|----------|----------|-------|
| | 18 | 22 25 27 | 42 43 |
| 18 | 0 | 4 | 24 |
| 22 25 27 | 4 | 0 | 15 |
| 42 43 | 24 | 15 | 0 |

22,25,27

18,22,25,27

| | | |
|-------------|-------------|-------|
| | 18 22 25 27 | 42 43 |
| 18 22 25 27 | 0 | 15 |
| 42 43 | 15 | 0 |

18,22,25,27,42,43



Hierarchical Clustering: Problems and Limitations

- Once a decision is made to combine two clusters, it cannot be undone
- No objective function is directly minimized
- Different schemes have problems with one or more of the following:
 - Sensitivity to noise and outliers
 - Difficulty handling different sized clusters and convex shapes
 - Breaking large clusters



Appendix

Mathematical Details

- Euclidean distance:


$$d(P, Q) = \sqrt{\sum (p_i - q_i)^2}$$

- Cosine similarity:

$$s(P, Q) = \frac{\sum p_i \cdot q_i}{\sqrt{\sum p_i^2 \times \sum q_i^2}}$$

Final Comment

- Jain and Dubes (Algorithms for Clustering Data):
 - 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.”

- 
- Assignment 2 has been released online

 - Start your project ASAP!
 - Form Groups
 - Project Proposal – 29th Aug (next Monday)
 - Project Consultation
 - Tomorrow 2-4pm
 - week 9
 - Project Presentation – week 10/week 11
 - Show your demo-system
 - Show your interesting results
 - Project Report