Cluster Analysis

What is Cluster Analysis?

- Cluster: a collection of data objects
  - Similar to one another within the same cluster
  - Dissimilar to the objects in other clusters
- Cluster analysis
  - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters

What is Cluster Analysis?

- Clustering analysis is an important human activity
- Early in childhood, we learn how to distinguish between cats and dogs
- Unsupervised learning: no predefined classes
- Typical applications
  - As a stand-alone tool to get insight into data distribution
  - As a preprocessing step for other algorithms

Clustering: Rich Applications and Multidisciplinary Efforts

- Pattern Recognition
- Spatial Data Analysis
  - Create thematic maps in GIS by clustering feature spaces
  - Detect spatial clusters or for other spatial mining tasks
- Image Processing
- Economic Science (especially market research)
- WWW
  - Document classification
  - Cluster Weblog data to discover groups of similar access patterns

Quality: What Is Good Clustering?

- A good clustering method will produce high quality clusters with
  - high intra-class similarity

    (Similar to one another within the same cluster)
  - low inter-class similarity

    (Dissimilar to the objects in other clusters)

- The quality of a clustering method is also measured by its ability to discover some or all of the hidden patterns

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 $i = (x_{i1}, x_{i2}, ..., x_{ip})$ and $j = (x_{j1}, x_{j2}, ..., x_{jp})$ 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:

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

- Also, one can use weighted distance, parametric Pearson correlation, or other disimilarity measures

Major Clustering Approaches

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

Some Other Major Clustering Approaches

- Grid-based approach:
  - based on a multiple-level granularity structure
  - Typical methods: STING, WaveCluster, CLIQUE
- Model-based:
  - A model is hypothesized for each of the clusters and tries to find the best fit of that model to each other
  - Typical methods: EM, SOM, COBWEB
- Frequent pattern-based:
  - Based on the analysis of frequent patterns
  - Typical methods: pCluster
- User-guided or constraint-based:
  - Clustering by considering user-specified or application-specific constraints
  - Typical methods: COD (obstacles), constrained clustering

Clustering Approaches

1. Partitioning Methods
2. Hierarchical Methods
3. Density-Based Methods

Partitioning Algorithms: Basic Concept

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

$$\Sigma_{m=1}^{k}\Sigma_{t_{mi}\in Km}(C_m - t_{mi})^2$$

Partitioning Algorithms: Basic Concept

- Given a *k*, find a partition of *k clusters* that optimizes the chosen partitioning criterion

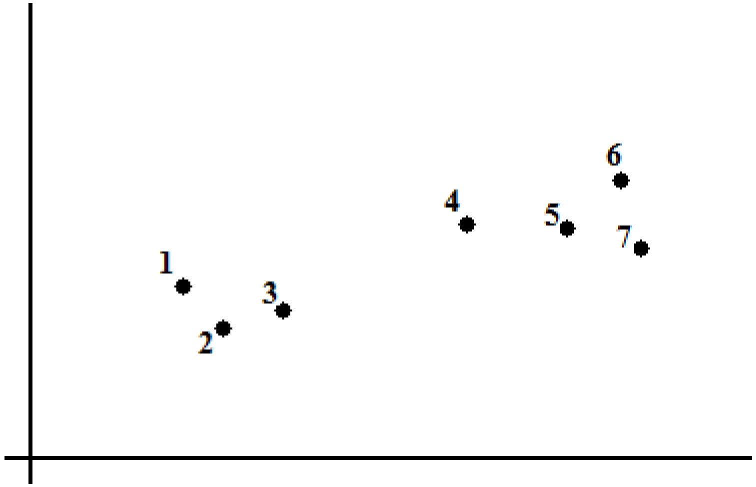
  - Global optimal: exhaustively enumerate all partitions

  - Heuristic methods: *k-means* and *k-medoids* algorithms
    - *k-means* (MacQueen'67): Each cluster is represented by the center of the cluster
    - *k-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Each cluster is represented by one of the objects in the cluster

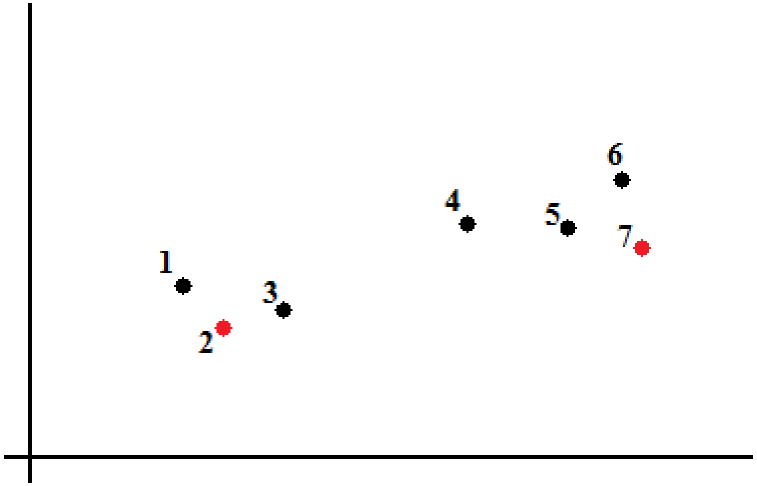
The *K-Means* Clustering Method

- Given *k*, the *k-means* algorithm is implemented in four steps:
  1. Partition objects into *k* nonempty subsets
  2. Compute seed points as the centroids of the clusters of the current partition (the centroid is the center, i.e., *mean point*, of the cluster)
  3. Assign each object to the cluster with the nearest seed point
  4. Go back to Step 2, stop when no more new assignment

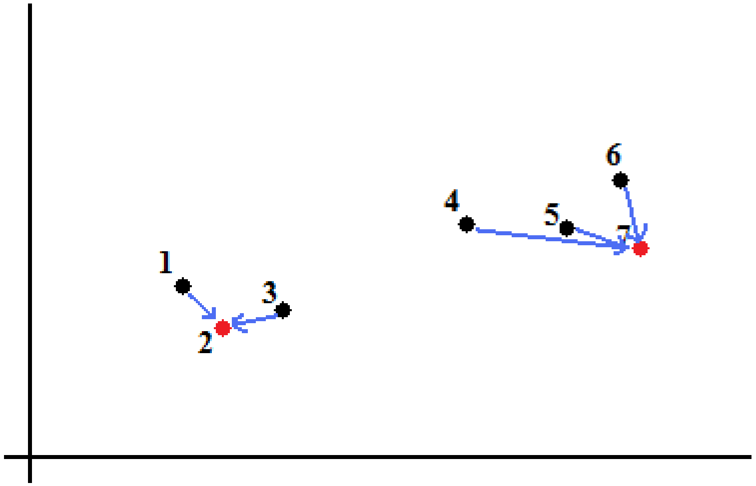
K-means Clustering



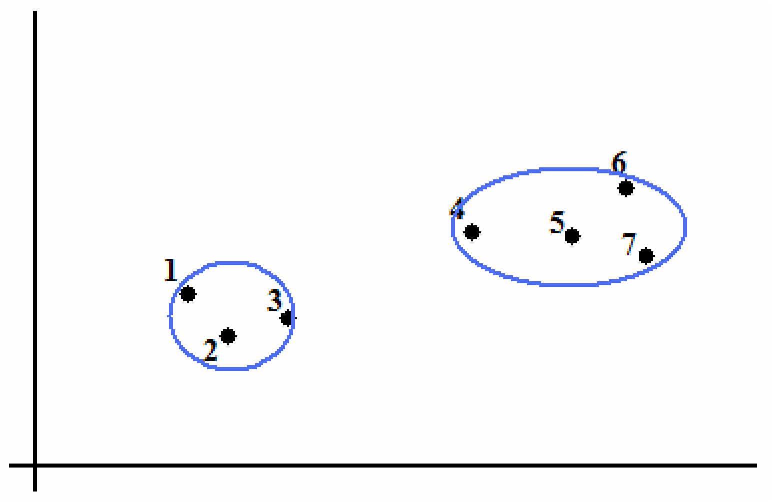
K-means Clustering



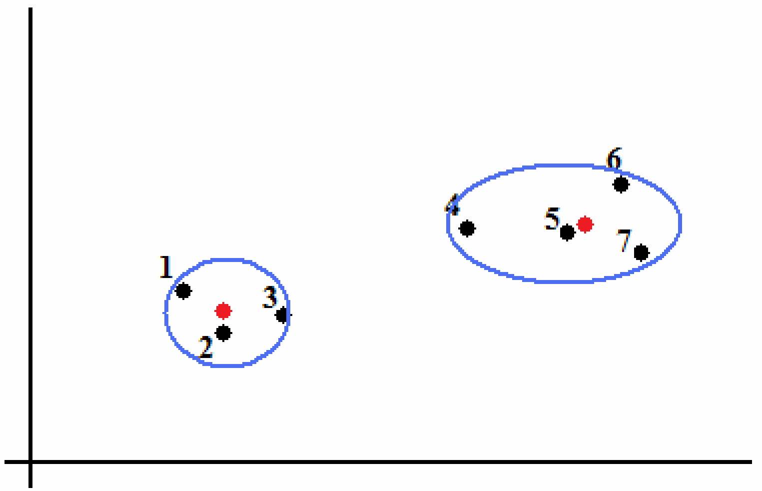
K-means Clustering



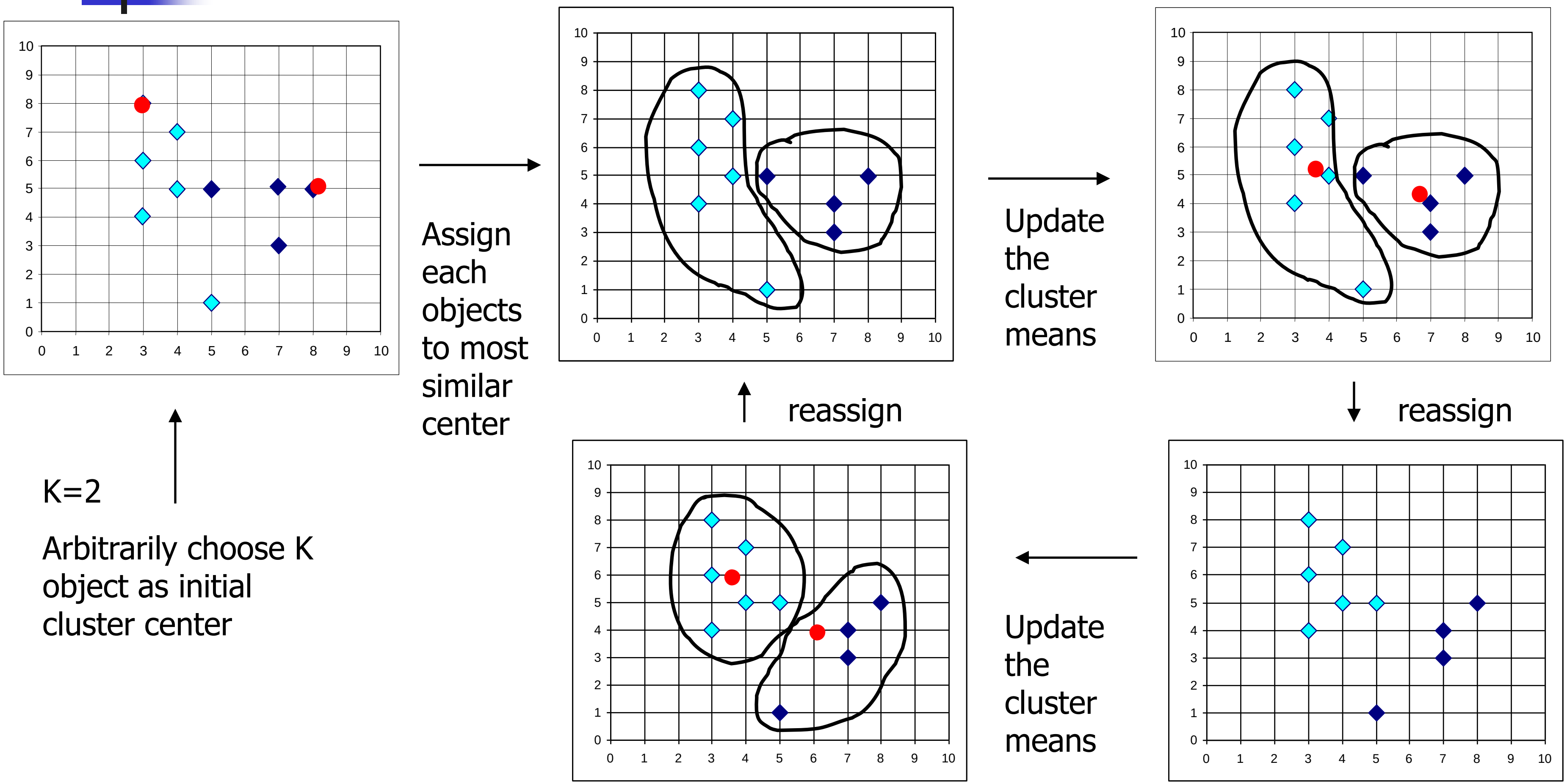
K-means Clustering



K-means Clustering

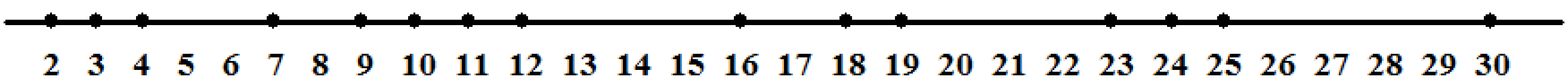


The *K-Means* Clustering Method



Example

- Run K-means clustering with 3 clusters (initial centroids: 3, 16, 25) for at least 2 iterations



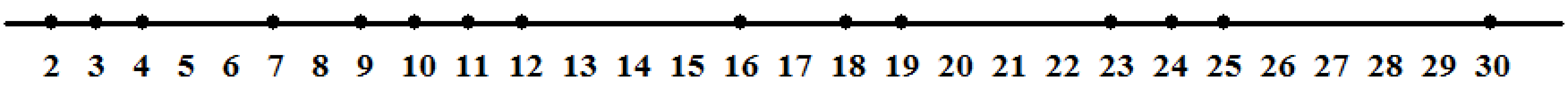
Example

- Centroids:

3 – 2 3 4 7 9 new centroid: 5

16 – 10 11 12 16 18 19 new centroid: 14.33

25 – 23 24 25 30 new centroid: 25.5



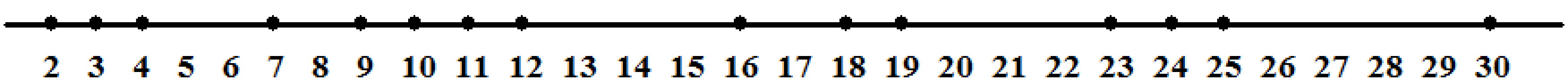
Example

- Centroids:

5 – 2 3 4 7 9 new centroid: 5

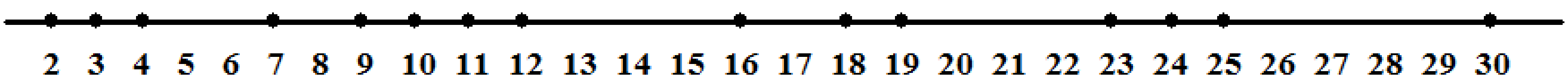
14.33 – 10 11 12 16 18 19 new centroid: 14.33

25.5 – 23 24 25 30 new centroid: 25.5



In class Practice

- Run K-means clustering with 3 clusters (initial centroids: 3, 12, 19) for at least 2 iterations



Typical Alternatives to Calculate the Distance between Clusters

- Centroid: distance between the centroids of two clusters, i.e., $dis(K_i, K_j) = dis(C_i, C_j)$
  - Centroid: the "middle" of a cluster $C_m = \frac{\sum_{i=1}^{N}(t_{ip})}{N}$

- Medoid: distance between the medoids of two clusters, i.e., $dis(K_i, K_j) = dis(M_i, M_j)$
  - Medoid: one chosen, centrally located object in the cluster

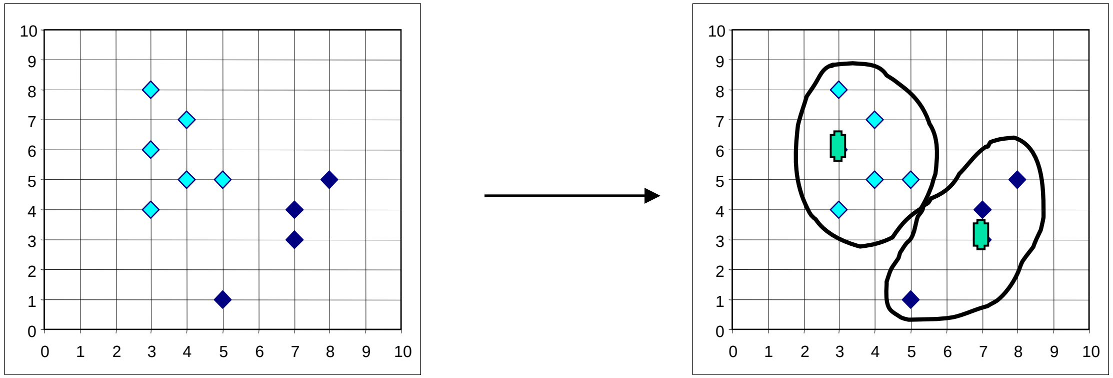
Comments on the *K-Means* Method

- Strength: *Relatively efficient*: $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, k, $t << n$.

- Weakness
  - Applicable only when *mean* is defined, then what about categorical data?
  - Need to specify *k,* the *number* of clusters, in advance
  - Unable to handle noisy data and *outliers*
  - Not suitable to discover clusters with *non-convex shapes*

What Is the Problem of the 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.

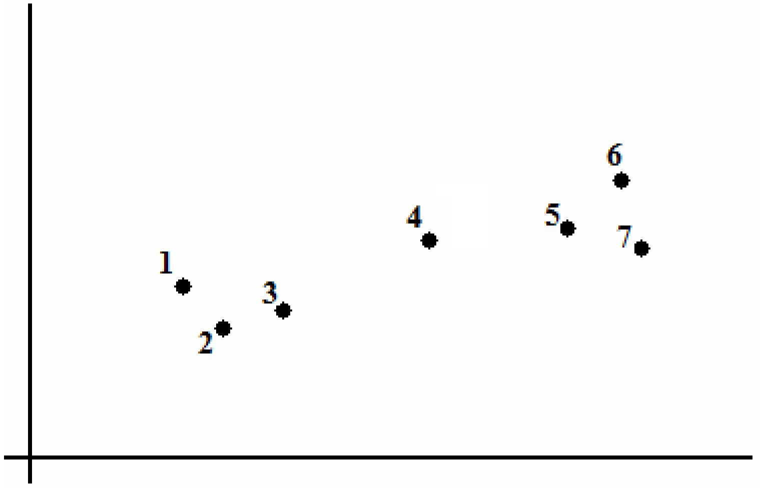


Fuzzy C-means Clustering

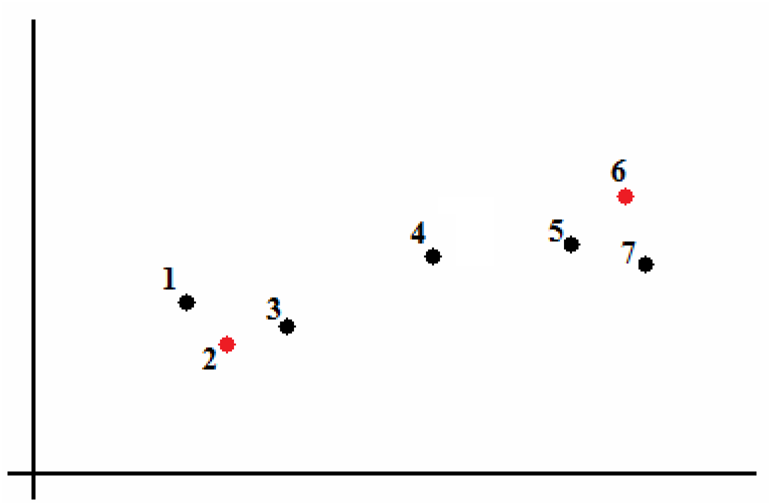
- Fuzzy c-means (FCM) is a method of clustering which allows one piece of data to belong to two or more clusters.

- This method (developed by Dunn in 1973 and improved by Bezdek in 1981) is frequently used in pattern recognition.

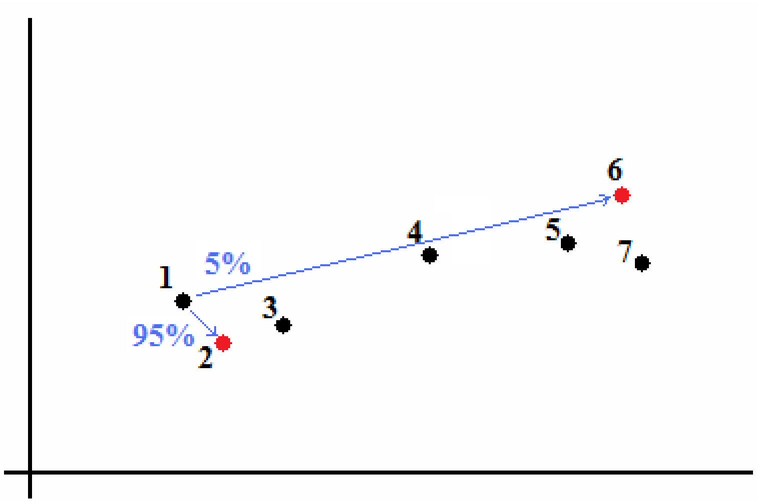
Fuzzy C-means Clustering



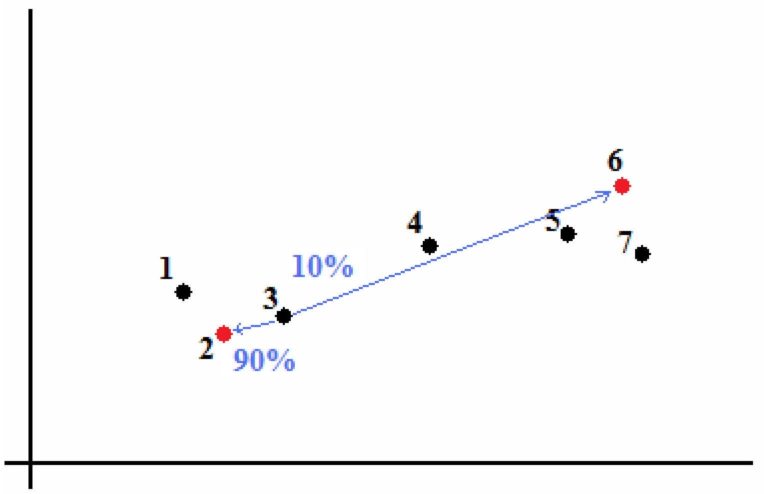
Fuzzy C-means Clustering



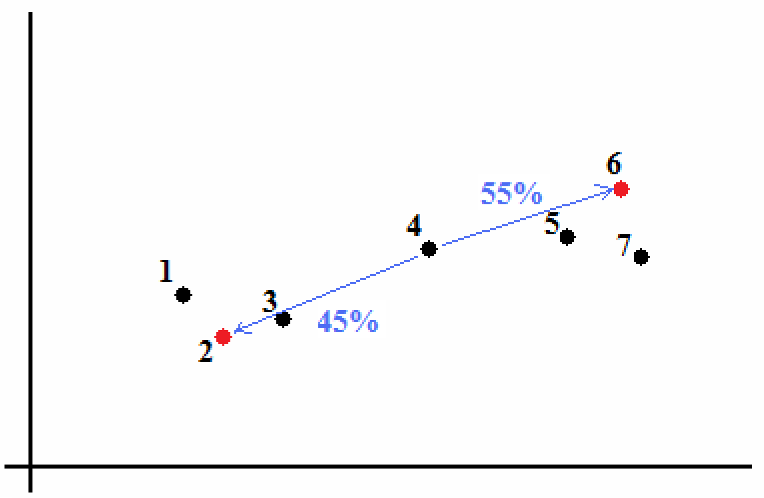
Fuzzy C-means Clustering



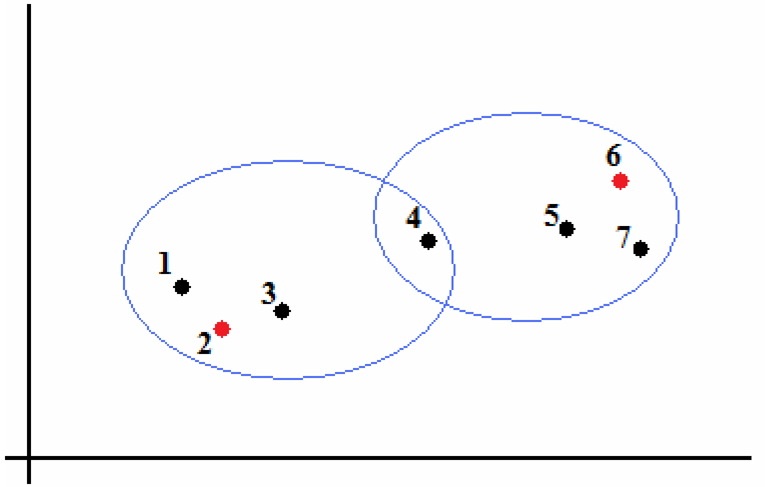
Fuzzy C-means Clustering



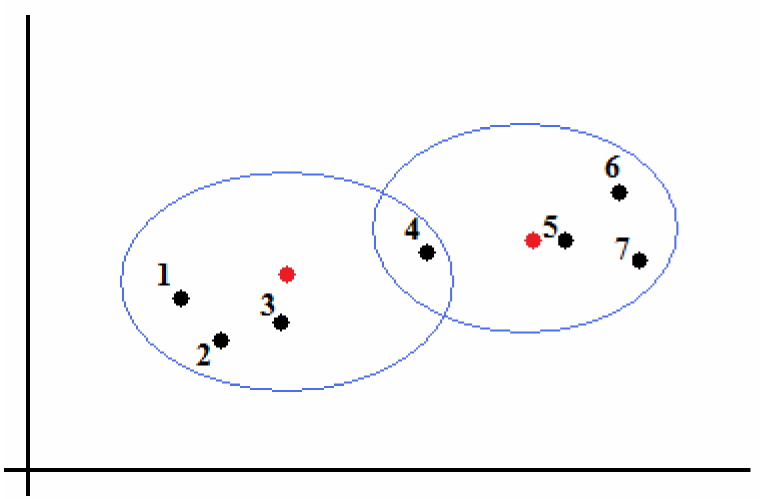
Fuzzy C-means Clustering



Fuzzy C-means Clustering



Fuzzy C-means Clustering



Fuzzy C-means Clustering

1. *Initialize* $U=[u_{ij}]$ *matrix,* $U^{(0)}$
2. *At k-step: calculate the centers vectors* $C^{(k)}=[c_j]$ *with* $U^{(k)}$

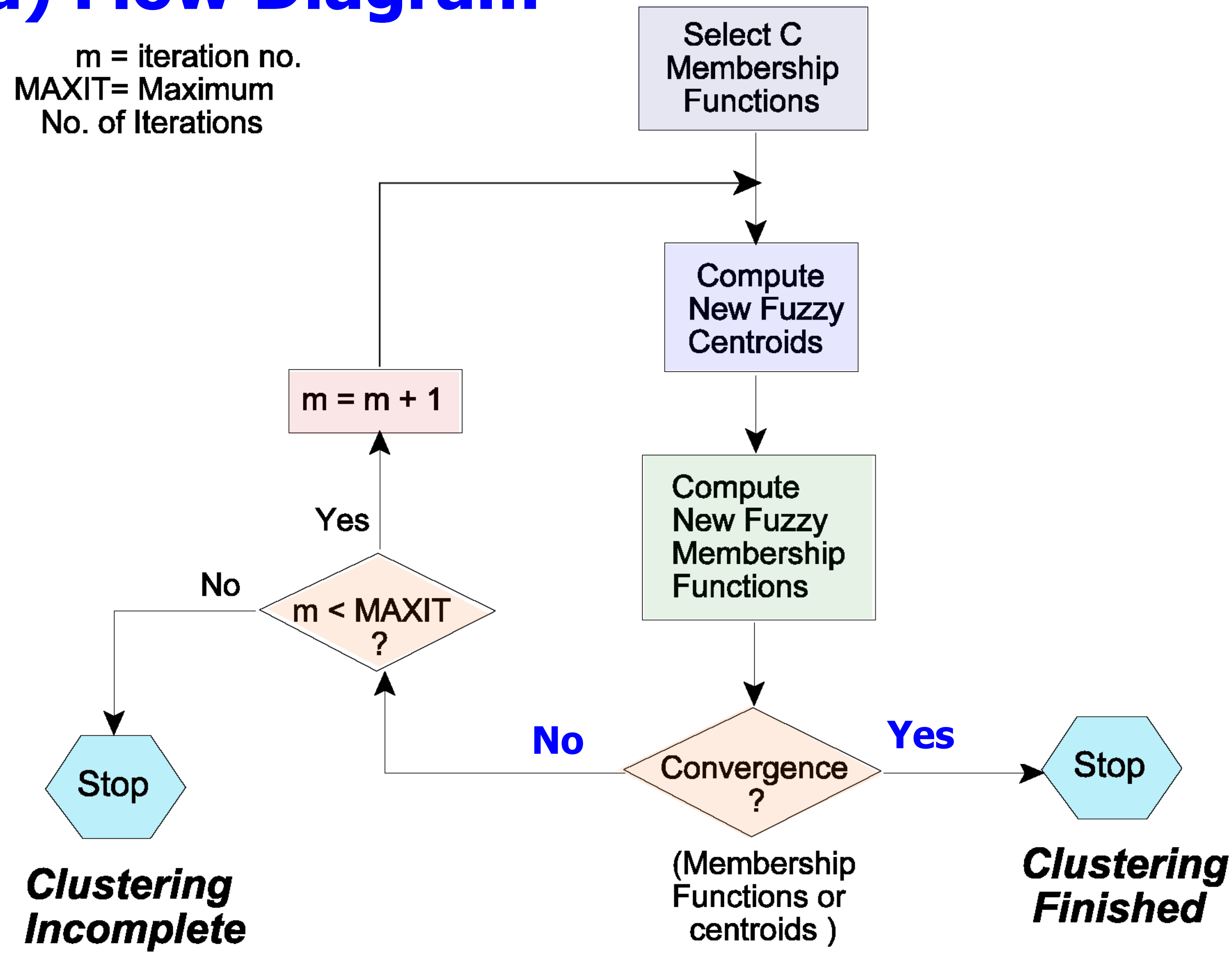
$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^m \cdot x_i}{\sum_{i=1}^{N} u_{ij}^m}$$

3. *Update* $U^{(k)}$ *,* $U^{(k+1)}$

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}}$$

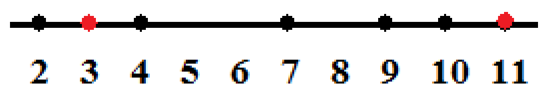
4. *If* $\| U^{(k+1)} - U^{(k)} \| < \varepsilon$ *then STOP; otherwise return to step 2*

Fuzzy C-Means Clustering Algorithm
(a) Flow Diagram



Fuzzy C-means Clustering

- For example: we have initial centroid 3 & 11 (with m=2)



$$u_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|}\right)^{\frac{2}{m-1}}}$$

u_{ij} is the degree of membership of x_i in the cluster j.

- For node 2 (1st element):

U11 = $$\frac{1}{\left(\frac{2-3}{2-3}\right)^{\frac{2}{2-1}} + \left(\frac{2-3}{2-11}\right)^{\frac{2}{2-1}}} = \frac{1}{1+\frac{1}{81}} = \frac{81}{82} = 98.78\%$$

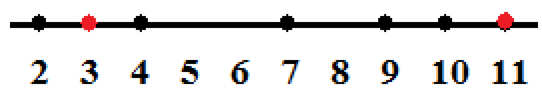
The membership of first node to first cluster

U12 = $$\frac{1}{\left(\frac{2-11}{2-3}\right)^{\frac{2}{2-1}} + \left(\frac{2-11}{2-11}\right)^{\frac{2}{2-1}}} = \frac{1}{81+1} = \frac{1}{82} = 1.22\%$$

The membership of first node to second cluster

Fuzzy C-means Clustering

- For example: we have initial centroid 3 & 11 (with m=2)



u_{ij} is the degree of membership of x_i in the cluster j.

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}}$$

- For node 3 (2^{nd} element):

U_{21} = 100%

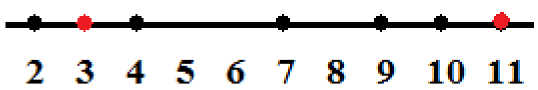
The membership of second node to first cluster

U_{22} = 0%

The membership of second node to second cluster

Fuzzy C-means Clustering

- For example: we have initial centroid 3 & 11 (with m=2)



$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\left\| x_i - c_j \right\|}{\left\| x_i - c_k \right\|} \right)^{\frac{2}{m-1}}}$$

u_{ij} is the degree of membership of x_i in the cluster j.

- For node 4 (3rd element):

U31 = $\dfrac{1}{\left(\frac{4-3}{4-3}\right)^{\frac{2}{2-1}} + \left(\frac{4-3}{4-11}\right)^{\frac{2}{2-1}}} = \dfrac{1}{1+\frac{1}{49}} = \dfrac{1}{\frac{50}{49}} = 98\%$

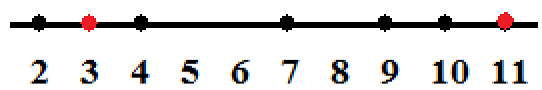
The membership of first node to first cluster

U32 = $\dfrac{1}{\left(\frac{4-11}{4-3}\right)^{\frac{2}{2-1}} + \left(\frac{4-11}{4-11}\right)^{\frac{2}{2-1}}} = \dfrac{1}{49+1} = \dfrac{1}{50} = 2\%$

The membership of first node to second cluster

Fuzzy C-means Clustering

- For example: we have initial centroid 3 & 11 (with m=2)



$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\| x_i - c_j \|}{\| x_i - c_k \|} \right)^{\frac{2}{m-1}}}$$

u_{ij} is the degree of membership of x_i in the cluster j.

- For node 7 (4th element):

U41 = $\frac{1}{\left(\frac{7-3}{7-3}\right)^{\frac{2}{2-1}} + \left(\frac{7-3}{7-11}\right)^{\frac{2}{2-1}}} = \frac{1}{1+1} = \frac{1}{2} = 50\%$

The membership of fourth node to first cluster

U42 = $\frac{1}{\left(\frac{7-11}{7-3}\right)^{\frac{2}{2-1}} + \left(\frac{7-11}{7-11}\right)^{\frac{2}{2-1}}} = \frac{1}{1+1} = \frac{1}{2} = 50\%$

The membership of fourth node to second cluster

Fuzzy C-means Clustering

- C1= $\dfrac{(98.78\%)^2 * 2 + (100\%)^2 * 3 + (98\%)^2 * 4 + (50\%)^2 * 7 + \ldots}{(98.78\%)^2 + (100\%)^2 + (98\%)^2 + (50\%)^2 + \ldots}$

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} \cdot x_i}{\sum_{i=1}^{N} u_{ij}^{m}}$$

Example – Application of Fuzzy Clustering Algorithm

Given the following set of data vectors

$$x_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad x_2 = \begin{pmatrix} 1 \\ 2 \end{pmatrix} \quad x_3 = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

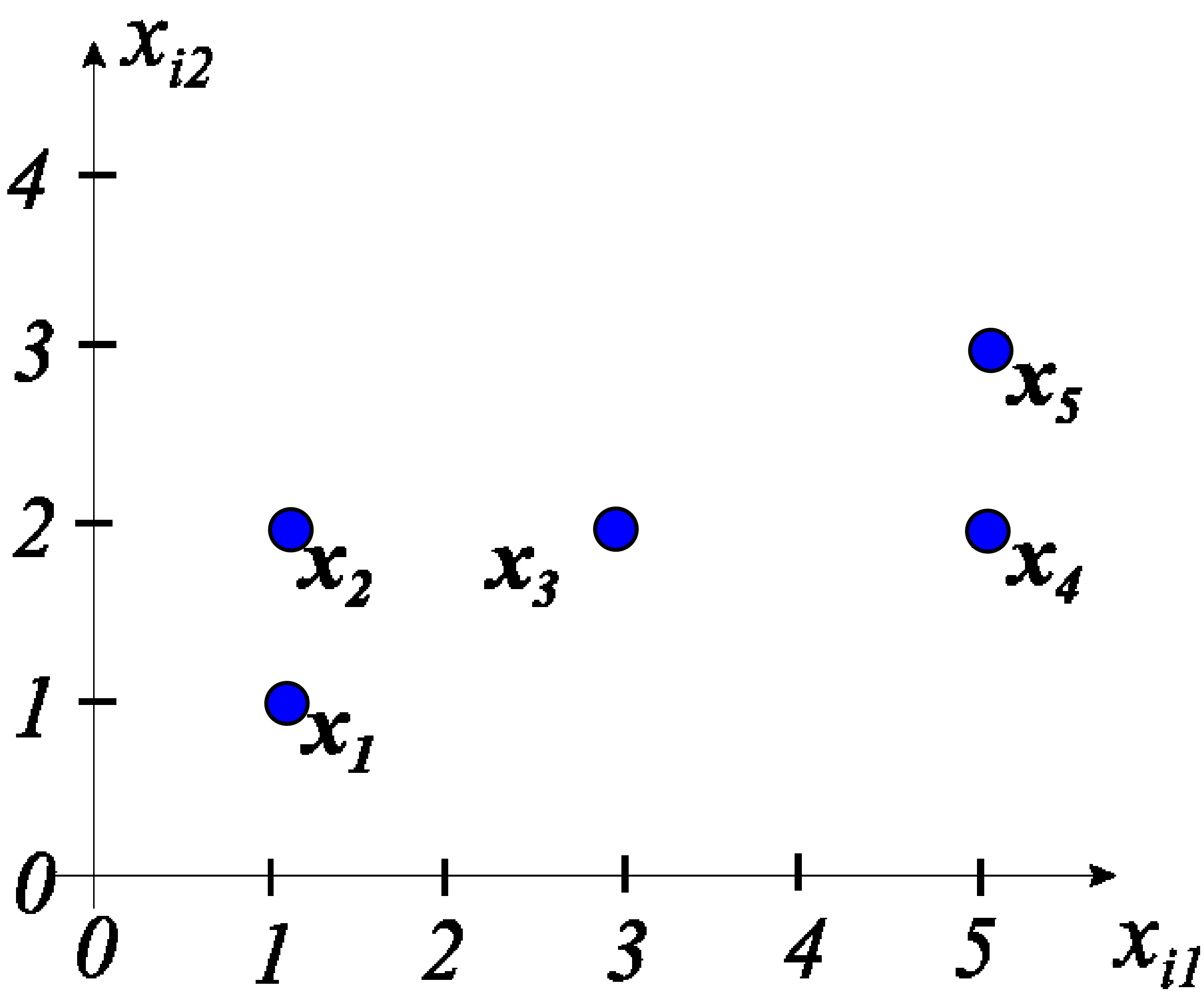
$$x_4 = \begin{pmatrix} 5 \\ 2 \end{pmatrix} \quad x_5 = \begin{pmatrix} 5 \\ 3 \end{pmatrix}$$

(a) Perform a **Fuzzy Clustering of the data** using **Fuzzy C-Means Algorithm** to obtain two fuzzy clusters. Try several initial conditions. Is the result unique? (use MAXIT=1000)

(b) Using the results of (a) **give a crisp clustering** of the data.

(c) Repeat (a) and (b) for **three Fuzzy Clusters.**

Plot of Data for Fuzzy clustering example



(a) **Solution for two clusters**

- Fuzzy cluster membership functions randomly selected

$$F_1(1): [\ .5302 \quad .2725 \quad .1124 \quad .3022 \quad .4881\ ]^T$$

$$F_2(1): [\ .4698 \quad .7275 \quad .8876 \quad .6978 \quad .5119\ ]^T$$

$$J(1) = 9.7305$$

Calculation of Fuzzy Centroids

$$V_1(2) = [\ 2.837 \quad 1.955\ ]^T$$

$$V_2(2) = [\ 3.084 \quad 2.023\ ]^T$$

Calculation of New Membership Functions

$$F_1(2):\ [\,.5230 \quad .5184 \quad .0342 \quad .4818 \quad .4778\,]^T$$

$$F_2(2):\ [\,.4770 \quad .4816 \quad .9658 \quad .5182 \quad .5222\,]^T$$

Calculation of Performance

$$J(2) = 8.972$$

Not the same as preceding iteration Membership function (No convergence)

Number of iterations not greater than 1000 therefore the iterations continue.

Results Converge at Iteration 17

Cluster membership Functions

$$F_1(17):\ [\ .8879 \quad .6596 \quad .5302 \quad .2725 \quad .1124\ ]^T$$

$$F_2(17):\ [\ .1121 \quad .3404 \quad .4698 \quad .7275 \quad .8876\ ]^T$$

Performance Measure

$$J(17) \ = \ ?.??$$

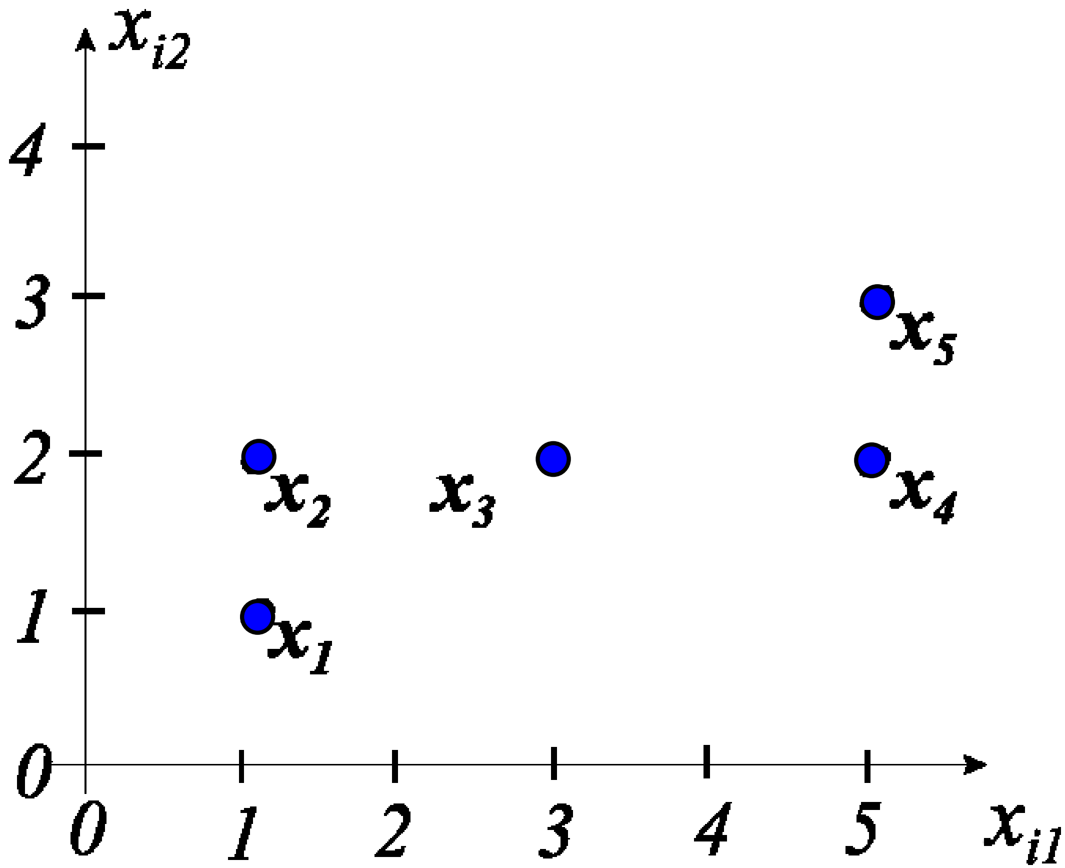
Cluster Centroids

$$V_1(17) \ = \ [\ 1.572 \quad 1.513\ ]^T$$

$$V_2(17) \ = \ [\ 4.427 \quad 2.465\ ]^T$$

(a) Final Cluster membership Functions

$$Cl_1: \quad [\ .8879 \quad .6596 \quad .5302 \quad .2725 \quad .1124\ ]^T$$
$$Cl_2: \quad [\ .1121 \quad .3404 \quad .4698 \quad .7275 \quad .8876\ ]^T$$



(b) Solution Crisp Clustering

Fuzzy Membership functions

$$Cl_1 : \quad [\,.8879 \quad .6596 \quad .5302 \quad .2725 \quad .1124\,]^T$$

$$Cl_2 : \quad [\,.1121 \quad .3404 \quad .4698 \quad .7275 \quad .8876\,]^T$$

Crisp Membership functions

$$Cl_1 : [\,1 \quad 1 \quad 1 \quad 0 \quad 0\,]^T$$

$$Cl_2 : [\,0 \quad 0 \quad 0 \quad 1 \quad 1\,]^T$$

Set Assignment

$$Cl_1 = \{\, \boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3 \,\}$$

$$Cl_2 = \{\, \boldsymbol{x}_4 , \boldsymbol{x}_5 \,\}$$

(c) Solution for three fuzzy clusters

- Applying the Fuzzy Clustering Algorithm convergence was obtained in ?? iterations as

$$V_1(L) = [\,1.001 \quad 1.491\,]^T$$
$$V_2(L) = [\,4.999 \quad 2.509\,]^T$$
$$V_3(L) = [\,3.000 \quad 2.000\,]^T$$

$$J_m = 0.9337$$

Final Cluster membership functions

$$Cl_1 : [\,.9421 \quad .9254 \quad .0000 \quad .0147 \quad .0124\,]^T$$
$$Cl_2 : [\,.0124 \quad .0147 \quad .0000 \quad .9254 \quad .9421\,]^T$$
$$Cl_3 : [\,.0455 \quad .0599 \quad 1.000 \quad .0599 \quad .0455\,]^T$$

(c) Solution Crisp Clustering

Membership functions

$$F_1 : [\ 1, 1, 0, 0, 0\ ]$$
$$F_2 : [\ 0, 0, 0, 1, 1\ ]$$
$$F_3 : [\ 0, 0, 1, 0, 0\ ]$$

Set Assignment

$$Cl_1 = \{\ \boldsymbol{x}_1 , \boldsymbol{x}_2\ \}$$
$$Cl_2 = \{\ \boldsymbol{x}_4 , \boldsymbol{x}_5\ \}$$
$$Cl_3 = \{\ \boldsymbol{x}_3\ \}$$

"Crisp Clusters"

Comments: Fuzzy Clustering Can be used to Produce Hard Clustering

The larger the value of m the fuzzier the clusters

The Fuzzy algorithm is relatively stable and usually converges in a reasonable number of iterations

The Fuzzy algorithm is relatively insensitve to initial conditions

Of the two different fuzzy clusterings given below, which clustering is the Fuzzier ???

 # 1

Cl_{1} : [0.52 0.51 0.04 0.47 0.46]

Cl_{2} : [0.48 0.49 0.96 0.5 0.53]

or

 # 2

Cl_{1} : [0.89 0.85 0.04 0.26 0.15]

Cl_{2} : [0.11 0.15 0.96 0.74 0.85]

ANSWER: #1 is the fuzzier of the two different clusterings

# 1	Cl_1 : [0.52 0.51 0.04 0.47 0.46]	
	Cl_2 : [0.48 0.49 0.96 0.53 0.53]	

# 2	Cl_1 : [0.89 0.85 0.04 0.26 0.15]	
	Cl_2 : [0.11 0.15 0.96 0.74 0.85]	

Why is #1 the Fuzzier of the two ???

ANSWER: Because the cluster membership functions contain many entries close to 0.5 (for the two class case) as opposed to values close to 0 and 1.

*For the M class case values close to 1/M would indicate most fuzziness..