INTRODUCTORY APPLIED MACHINE LEARNING

Yan-Fu Kuo

Dept. of Biomechatronics Engineering

National Taiwan University

Today:

- k-means clustering
- Fuzzy c-means clustering
- Hierarchical clustering
- Cluster evaluation

Outline

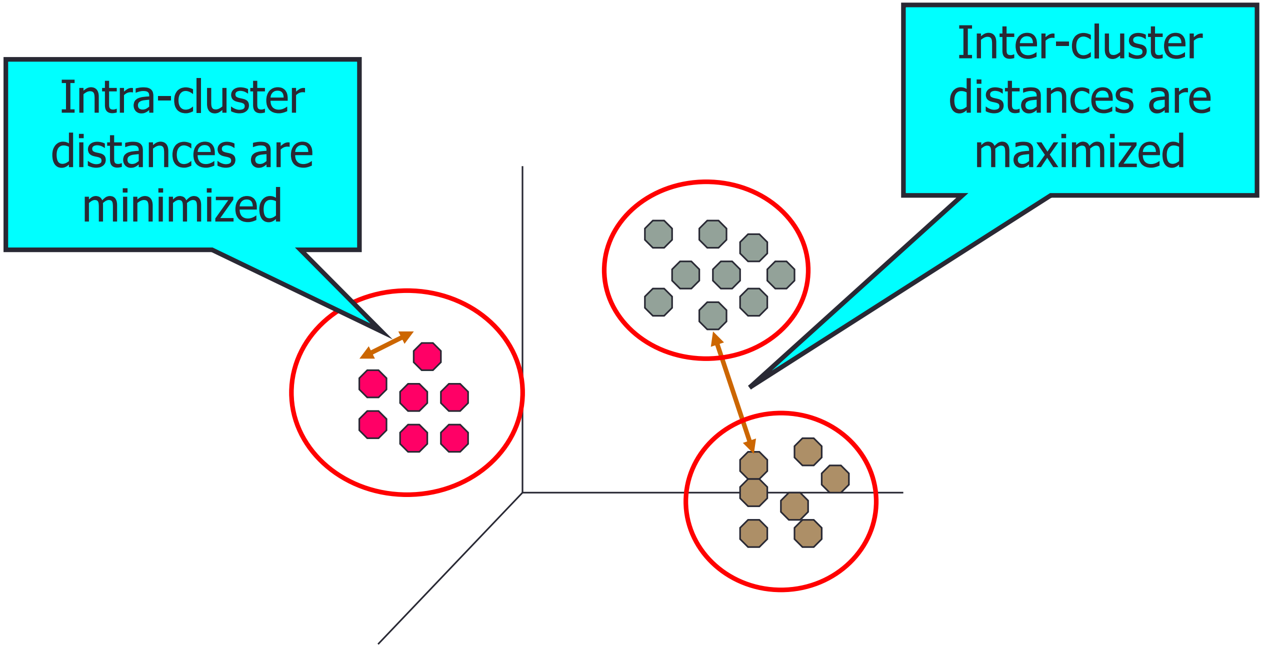
- Goal of the lecture
- Clustering overview
- Type of clustering
- k-means
- Fuzzy c-means
- Agglomerative hierarchical clustering
- Cluster evaluation

Goals

- After this, you should be able to:
  - Have the knowledge of cluster types
  - Understand the principles of clustering
  - Apply k-means and hierarchical clustering methods
  - Understand how to evaluate clustering results

What is Cluster Analysis?

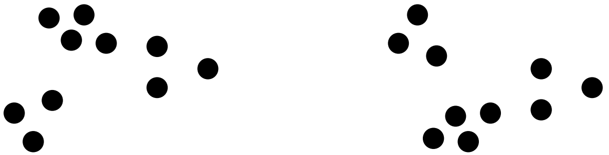
- Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups



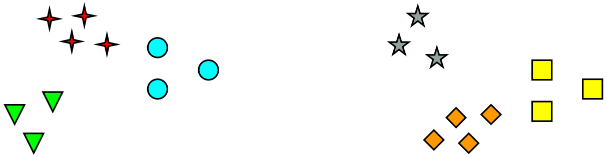
Intra-cluster distances are minimized

Inter-cluster distances are maximized

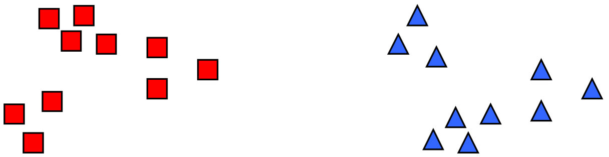
Notion of a Cluster Can Be Ambiguous



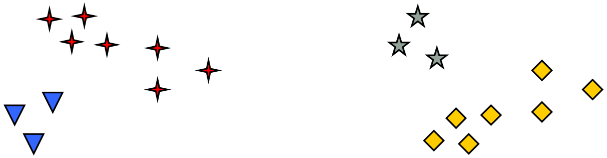
How many clusters?



Six Clusters



Two Clusters

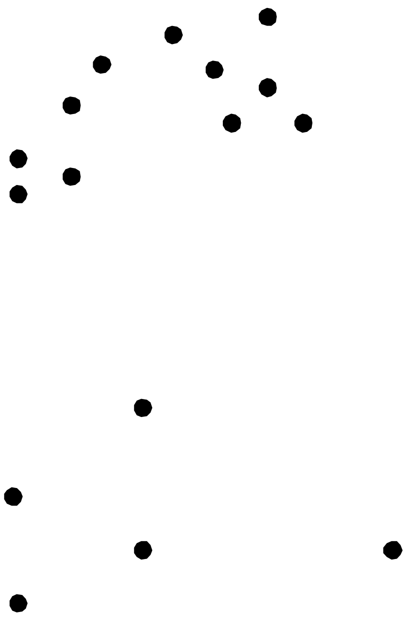


Four Clusters

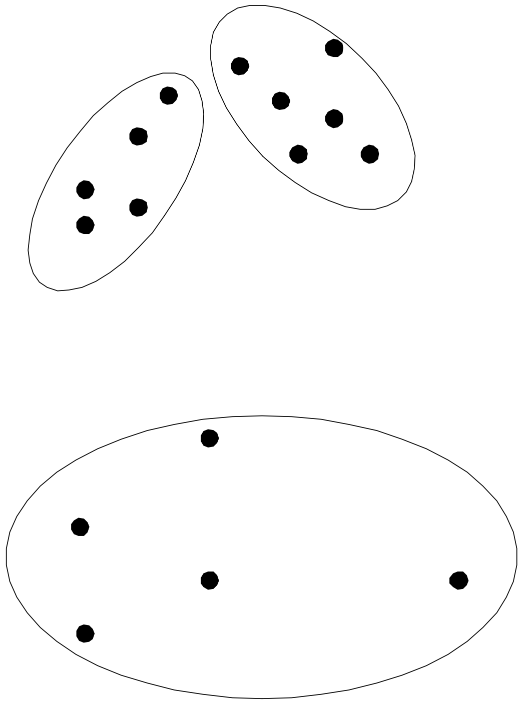
Types of Clustering

- Clustering is a method to partition data points into sets
- Two major clustering methods – partitional and hierarchical
    - 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

Partitional Clustering

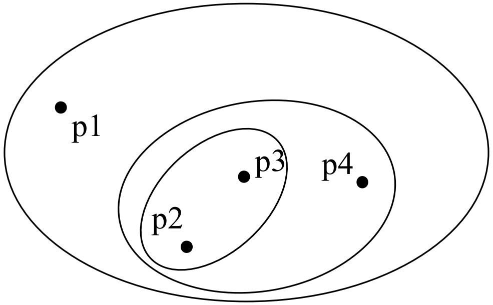


Original Points

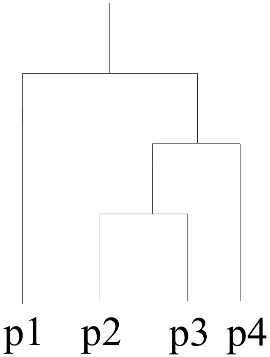


A Partitional Clustering

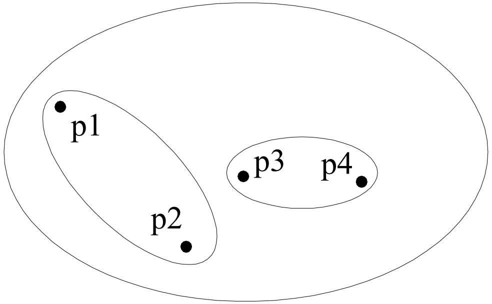
Hierarchical Clustering



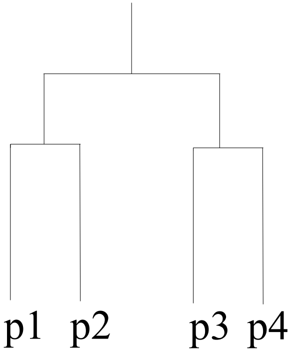
Traditional Hierarchical Clustering



Traditional Dendrogram

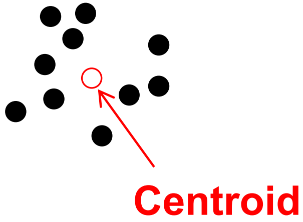


Non-traditional Hierarchical Clustering



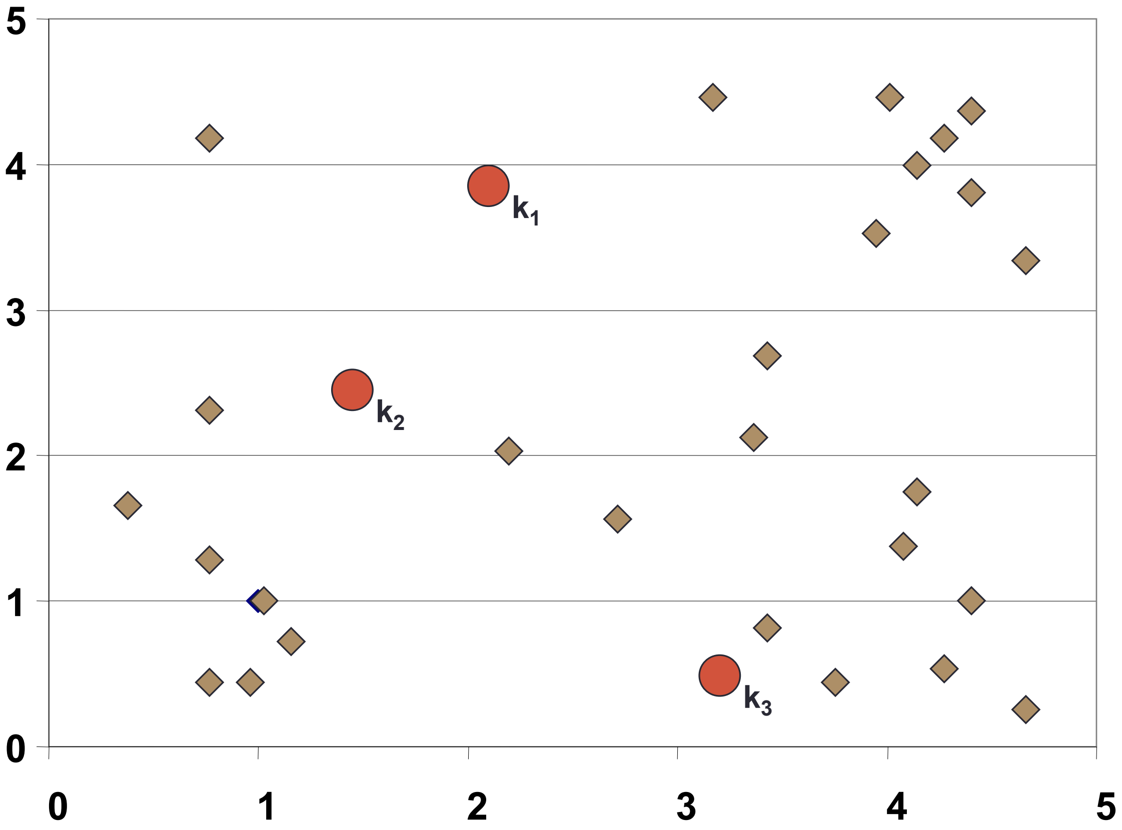
Non-traditional Dendrogram

k-means Clustering

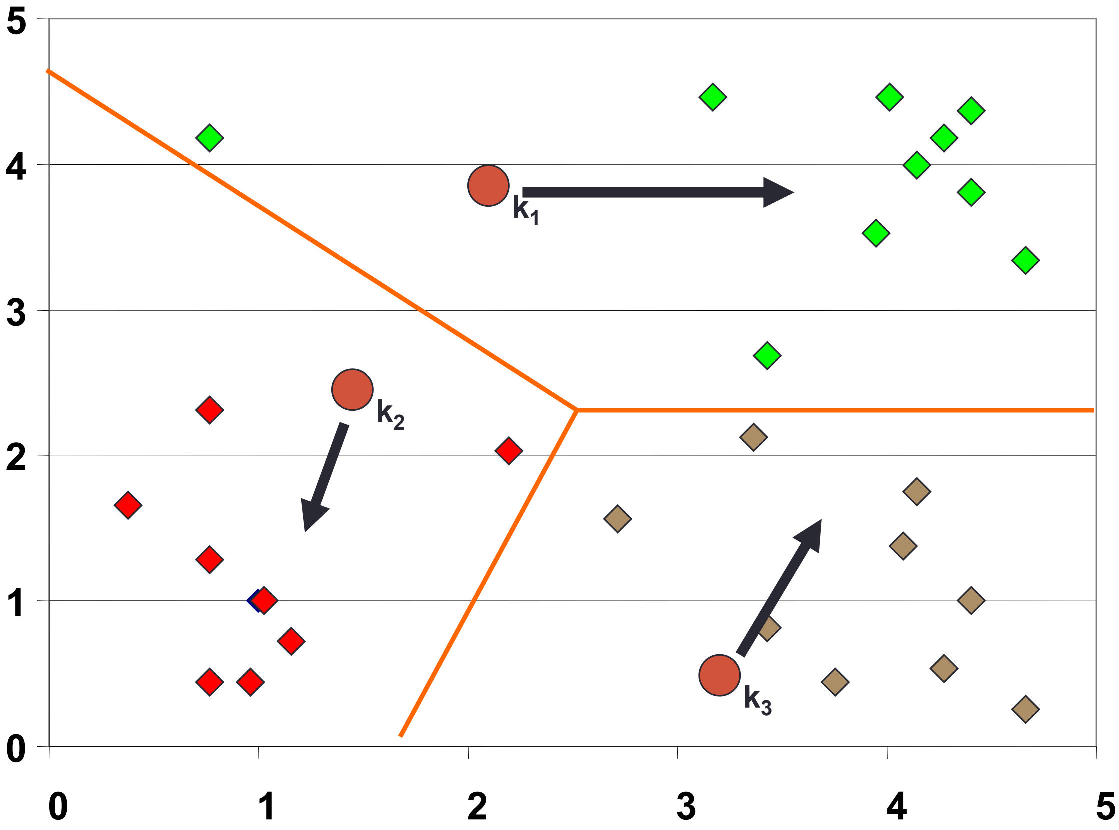


- Partitional clustering approach
- Each cluster is associated with a centroid (center point)
- The objective is to minimize within-group variance
- Determine the number of clusters K
- Iteratively update the cluster for data points:
  1. Select K points randomly to represent initial group centroids
  2. Assign each object to the group that has the closest centroid
  3. When all objects have been assigned, recalculate the centroid positions of the K groups
  4. Repeat steps 2 and 3 until the centroids no longer move

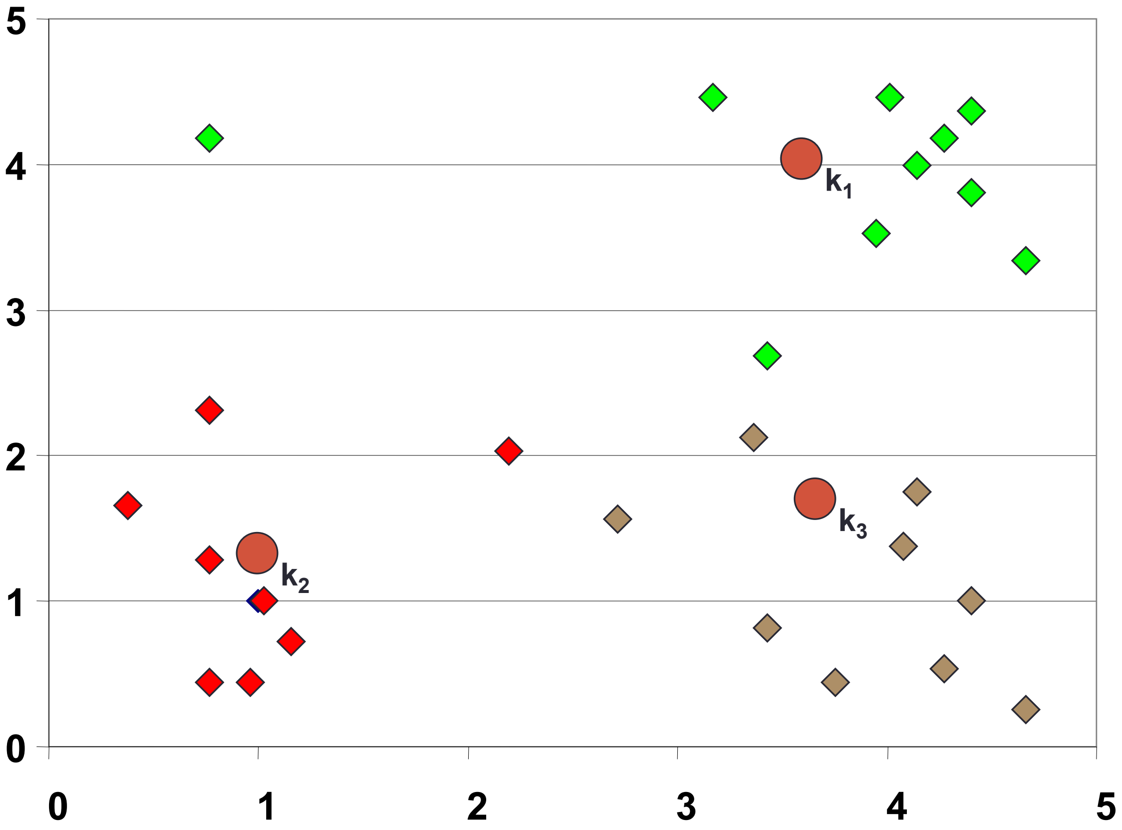
Example: Step 1



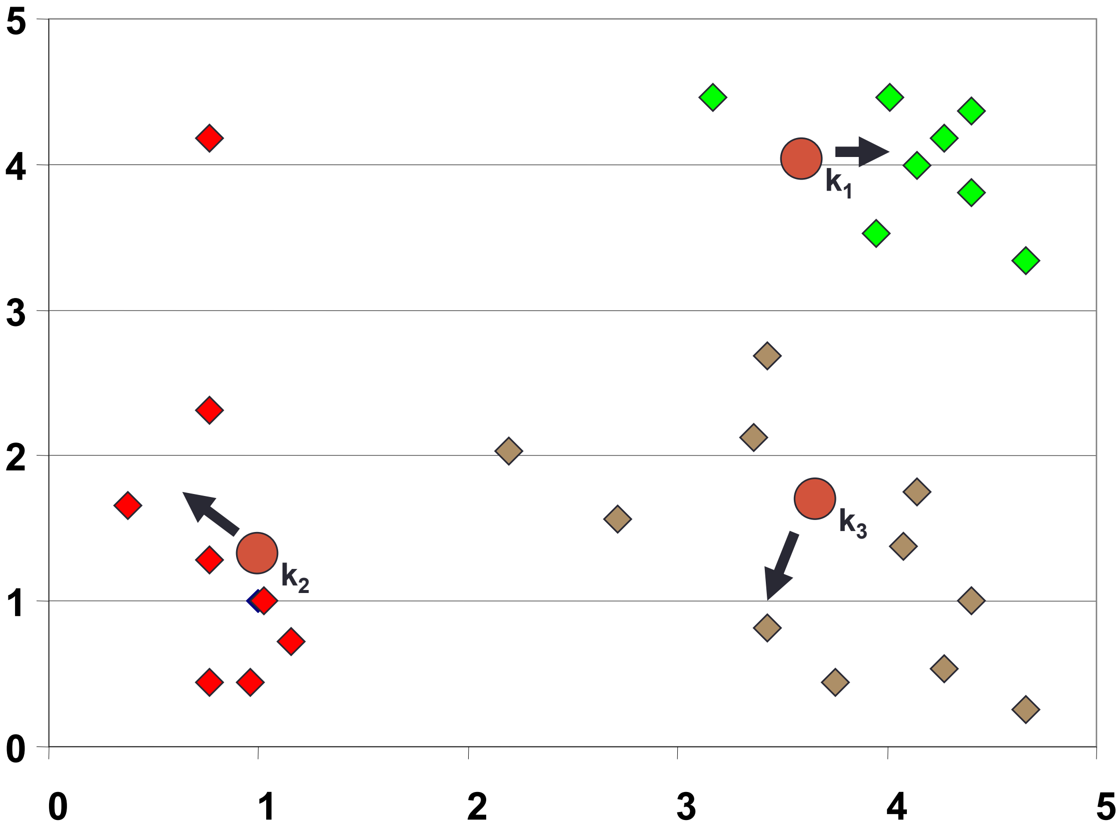
Example: Step 2



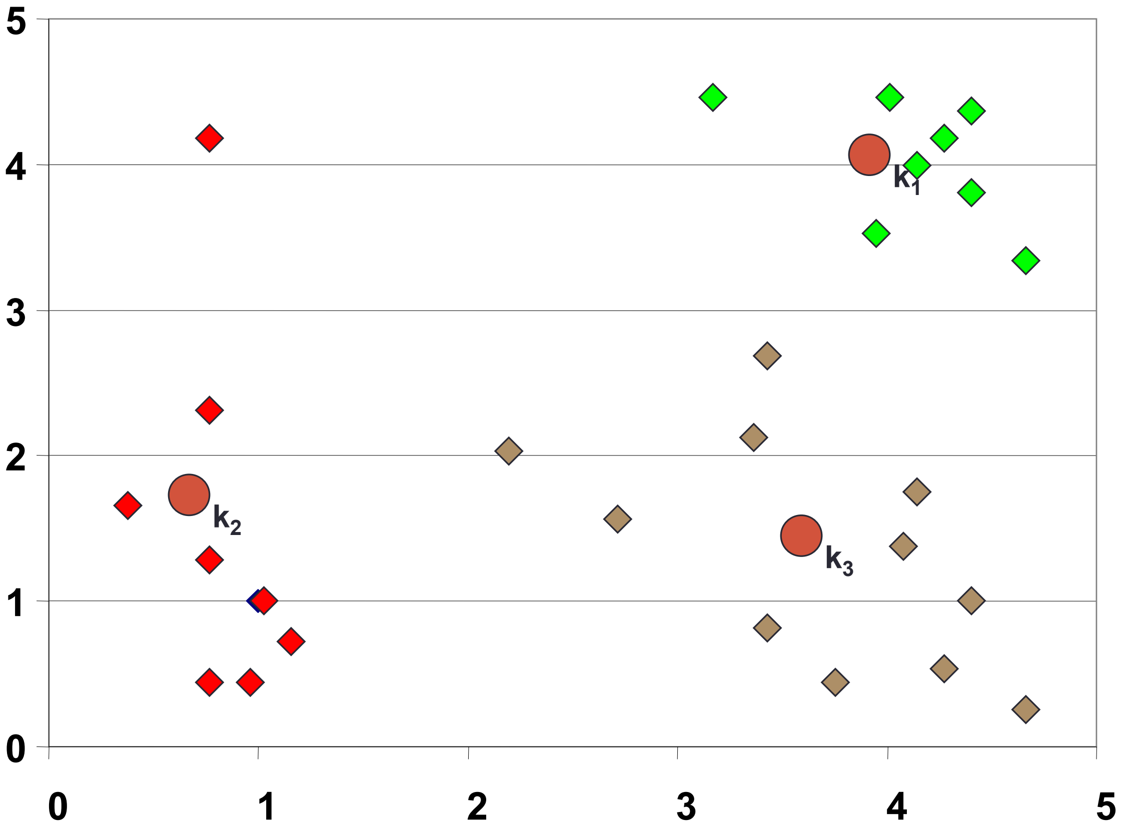
Example: Step 3



Example: Step 4



Example: Step 4



Scatter of the Data Set

- Let $\boldsymbol{x}_i \in \Re^P, i = 1 \ldots N$, denote points of a data set
- Note that there is NO label y
- Consider total scatter of the data set:

$$t = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} d(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \Re$$

where $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is a distance metric (e.g., Euclidean distance)

Within-cluster and between-cluster Scatter

- Suppose the data points are partitioned into K subsets $\{S_1, \dots, S_K\}$, the “total scatter” is:

$$t = \frac{1}{2} \sum_{m=1}^{K} \sum_{\boldsymbol{x}_i \in S_m} \left(\sum_{\boldsymbol{x}_j \in S_m} d(\boldsymbol{x}_i, \boldsymbol{x}_j) + \sum_{\boldsymbol{x}_j \notin S_m} d(\boldsymbol{x}_i, \boldsymbol{x}_j) \right)$$

- Let

$$w \equiv \frac{1}{2} \sum_{m=1}^{K} \sum_{\boldsymbol{x}_i \in S_m} \sum_{\boldsymbol{x}_j \in S_m} d(\boldsymbol{x}_i, \boldsymbol{x}_j)\, ,\, b \equiv \frac{1}{2} \sum_{m=1}^{K} \sum_{\boldsymbol{x}_i \in S_m} \sum_{\boldsymbol{x}_j \notin S_m} d(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

be within-cluster scatter and between-cluster scatter, respectively

Within-cluster Scatter

- Suppose Euclidean distance is applied
- Within-cluster point scatter can be formulated as

$$w = \frac{1}{2}\sum_{m=1}^{K}\sum_{\boldsymbol{x}_i \in S_m}\sum_{\boldsymbol{x}_j \in S_m} d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{1}{2}\sum_{m=1}^{K}\sum_{\boldsymbol{x}_i \in S_m}\sum_{\boldsymbol{x}_j \in S_m} \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2$$

$$= \sum_{m=1}^{K} N_m \sum_{\boldsymbol{x}_i \in S_m} \|\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m\|^2 \in \Re$$

where $N_m \in \aleph$ is the number of observations in the mth cluster, and $\overline{\boldsymbol{x}}_m \in \Re^P$ is the mean of the mth cluster

- The objective is to minimize the within-cluster scatter

Cluster Assignment

- The mean of a cluster $\overline{\boldsymbol{x}}_m$ is defined as:

$$\overline{\boldsymbol{x}}_m = \frac{\sum_{\boldsymbol{x}_i \in S_m} \boldsymbol{x}_i}{N_m}, \quad m = 1 \dots K$$

- Note that the data points in set S_m is defined and changed in each iteration
- For a current set of cluster means, assign the cluster $c(\boldsymbol{x}_i)$ for each $\boldsymbol{x}_i$ in every iteration as:

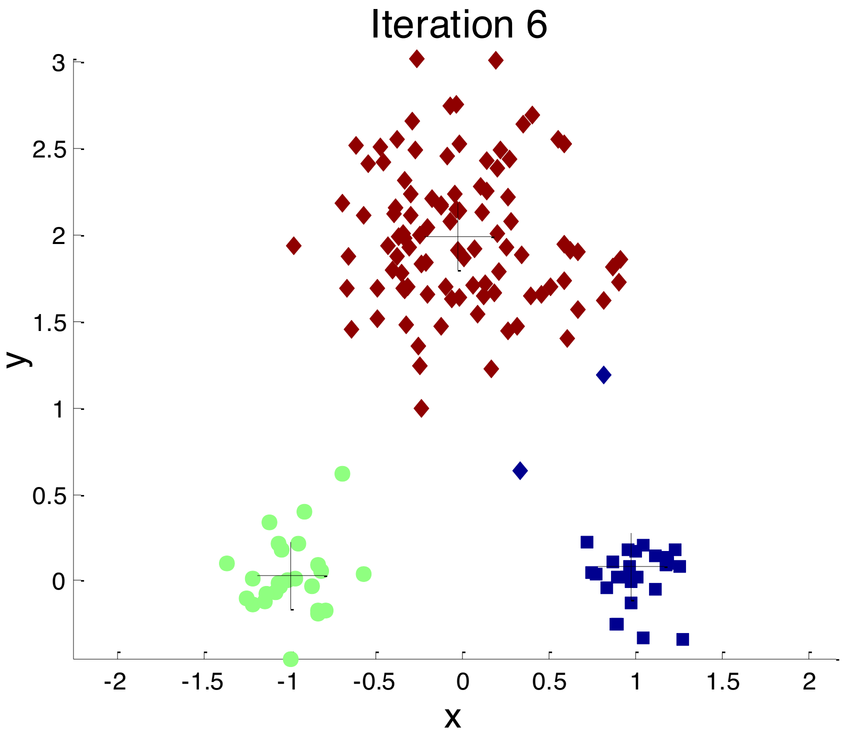
$$c(\boldsymbol{x}_i) = \underset{1 \le m \le K}{\operatorname{argmin}} \|\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m\|^2, \;\; i = 1 \dots N$$

- Iterate above two steps until convergence

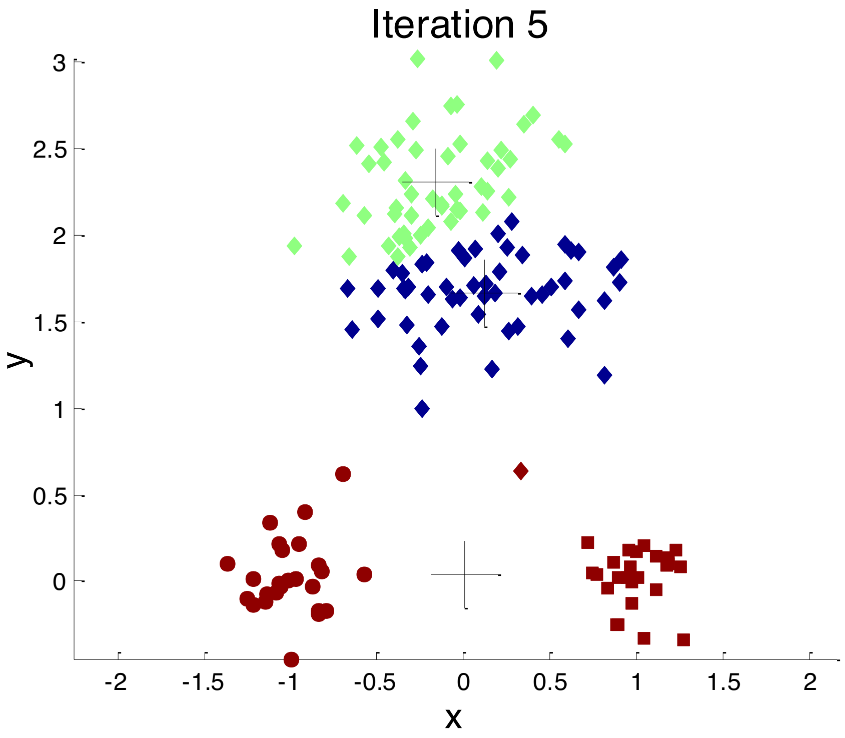
k-means Clustering – Details

- Initial centroids are often chosen randomly
- 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”

Importance of Choosing Initial Centroids



Importance of Choosing Initial Centroids



Solutions to Initial Centroids Problem

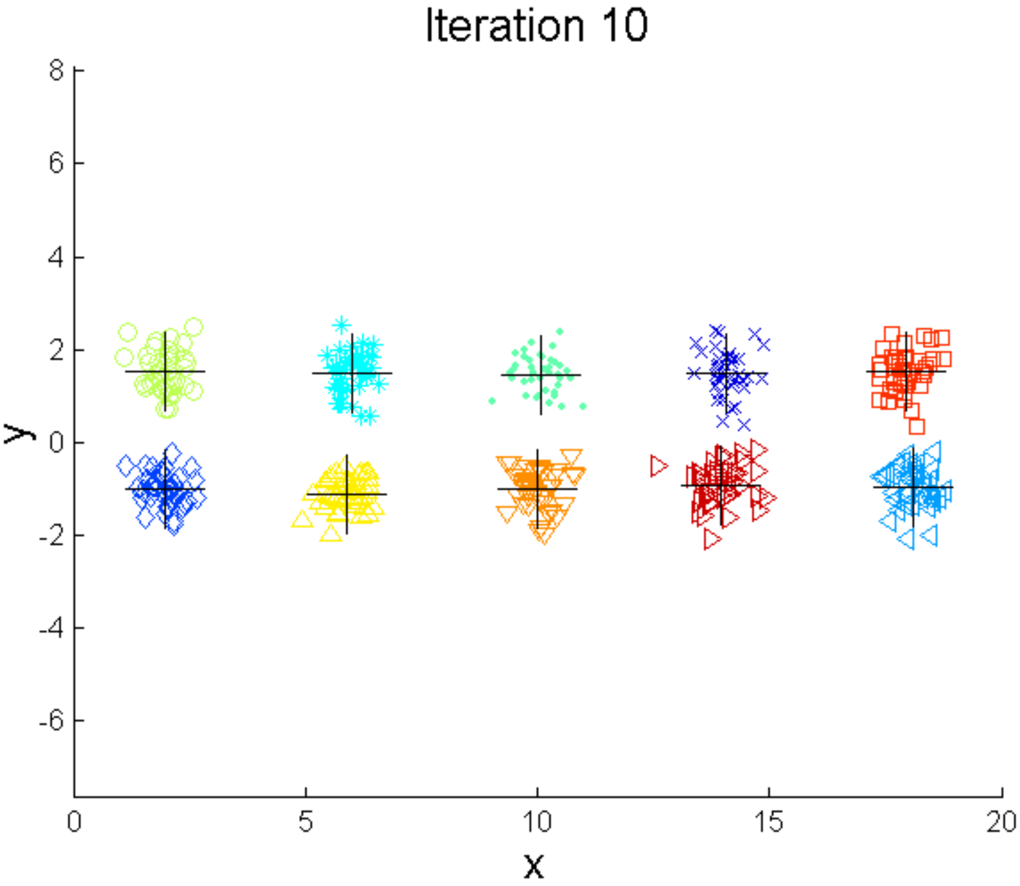
- Multiple runs – does not always work
- Sample and use hierarchical clustering to determine initial centroids
- Bisecting k-means

Bisecting k-means

- Iteratively split the cluster until the number of clustering reaches K
- Procedure:

```
1. Initialize the complete data set as a cluster ($k = 1$)
2. While ($k < K$)
   {
       Bisect a selected cluster using k-means
       $k = k + 1$
   }
```

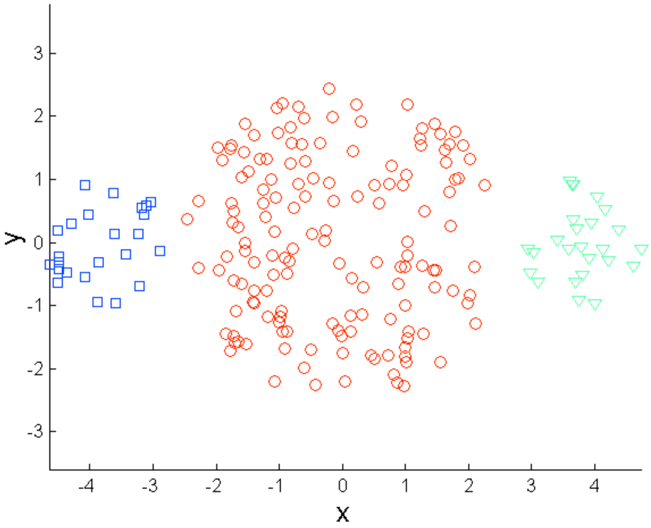
Bisecting K-means Example



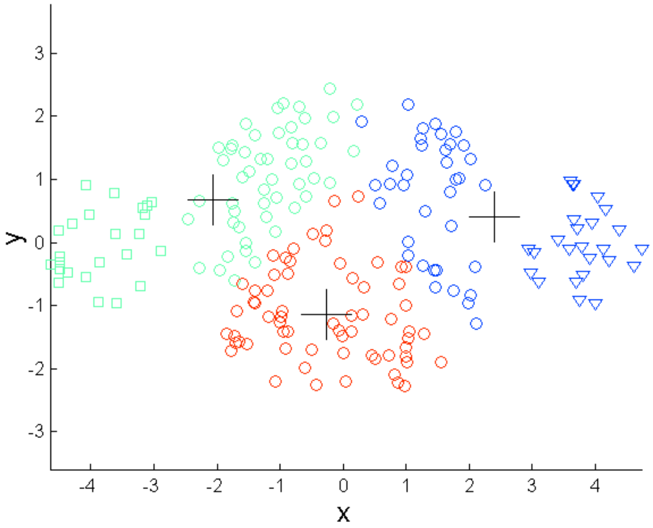
Limitations of k-means

- k-means has problems when clusters are of different
  - Sizes
  - Densities
  - Non-globular shapes
- k-means has problems when the data contains outliers

Limitations of K-means: Differing Sizes

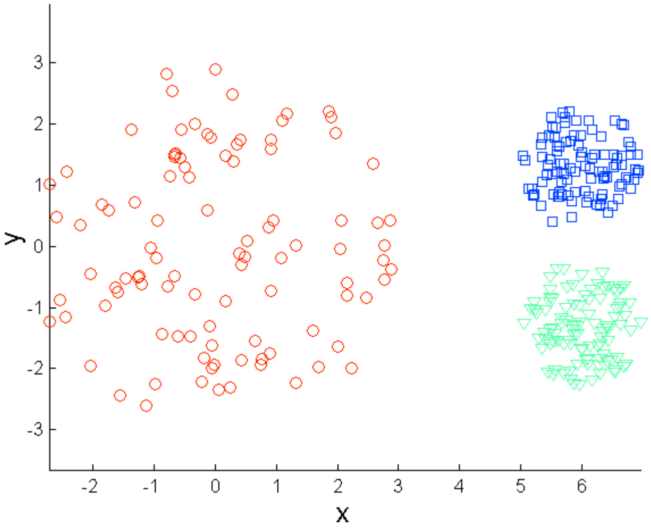


Original points

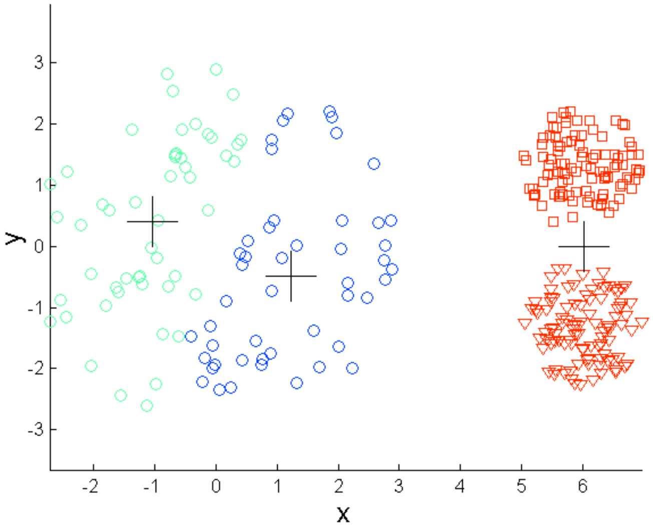


K-means (3 clusters)

Limitations of K-means: Differing Density

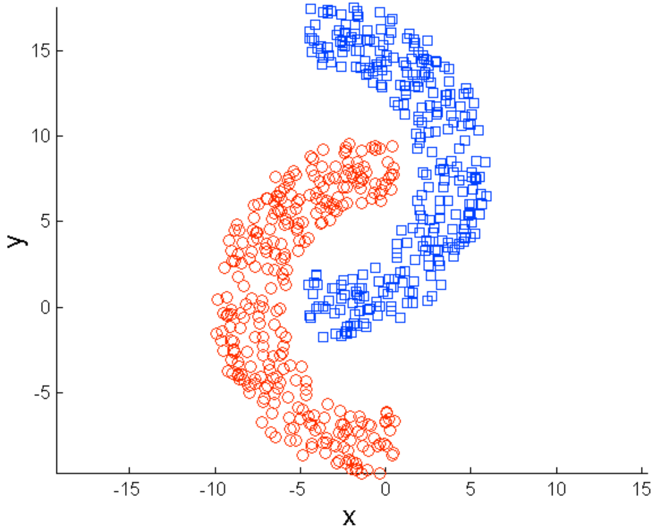


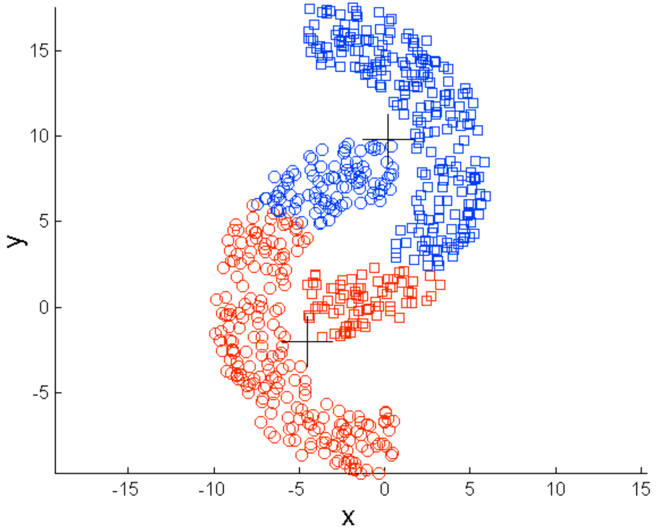
Original points



K-means (3 clusters)

Limitations of K-means: Non-globular Shapes

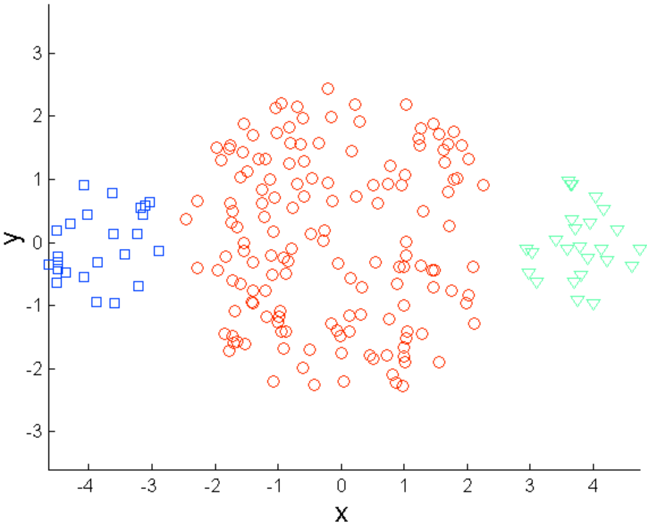




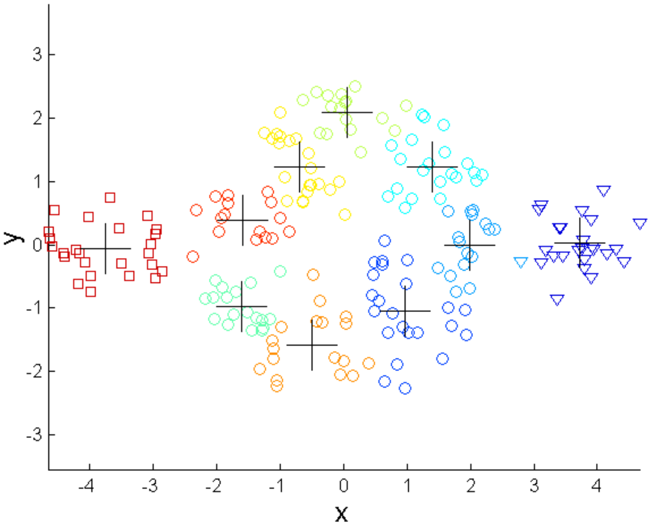
Original points

K-means (2 clusters)

Overcoming K-means Limitations



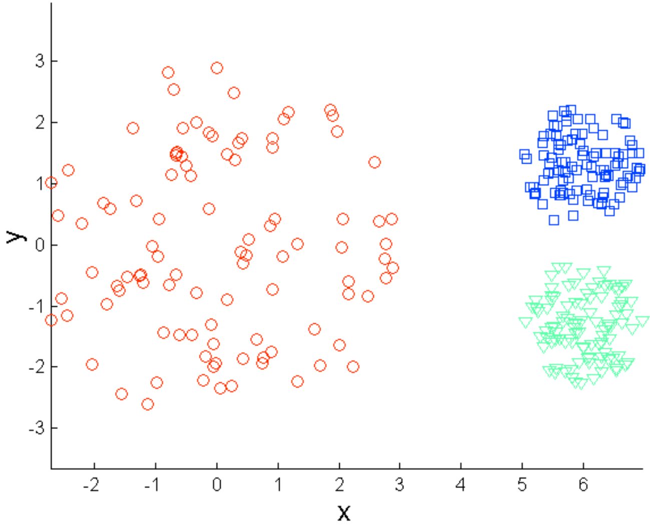
Original points



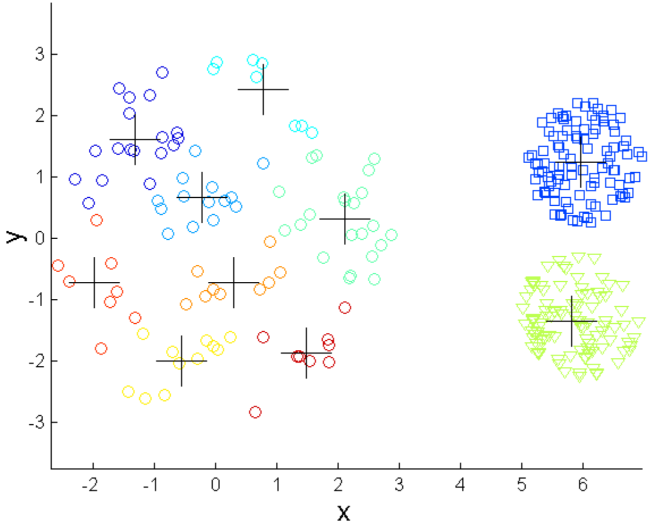
K-means clusters

One solution is to use many clusters, and put together the clusters

Overcoming K-means Limitations

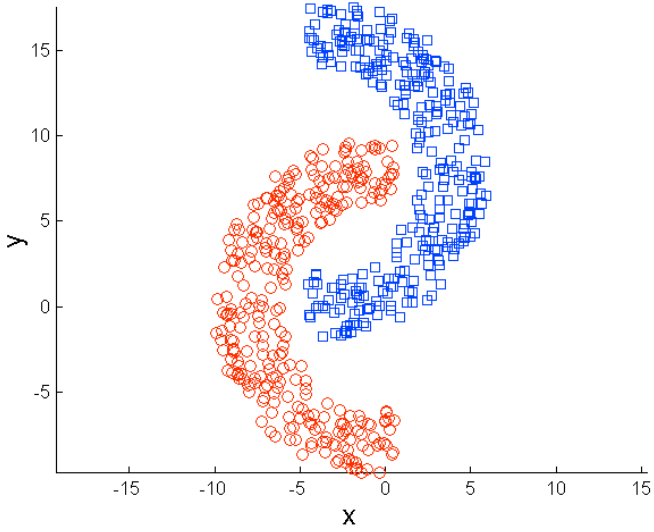


Original points

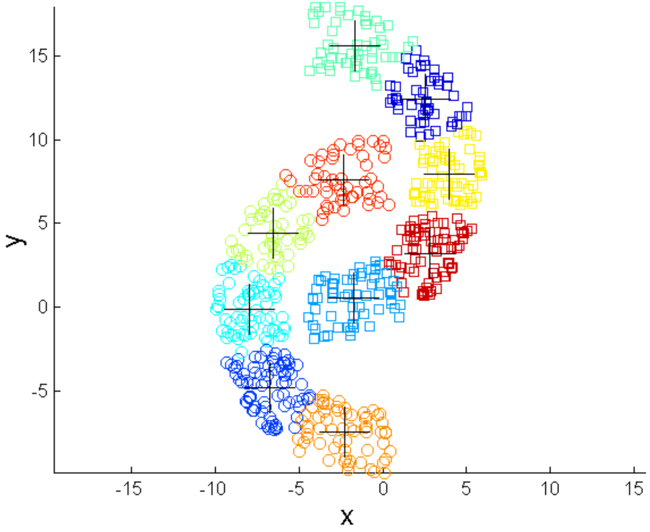


K-means clusters

Overcoming K-means Limitations



Original points

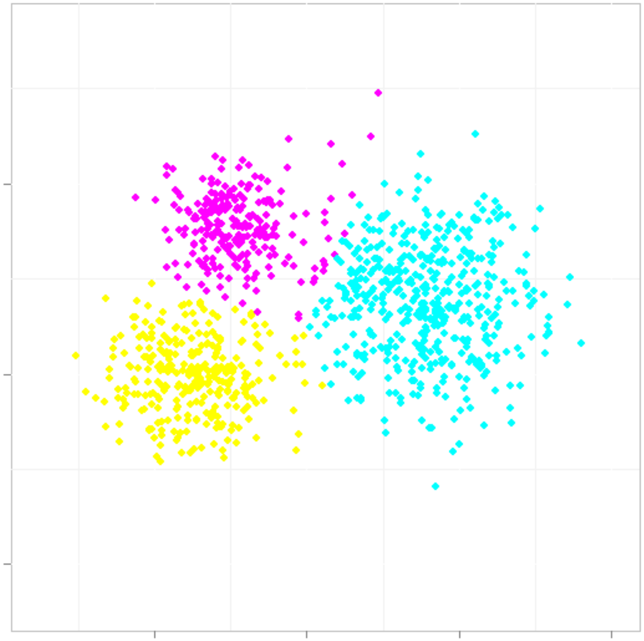


K-means clusters

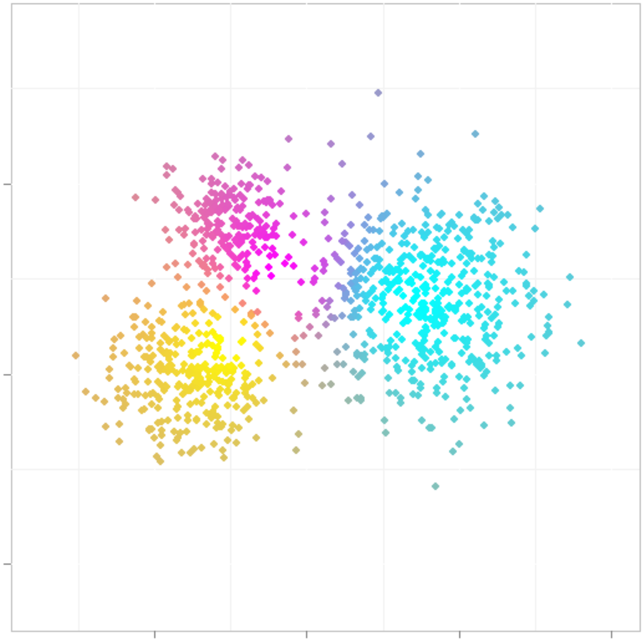
Fuzzy c-means

- One data point can belong to two or more clusters, i.e., overlapping clustering

K-means



Fuzzy c-means



Fuzzy c-means Cost Function

- Suppose there exists K clusters and N data points
- The cost function of fuzzy c-means:

$$s = \sum_{m=1}^{K} \sum_{i=1}^{N} \delta_{im}^{q} \|\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m\|^2 \in \Re,$$

where $\Re \ni \delta_{im} \in [0,1]$ is the weight that represents the degree of membership of a data point $\boldsymbol{x}_i$ to cluster m, and $1 < q \in \Re$ is a hyperparameter that determines the influence of the weights

- Note that $\sum_{m=1}^{K} \delta_{im} = 1$

Centroid of Fuzzy c-means

- For the mth cluster, the corresponding centroid $\overline{\boldsymbol{x}}_m$ is defined as:

$$\overline{\boldsymbol{x}}_m = \frac{\sum_{i=1}^{N} \delta_{im}^{q} \boldsymbol{x}_i}{\sum_{i=1}^{N} \delta_{im}^{q}},$$

which is weighted by its membership degree δ_{im}^{q}

Optimization Objective

- The objective is to update the
  1. weights
  2. centroids

  so that the cost function is minimized

Optimization Formulation

- The cost function and constraint:

$$\text{Minimize} \sum_{m=1}^{K}\sum_{i=1}^{N}\delta_{im}^{q}||\boldsymbol{x}_i-\overline{\boldsymbol{x}}_m||^2 \quad \text{subject to} \quad \sum_{m=1}^{K}\delta_{im}=1$$

- Lagrangian:

$$L=\sum_{m=1}^{K}\sum_{i=1}^{N}\delta_{im}^{q}||\boldsymbol{x}_i-\overline{\boldsymbol{x}}_m||^2-\lambda\left(\sum_{m=1}^{K}\delta_{im}-1\right)$$

How to Update the Weights?

- Setting the partial derivatives of the Lagrangian to zero:

$$\frac{\partial L}{\partial \lambda} = \sum_{m=1}^{K} \delta_{im} - 1 = 0 \quad \text{.........................(1)}$$

$$\frac{\partial L}{\partial \delta_{im}} = q\delta_{im}^{q-1} ||\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m||^2 - \lambda = 0 \quad \text{................(2)}$$

- From (2), we obtain:

$$\delta_{im}^{q-1} = \frac{\lambda}{q||\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m||^2} \quad \Rightarrow \quad \delta_{im} = (\frac{\lambda}{q||\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m||^2})^{\frac{1}{q-1}} \quad \text{....(3)}$$

How to Update the Weights? (Cont’d)

- From Eq. (1), we obtain

$$1 = \sum_{m=1}^{K} \delta_{im} = \sum_{m=1}^{K} \left(\frac{\lambda}{q\|\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m\|^2}\right)^{\frac{1}{q-1}}$$

$$= \left(\frac{\lambda}{q}\right)^{\frac{1}{q-1}} \sum_{m=1}^{K} \left(\frac{1}{\|\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m\|^2}\right)^{\frac{1}{q-1}}$$

$$\Rightarrow \left(\frac{\lambda}{q}\right)^{\frac{1}{q-1}} = \frac{1}{\sum_{m=1}^{K} \left(\frac{1}{\|\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m\|^2}\right)^{\frac{1}{q-1}}}$$

How to Update the Weights? (Cont'd)

- From (3), we obtain:

$$\delta_{im} = \left(\frac{\lambda}{q\|\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m\|^2}\right)^{\frac{1}{q-1}} = \left(\frac{\lambda}{q}\right)^{\frac{1}{q-1}} \left(\frac{1}{\|\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m\|^2}\right)^{\frac{1}{q-1}}$$

$$= \frac{\left(\frac{1}{\|\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m\|^2}\right)^{\frac{1}{(q-1)}}}{\sum_{m=1}^{K}\left(\frac{1}{\|\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m\|^2}\right)^{\frac{1}{(q-1)}}}$$

- The weight is updated by the above equation

How to Update the Centroid?

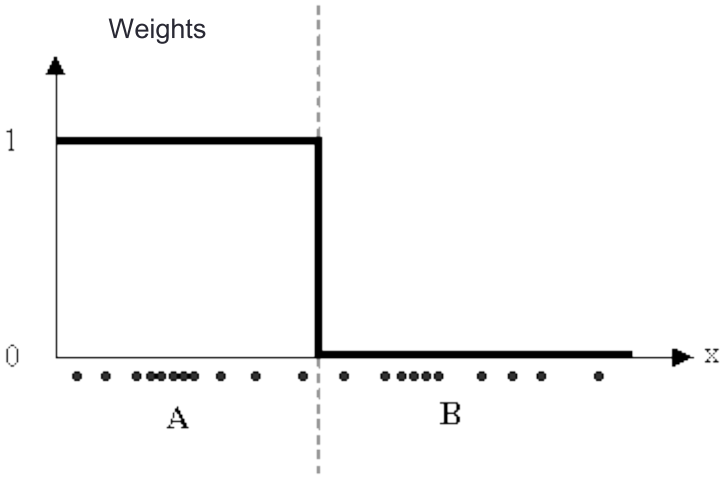
- Setting the derivative of the Lagrangian with respect to $\bar{\boldsymbol{x}}_m$:

$$\frac{\partial L}{\partial \bar{\boldsymbol{x}}_m} = -2\sum_{i=1}^{N} \delta_{im}^{q}(\boldsymbol{x}_i - \bar{\boldsymbol{x}}_m) = 0$$

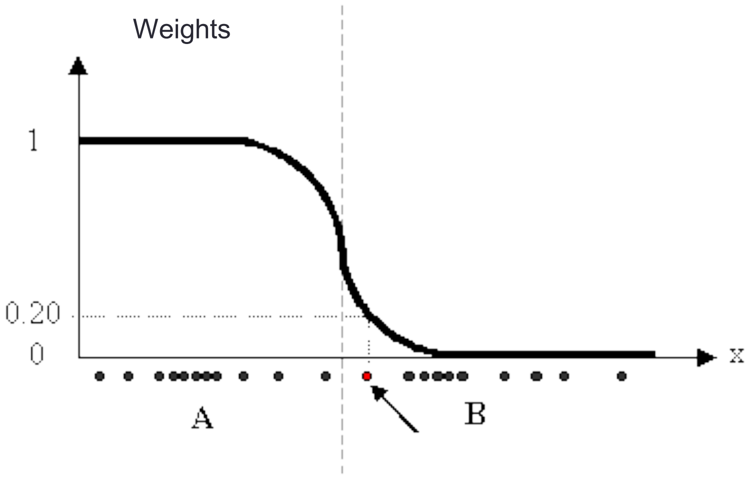
$$\Rightarrow \quad \bar{\boldsymbol{x}}_m = \frac{\sum_{i=1}^{N} \delta_{im}^{q} \boldsymbol{x}_i}{\sum_{i=1}^{N} \delta_{im}^{q}}$$

Weights of Fuzzy c-means and k-means

Weights of k-means



Weights of fuzzy c-means



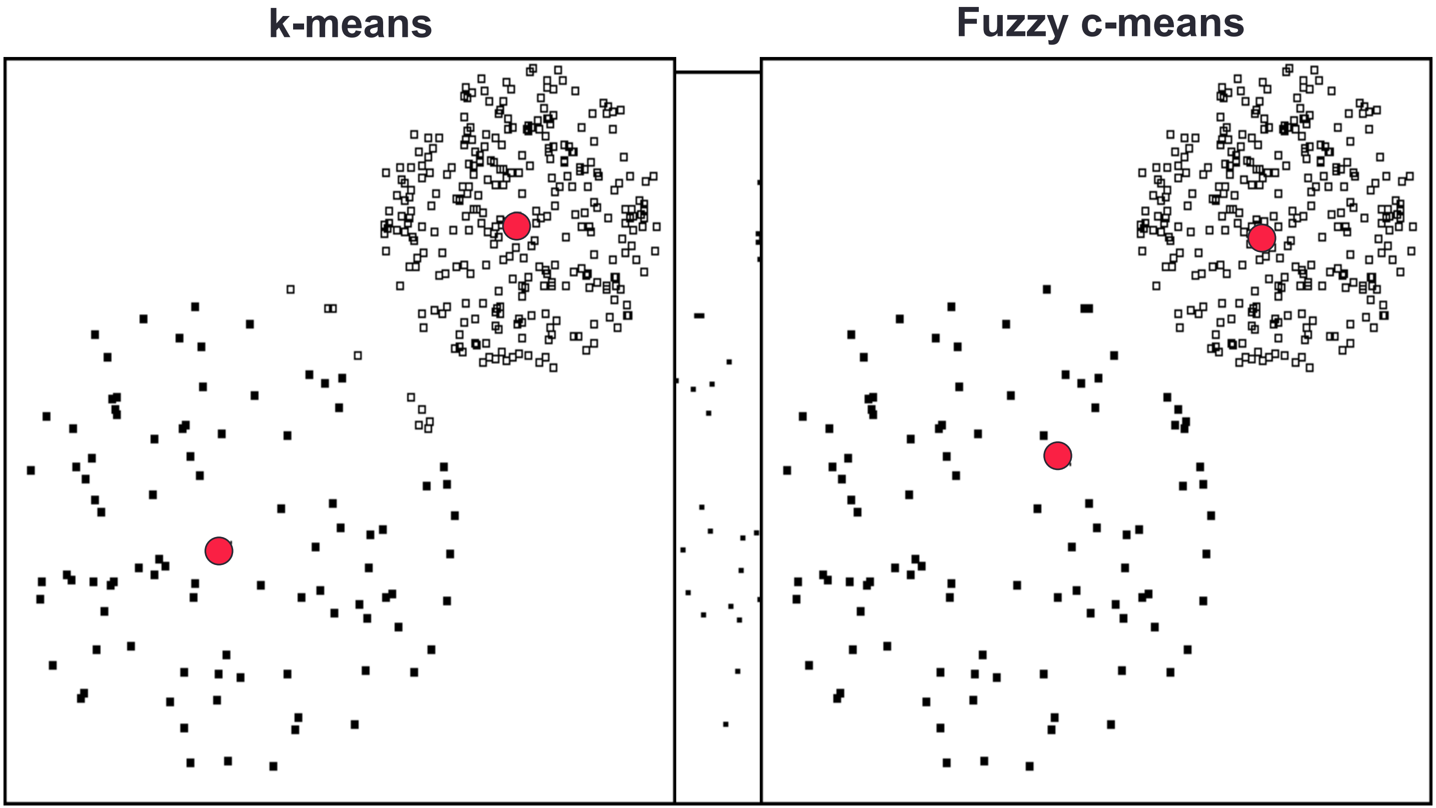
Picture from: Matteo Matteucci

Fuzzy c-means Procedure

- Procedure:

```
1. Select an initial fuzzy pseudo-weight
2. while(centroids hasn’t converge)
   {
       a) Compute the centroid of each cluster using the
          fuzzy weight
       b) Update the fuzzy weight
   }
```

Fuzzy c-means vs. k-means

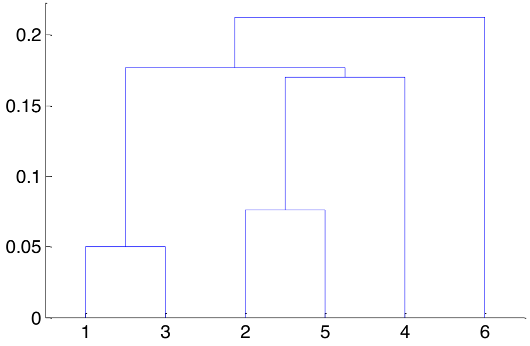


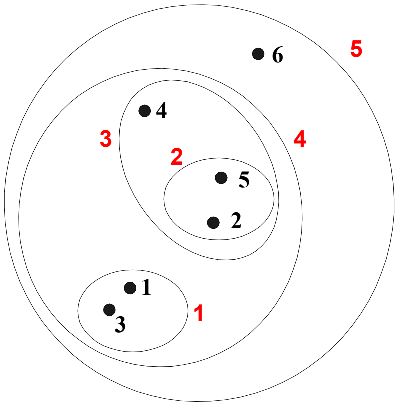
Fuzzy c-means Summary

- Allows a data point to be in multiple clusters
- Still need to define K, the number of clusters
- Need to define one more parameter q
- Clusters are sensitive to initial assignment of centroids

Hierarchical Clustering

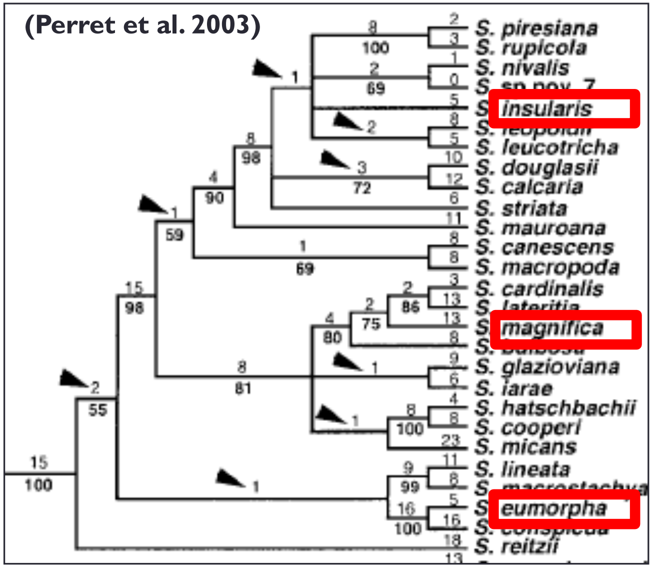
- Produces a set of nested clusters organized as a hierarchical tree
- Can be visualized as a dendrogram – a tree like diagram that records the sequences of merges or splits





Strengths of Hierarchical Clustering

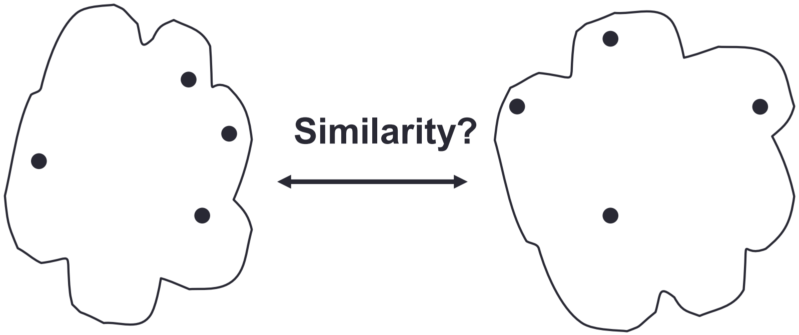
- Do not have to assume any particular number of clusters
- Any desired number of clusters can be obtained by ‘cutting’ the dendogram at a proper level
- They may correspond to meaningful taxonomies


(Perret et al. 2003)

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)
- How to measure cluster similarity?

How to Define Inter-cluster Similarity?



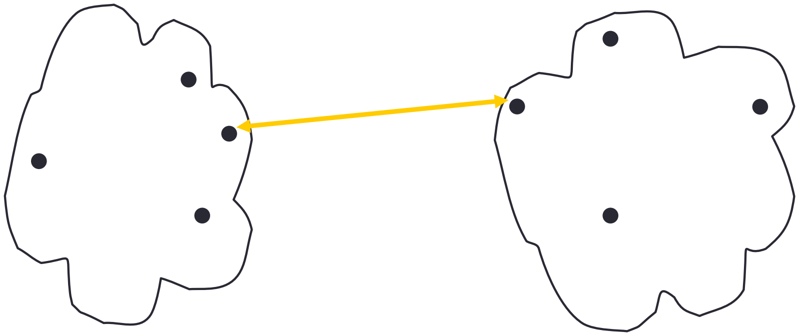
Minimum (or single-linkage)

Maximum (or complete-linkage)

Group average

Distance between centroids

Minimum (or Single-linkage)



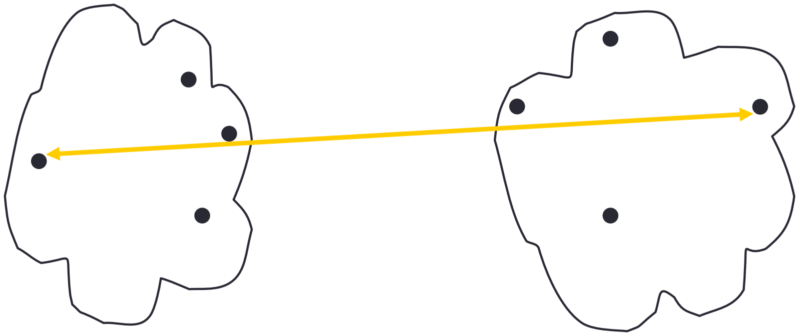
Minimum (or single-linkage)

Maximum (or complete-linkage)

Group average

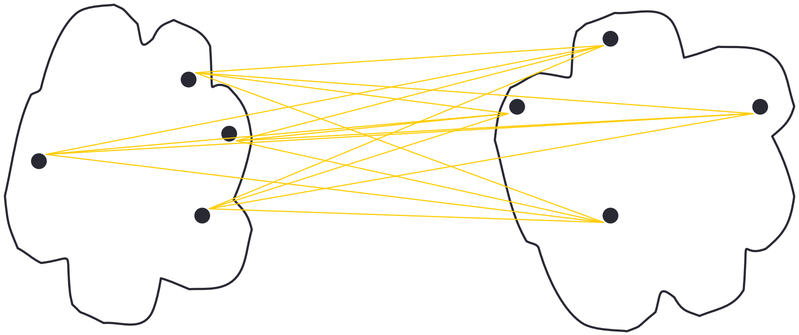
Distance between centroids

Maximum (or Complete-linkage)



- Minimum (or single-linkage)
- Maximum (or complete-linkage)
- Group average
- Distance between centroids

Group Average



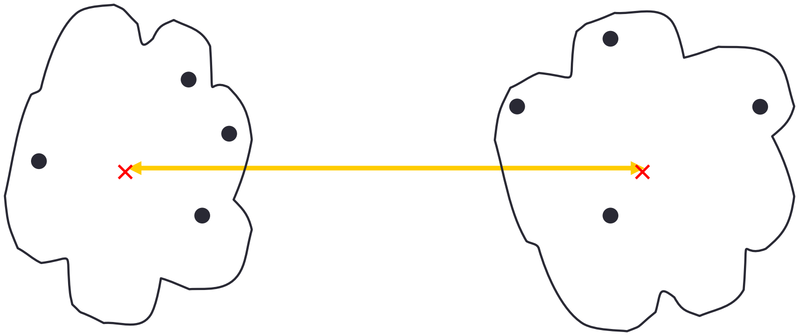
Minimum (or single-linkage)

Maximum (or complete-linkage)

Group average

Distance between centroids

Centroid Distance



Minimum (or single-linkage)

Maximum (or complete-linkage)

Group average

Distance between centroids

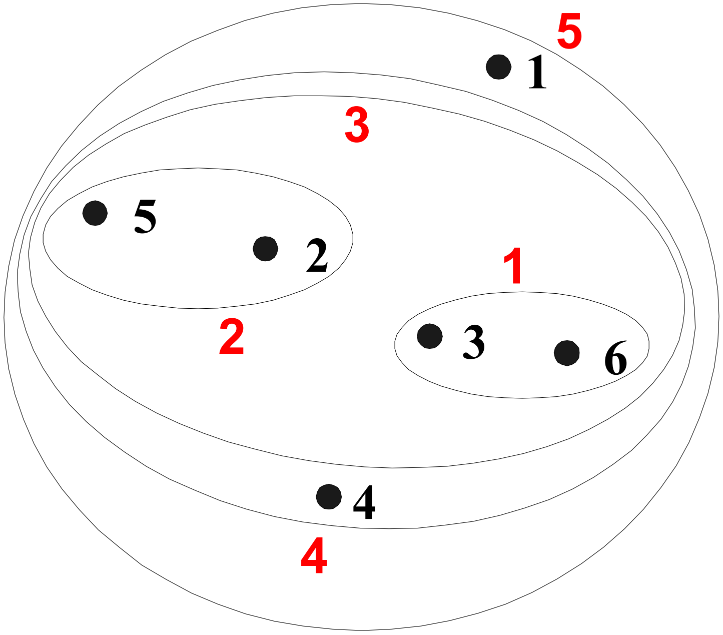
Agglomerative Clustering Algorithm

- Procedure:

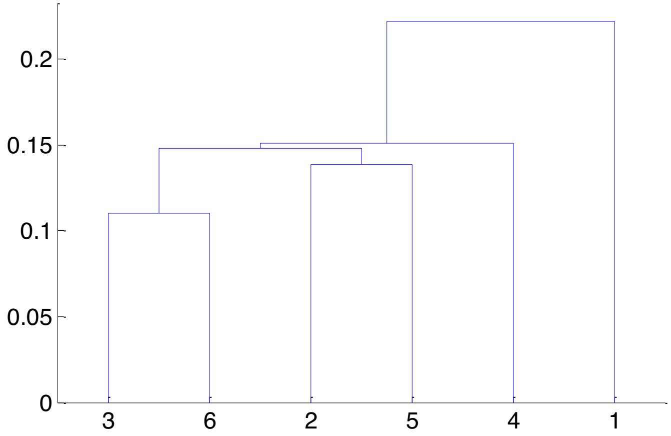
```
1. Let each data point be a cluster
2. Calculate the proximity matrix
3. while ($k > 1$)
   {
       Merge the two closest clusters
       Update the proximity matrix
   }
```

Clustering Using “Minimum” Similarity

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



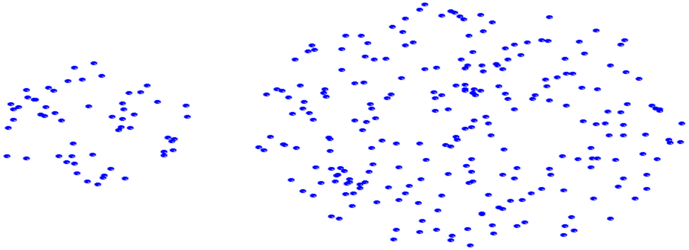
Nested clusters



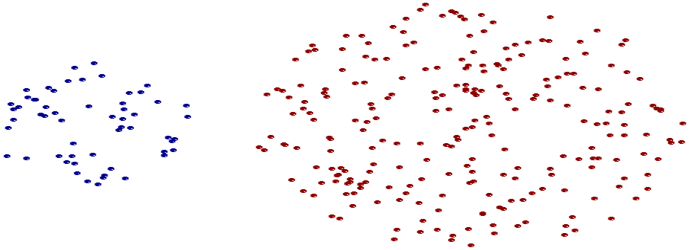
Dendrogram

Strength of Minimum

- Can handle non-elliptical shapes



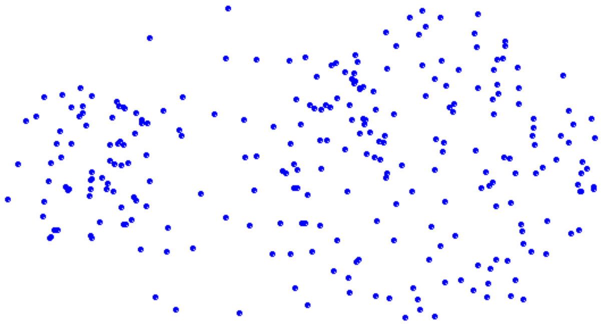
Original points



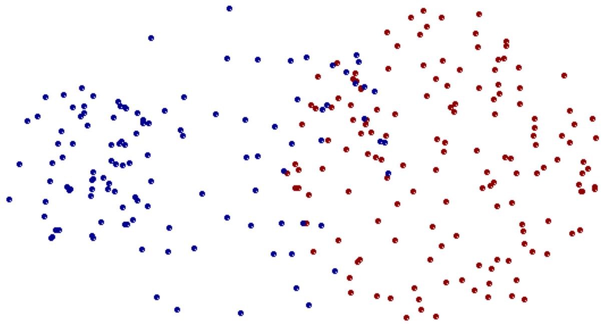
Two clusters

Limitations of Minimum

- Sensitive to noise and outliers



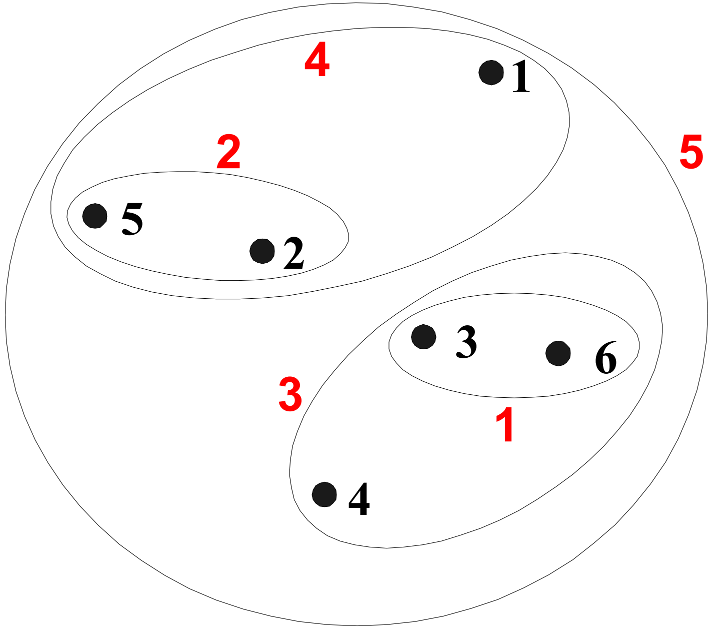
Original points



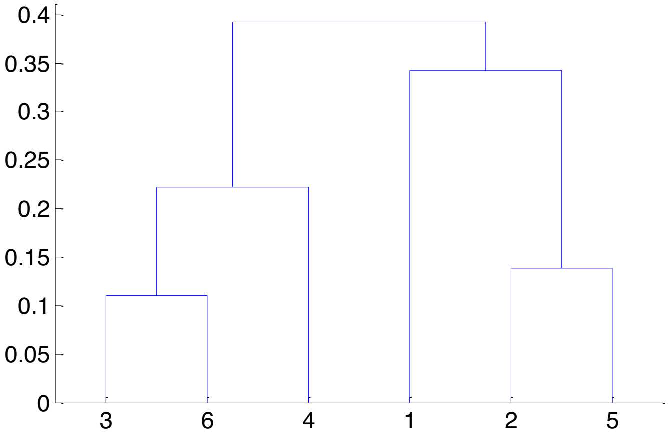
Two clusters

Clustering Using "Maximum" Similarity

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



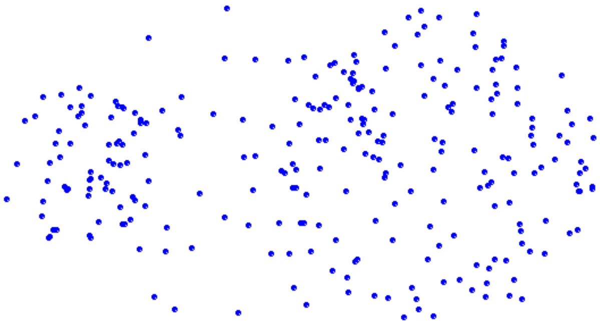
Nested clusters



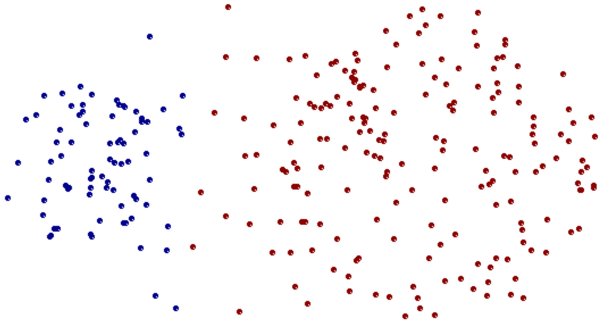
Dendrogram

Strength of Maximum

- Less susceptible to noise and outliers



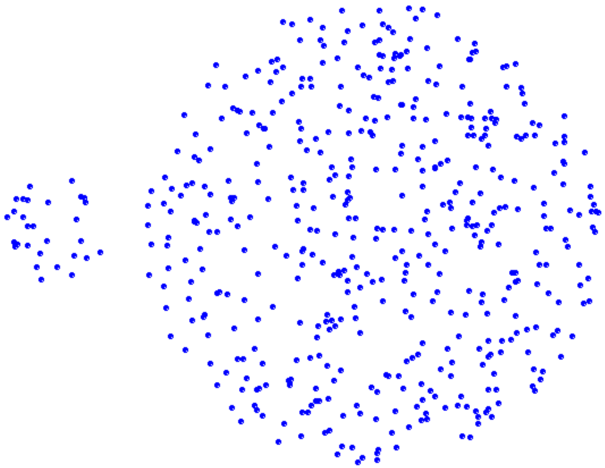
Original points



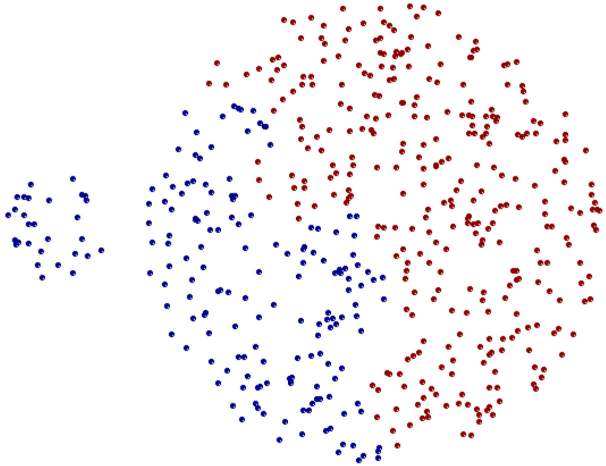
Two clusters

Limitations of Maximum

- Tends to break large clusters
- Biased towards globular clusters



Original points

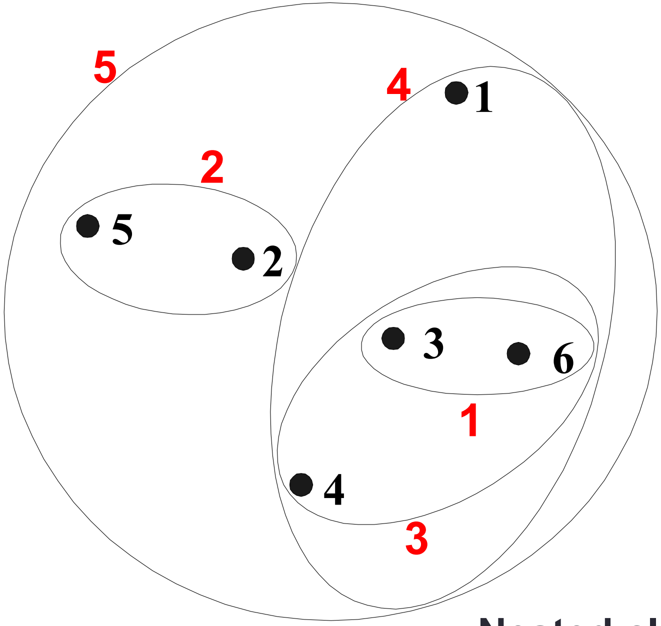


Two clusters

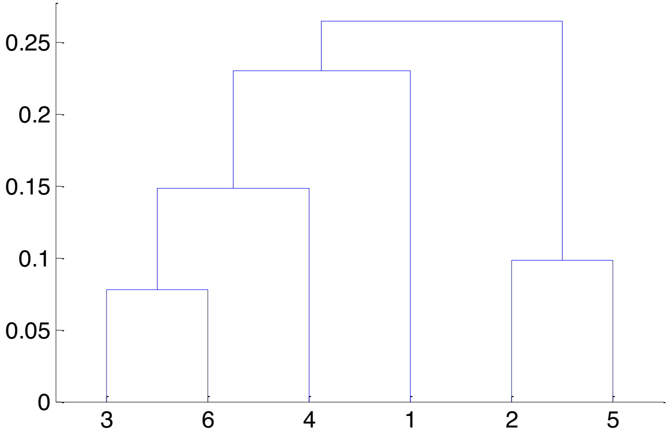
Cluster Similarity: Group Average

- Proximity of two clusters S_i and S_j is the average of pairwise proximity between points in the two clusters

$$proximity(S_i, S_j) = \frac{\sum_{\boldsymbol{x}_i \in S_i, \boldsymbol{x}_j \in S_j} proximity(\boldsymbol{x}_i, \boldsymbol{x}_j)}{|S_i| \cdot |S_j|}$$



Nested clusters

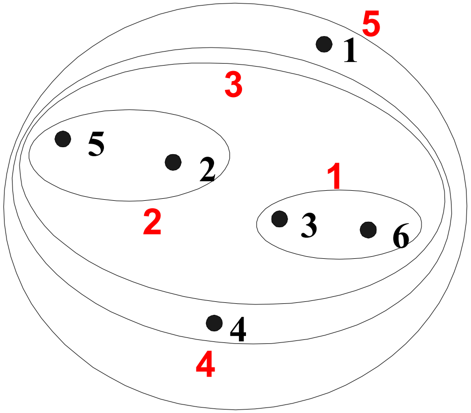


Dendrogram

Pro and Con of Group Average

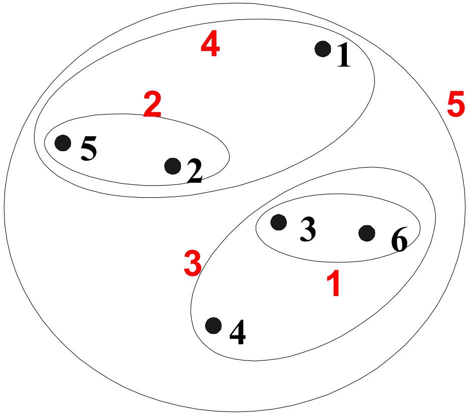
- Compromise between single and complete link
- Strengths: less susceptible to noise and outliers
- Limitations: biased towards globular clusters

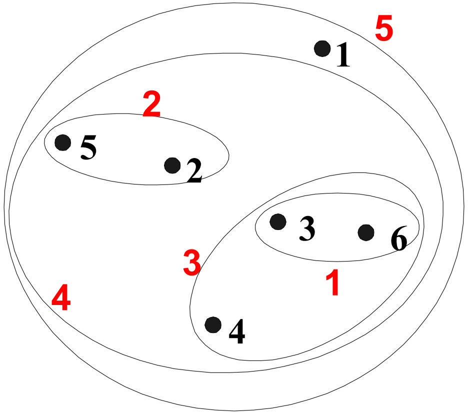
Comparison of Hierarchical Clustering Methods



Minimum

Maximum





Group average

Problems and Limitations of Hierarchical Clustering

- 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

What after Clustering?

- Cluster evaluation
- Why?
  - Compare the results of two different sets of cluster analyses to determine which is better
  - Determine the ‘correct’ number of clusters K
  - Determine the clustering tendency of a set of data, i.e., distinguish whether non-random structure actually exists in the data
- Approaches for cluster evaluation
  - Internal measures
  - Similarity matrix

Internal Measures: Cohesion and Separation

- Cluster cohesion: measures how closely related are objects in a cluster, e.g., within-cluster sum of squares w

$$w = \sum_{m=1}^{K} \sum_{\boldsymbol{x}_i \in S_m} d(\boldsymbol{x}_i, \overline{\boldsymbol{x}}_m) = \sum_{m=1}^{K} \sum_{\boldsymbol{x}_i \in S_m} (\boldsymbol{x}_i - \overline{\boldsymbol{x}}_m)^2 \in \mathfrak{R}$$

- Cluster separation: measure how distinct or well-separated a cluster is from other clusters, e.g., between-cluster sum of squares b

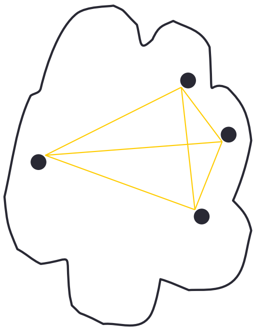
$$b = \sum_{m=1}^{K} |S_m| \cdot d(\overline{\boldsymbol{x}}_m, \overline{\boldsymbol{x}}) = \sum_{m=1}^{K} |S_m| \cdot (\overline{\boldsymbol{x}}_m - \overline{\boldsymbol{x}})^2 \in \mathfrak{R}$$

where $|S_m|$ is the size of cluster m

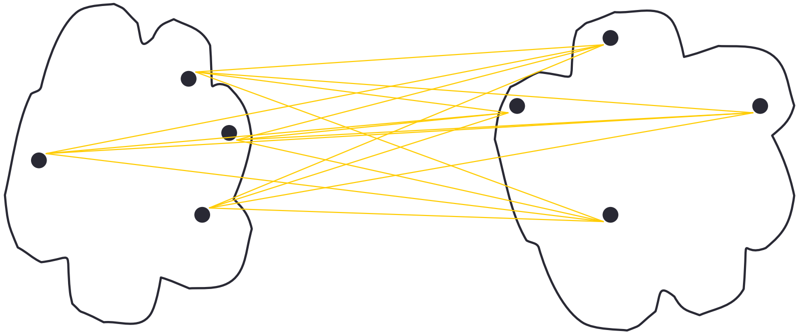
Geometric Explanation

- A proximity graph based approach can also be used for cohesion and separation
- Total sum of squares $t \in \Re$:

$$t = w + b = constant$$



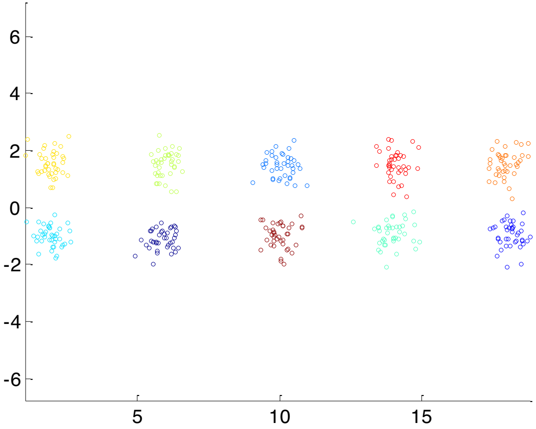
cohesion w

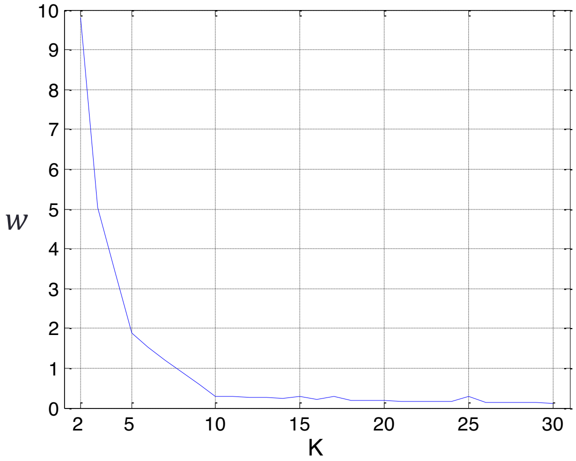


separation b

Using Within-cluster SS to Determine k

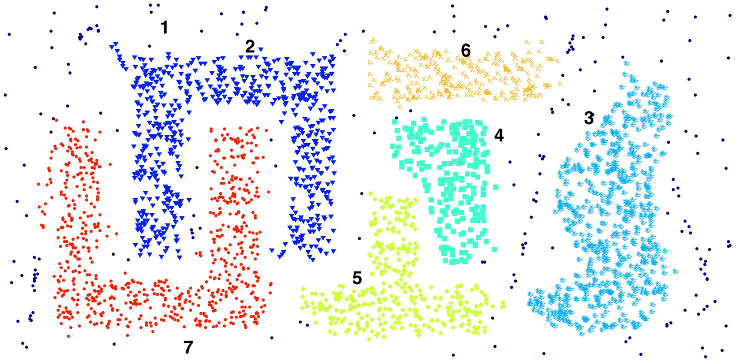
- Within−cluster sum of squares w may also be used to estimate the number of clusters
- Example:

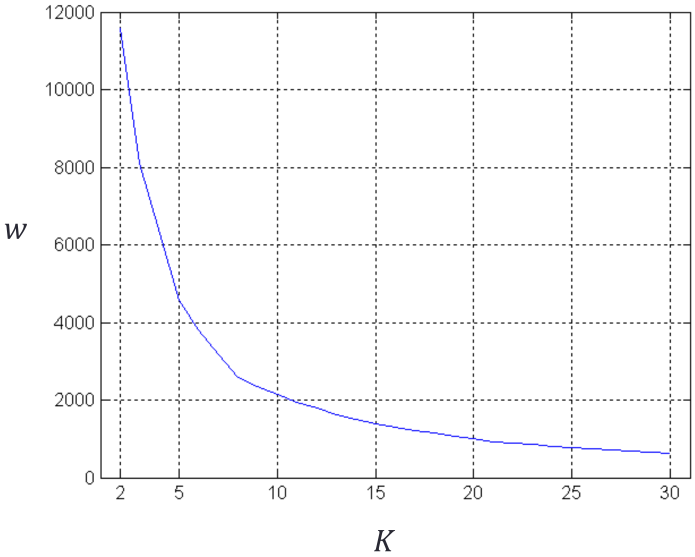




Using Within-cluster SS to Determine k (Cont’d)

- May not work on all data sets
- Example: w curve for a more complicated data set

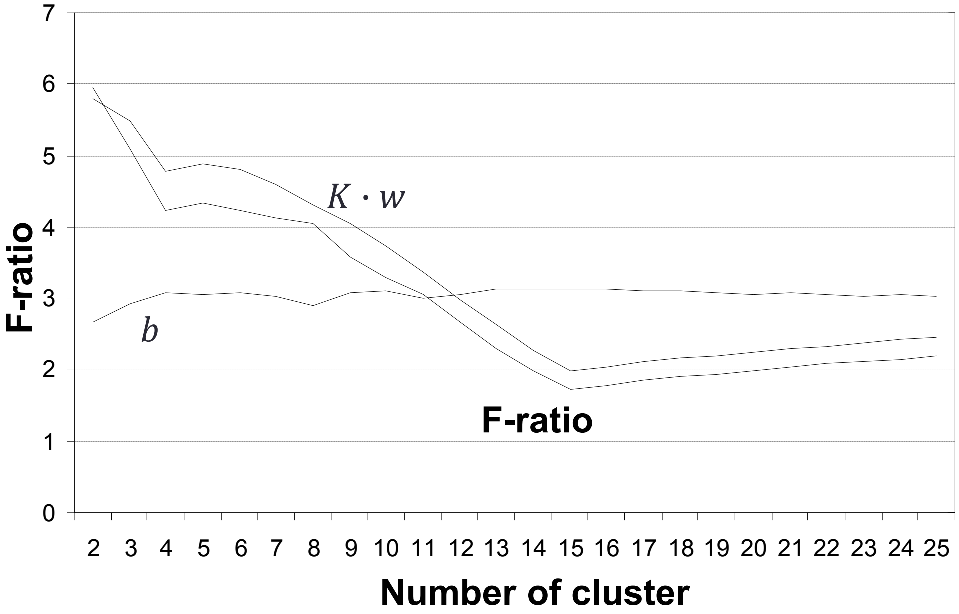




F-ratio Variance Test

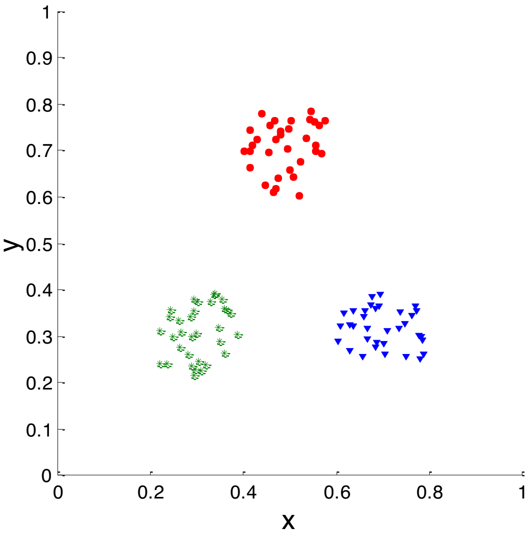
- F-ratio:

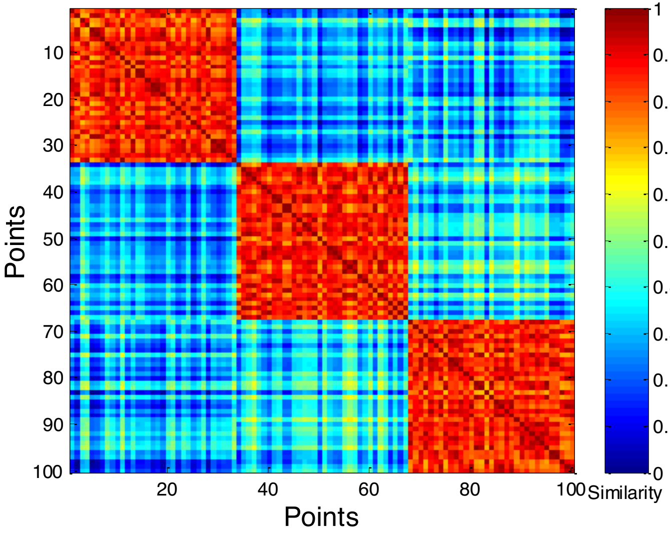
$$F = \frac{K \cdot w}{b}$$



Similarity Matrix for Cluster Validation

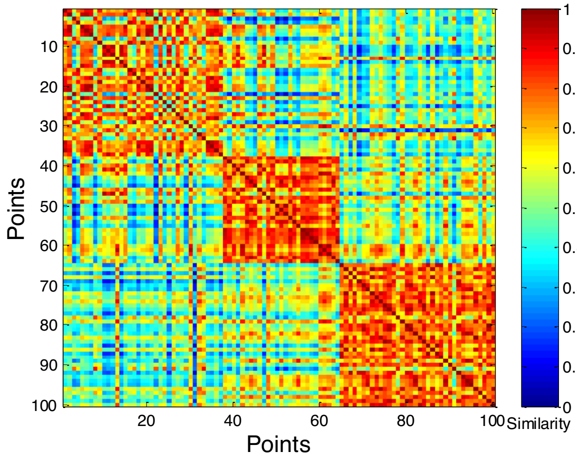
- Similarity matrix can be a tool for cluster validation
- Order the similarity matrix with respect to cluster labels and inspect visually

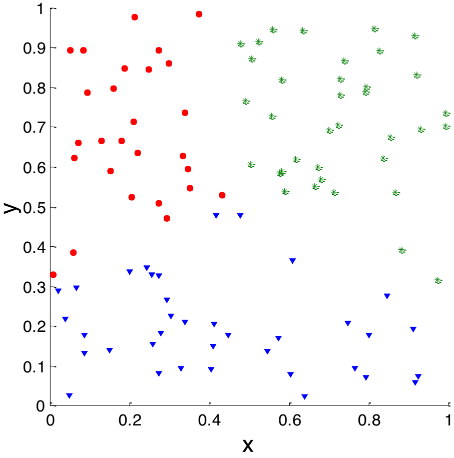




Similarity Matrix for Cluster Validation

- Clusters in random data are not so crisp
- Example: random data clustering with k-means





Summary

- Clustering is a subjective process; the same set of data items often needs to be partitioned differently for different applications
- The validation of clustering structures is the most difficult and frustrating part of cluster analysis
- Correlation (similarity) or WSS may not be a good measure for some density or contiguity based clusters

References

- P. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*