GUC
German University in Cairo

Winter 2019

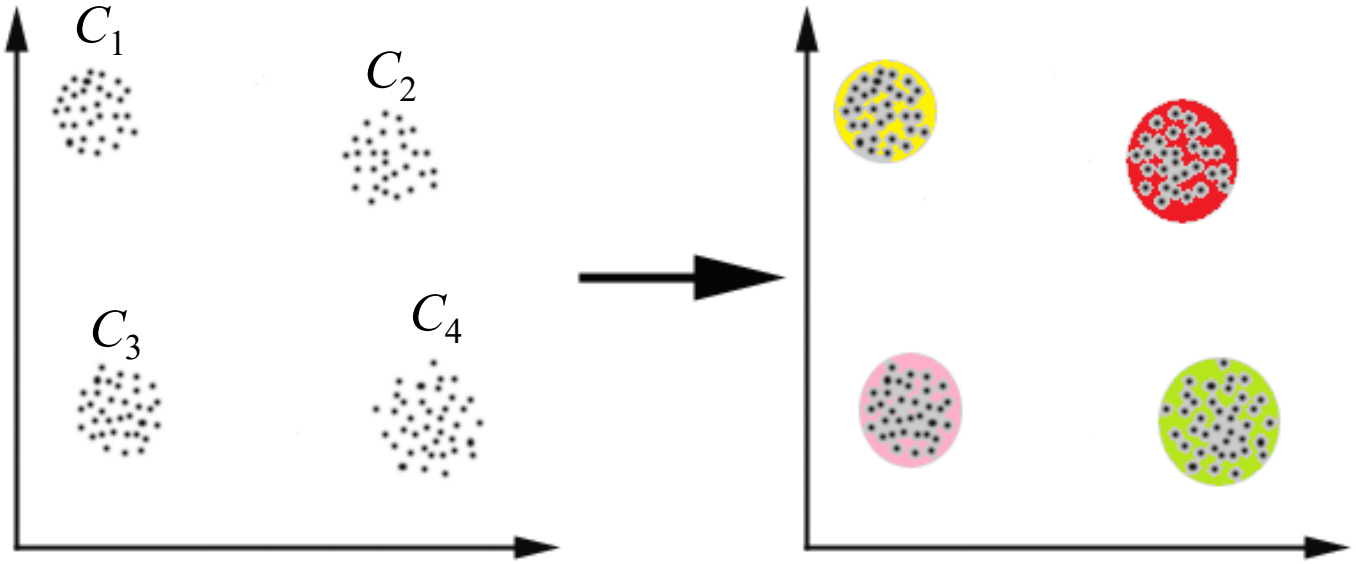
CSEN1022: Machine Learning

Clustering Techniques (1)

Seif Eldawlatly

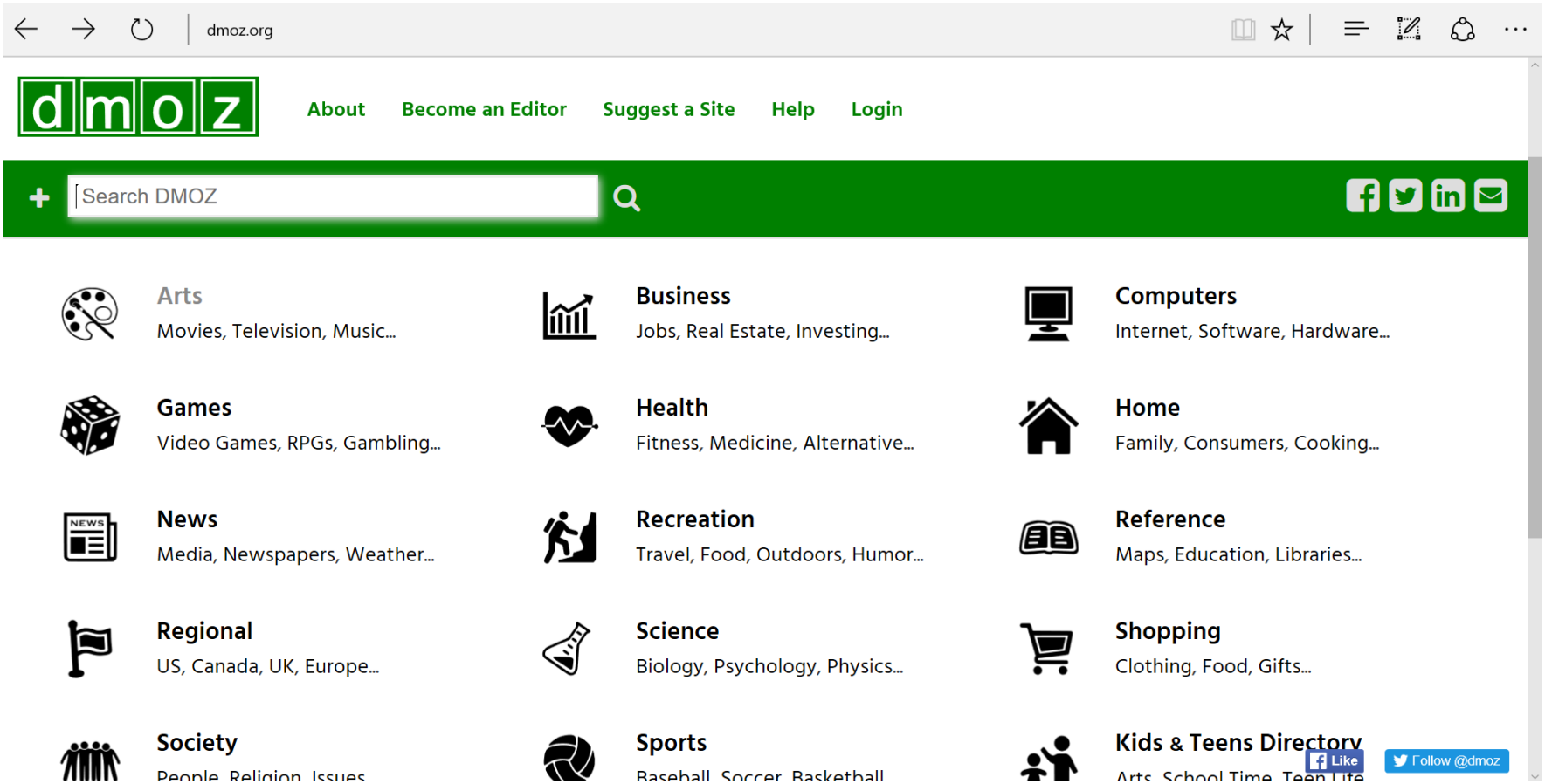
Unsupervised Learning

- Find hidden structure in unlabeled data



Example: Navigation

- Categorize websites



Example: Image Segmentation

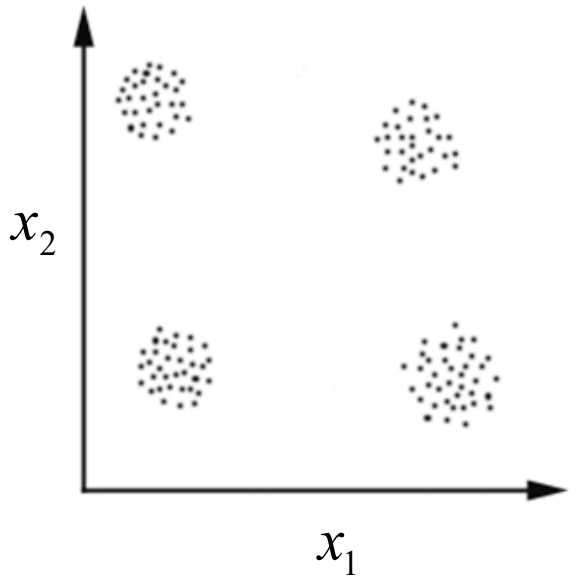
- Coloring pixels with close colors using a certain color

Original Image

Segmented Image

K-means Clustering

- Cluster: A group of data points whose inter-point distances are small compared with distances to points outside the cluster



- Objective Function: Minimize J

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

$\mathbf{x}_n$: Input data

$\boldsymbol{\mu}_k$: Center of cluster k

r_{nk}: Cluster membership $= 1$ if $\mathbf{x}_n \in C_k$

$= 0$ if $\mathbf{x}_n \notin C_k$

N : Number of data points

K : Number of clusters to look for

K-means Clustering

- Algorithm steps:
  - Step 1: Randomly choose clusters center $\boldsymbol{\mu}_k$
  - Step 2: Compute r_{nk}

$$r_{nk} = \begin{cases} 1 & \text{if } k = \arg\min_j \|\mathbf{x}_n - \boldsymbol{\mu}_j\|^2 \\ 0 & \text{otherwise.} \end{cases}$$

(Assign $\mathbf{x}_n$ to the cluster with closest center)

  - Step 3: Update $\boldsymbol{\mu}_k$

Take derivative of J w.r.t. $\boldsymbol{\mu}_k$ and equate with zero

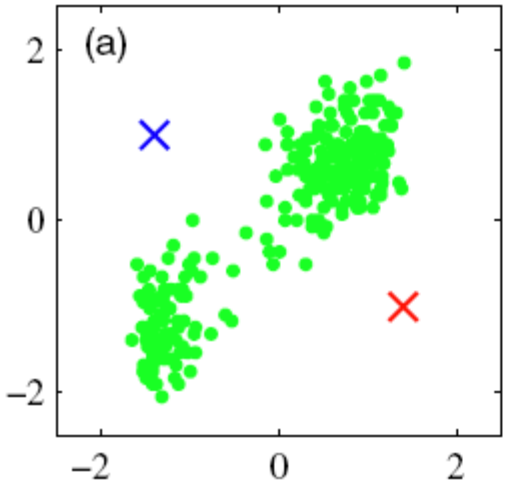
$$2\sum_{n=1}^{N} r_{nk}(\mathbf{x}_n - \boldsymbol{\mu}_k) = 0 \quad \rightarrow \quad \boldsymbol{\mu}_k = \frac{\sum_n r_{nk}\mathbf{x}_n}{\sum_n r_{nk}}$$

  - Back to Step 2 until convergence

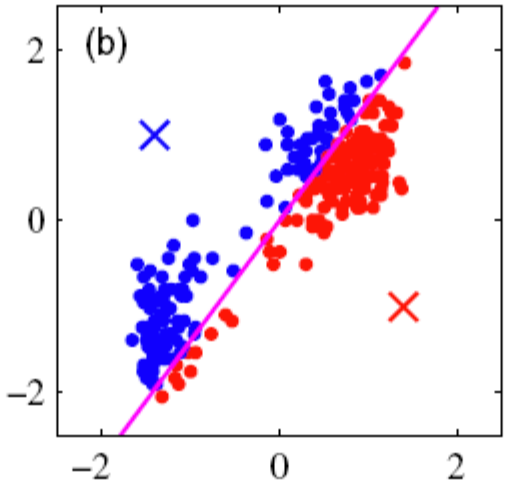
K-means Clustering

- Example

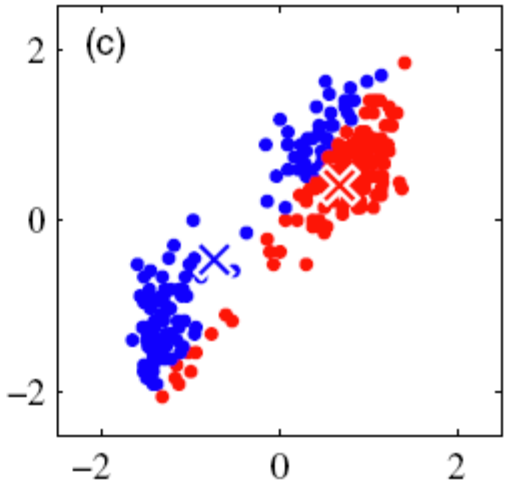
Randomly choose μ_k



Compute r_{nk}



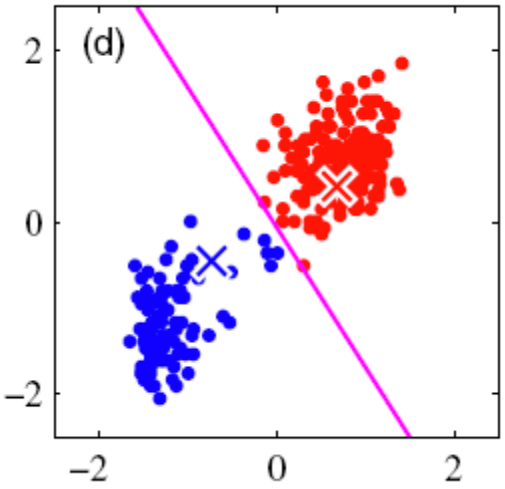
Update μ_k



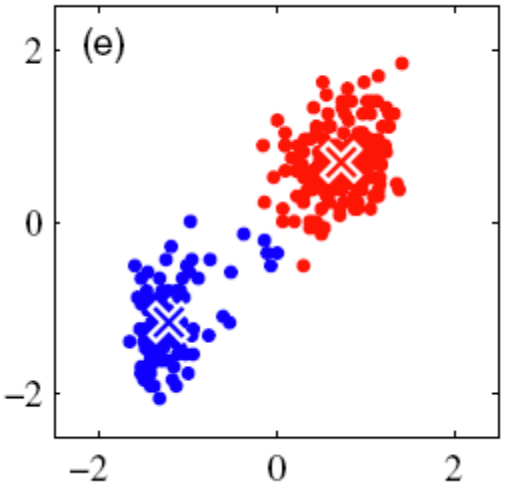
K-means Clustering

- Example

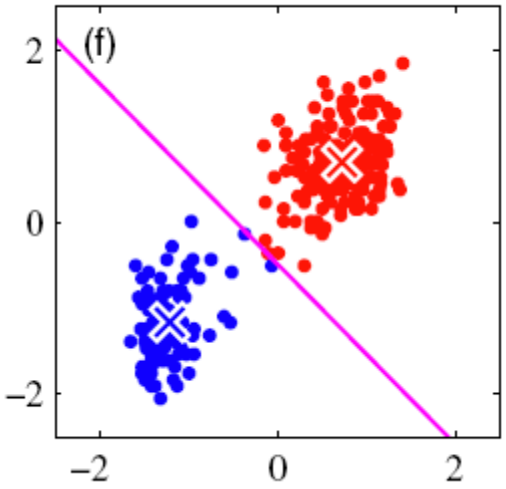
Compute r_{nk}



Update μ_k



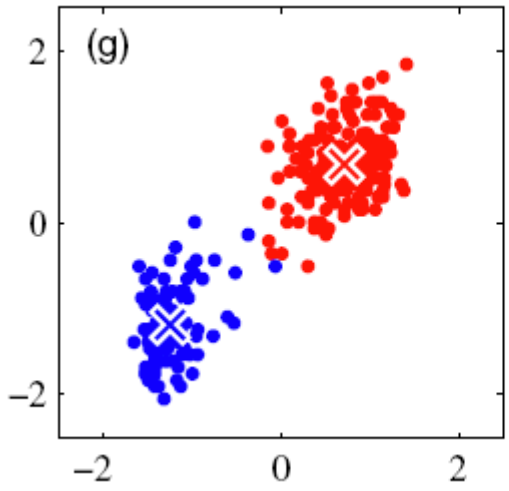
Compute r_{nk}



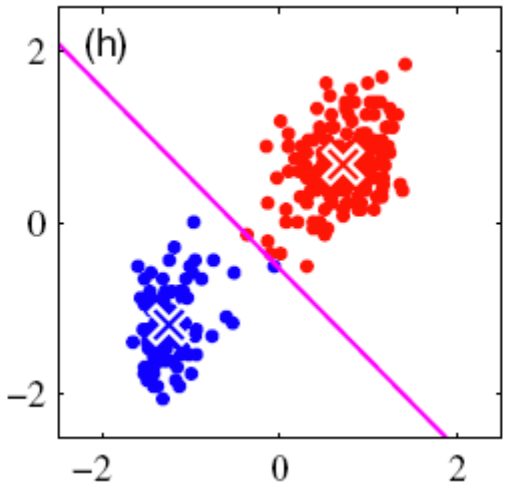
K-means Clustering

- Example

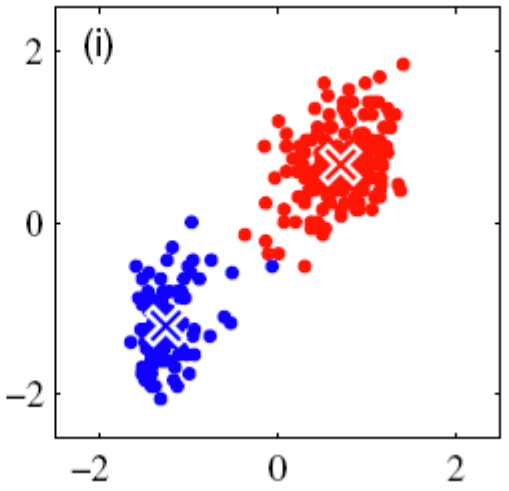
Update μ_k



Compute r_{nk}



Update μ_k



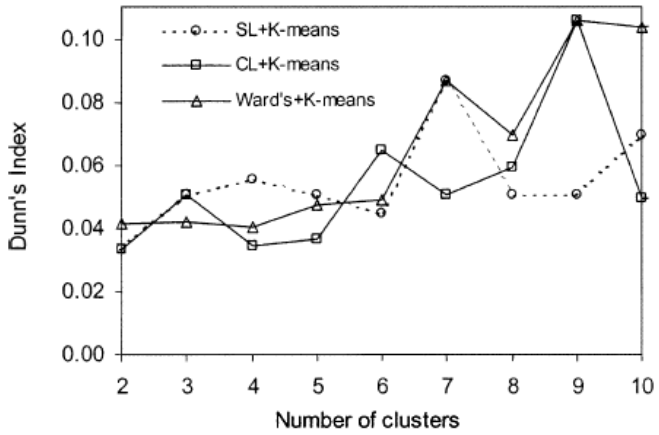
K-means Clustering

- Image Segmentation and Compression

Clustering Validity Indices

- How to choose K (or any other parameter)?

- Dunn Index:
  Find K that maximizes minimum separation between clusters and minimizes diameter of clusters



$$DI = \min_{\substack{1 \le i \le N, 1 \le j \le N \\ i \ne j}} d(C_i, C_j) \Big/ \max_{1 \le k \le N} diam(C_k)$$

where

$$d(C_i, C_j) = \min_{\mathbf{x} \in C_i, \mathbf{y} \in C_j} d(\mathbf{x}, \mathbf{y}), \; diam(C_k) = \max_{\mathbf{x} \in C_k, \mathbf{y} \in C_k} d(\mathbf{x}, \mathbf{y})$$

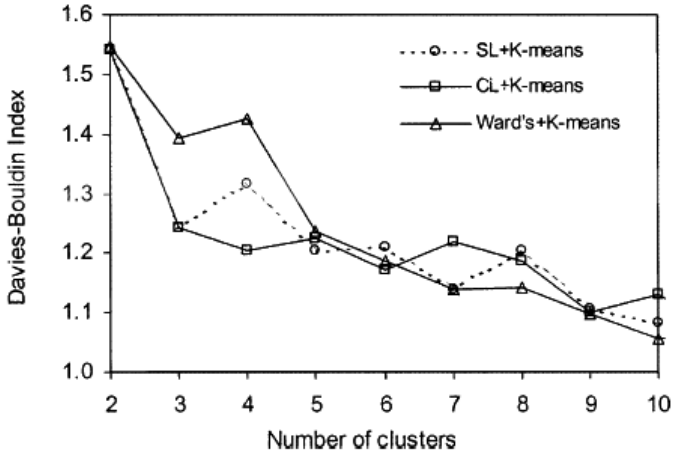
and $d(\mathbf{x}, \mathbf{y})$ is the Euclidean distance between points $\mathbf{x}$ and $\mathbf{y}$

- To choose K, find K that maximizes DI

Clustering Validity Indices

- How to choose K (or any other parameter)?

- Davis Bouldin Index:
  Find K that maximizes the distance between the clusters and their closest ones



$$DBI = \frac{1}{N}\sum_{i=1}^{N} \max_{j=1,\ldots,N,\, j\neq i} R_{ij}$$

where

$$R_{ij} = \frac{s_i + s_j}{d_{ij}}, \quad d_{ij} = d(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j), \quad s_i = \frac{1}{|C_i|}\sum_{\mathbf{x}\in C_i} d(\mathbf{x}, \boldsymbol{\mu}_i)$$

and $d(\mathbf{x}, \boldsymbol{\mu}_i)$ is the Euclidean distance between points $\mathbf{x}$ and $\boldsymbol{\mu}_i$ the center of class i

- To choose K, find K that minimizes DBI

Fuzzy C-means (FCM)

- Hard Partitioning: A data point can be a member of one cluster only $r_{nk} \in \{0,1\}$

- Soft Partitioning: A data point can be a member of more than one cluster but with different degrees of membership

$$r_{nk} \in [0,1], \sum_{k=1}^{K} r_{nk} = 1$$

- Objective Function

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk}^{q} \left\| \mathbf{x}_n - \mu_k \right\|^2$$

Subject to the constraint $\sum_{k=1}^{K} r_{nk} = 1$

where q is the fuzziness Index > 1

Fuzzy C-means (FCM)

- Objective Function: Minimize J subject to the constraints

$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{nk}^{q} \left\| \mathbf{x}_n - \mu_k \right\|^2 + \sum_{n=1}^{N} \lambda_n \left[\sum_{k=1}^{K} r_{nk} - 1 \right]$$

where λ_n are Lagrange multipliers.

- Taking derivative w.r.t. the following parameters and equate with zero

$$\mu_k \quad \rightarrow \quad \mu_k = \frac{\sum_{n=1}^{N} r_{nk}^{q} \mathbf{x}_n}{\sum_{n=1}^{N} r_{nk}^{q}}$$

$$r_{nk} \quad \rightarrow \quad r_{nk} = \frac{1}{\sum_{m=1}^{K} \left(\frac{\left\| \mathbf{x}_n - \mu_k \right\|^2}{\left\| \mathbf{x}_n - \mu_m \right\|^2} \right)^{\frac{1}{q-1}}}$$

Fuzzy C-means (FCM)

- Algorithm steps (similar to K-means):
  - Step 1: Randomly choose clusters center μ_k
  - Step 2: Compute r_{nk}

$$r_{nk} = \frac{1}{\sum_{m=1}^{K} \left(\frac{\left\| \mathbf{x}_n - \mu_k \right\|^2}{\left\| \mathbf{x}_n - \mu_m \right\|^2} \right)^{\frac{1}{q-1}}}$$

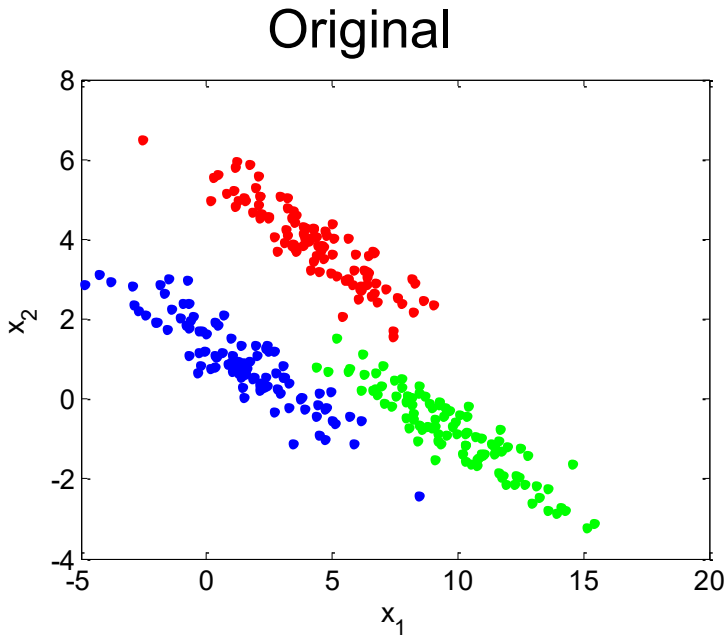
  - Step 3: Update μ_k

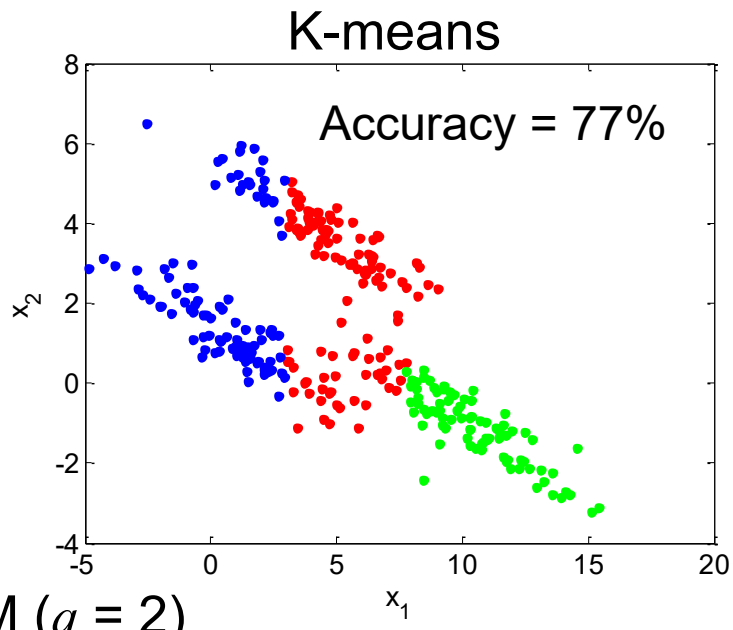
$$\mu_k = \frac{\sum_{n=1}^{N} r_{nk}^q \mathbf{x}_n}{\sum_{n=1}^{N} r_{nk}^q}$$

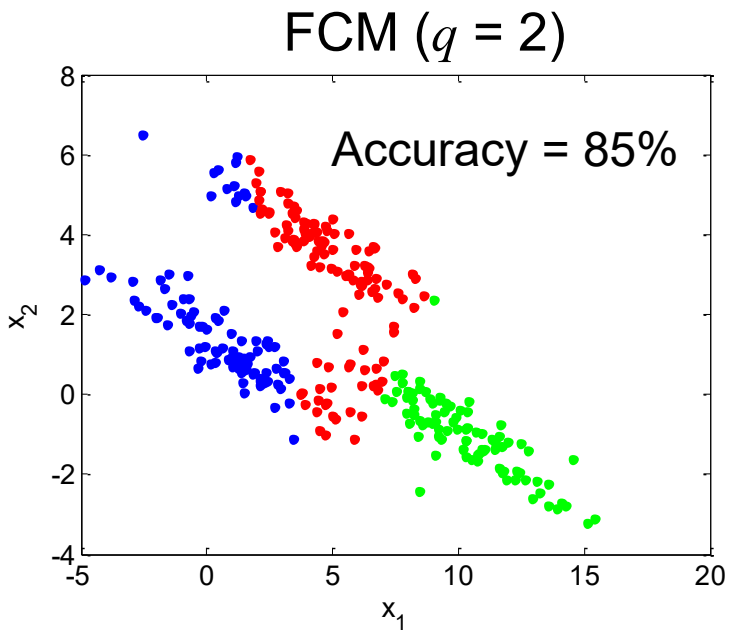
  - Back to Step 2 until convergence
  - Finally, $\mathbf{x}_n$ is assigned to cluster k^* that gives maximum membership

$$k^* = \arg\max_k r_{nk}$$

FCM vs. K-means

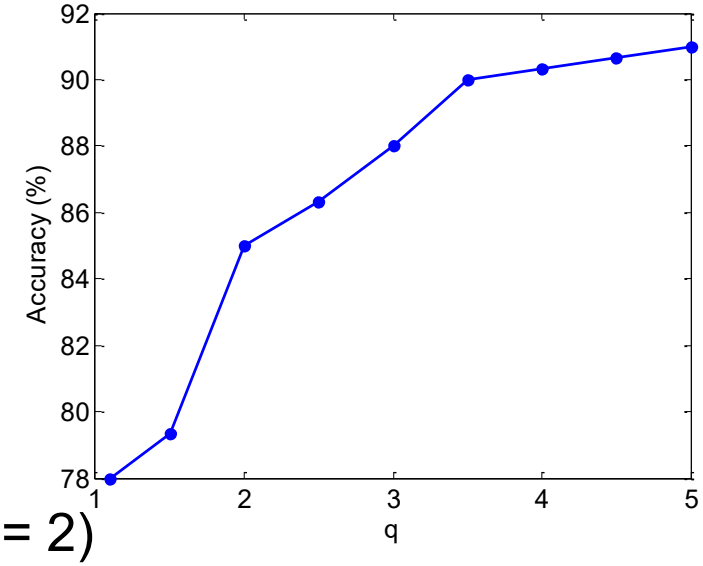




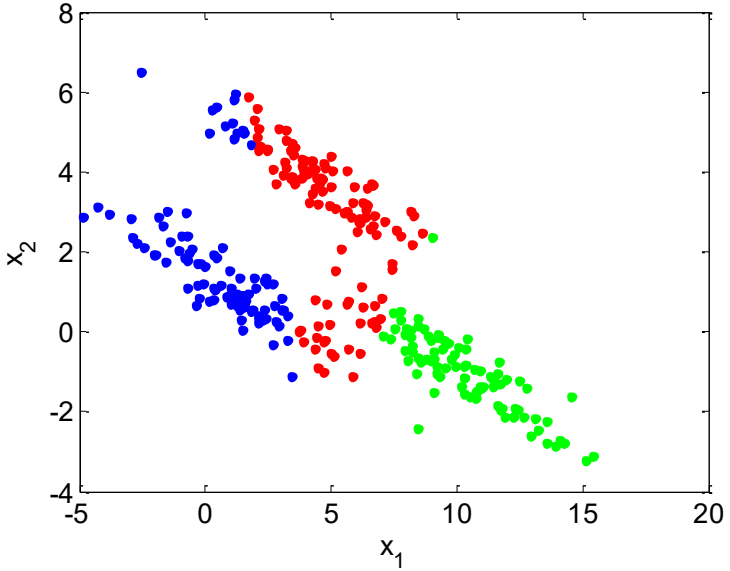


FCM vs. K-means

- Dependence on q



FCM (q = 2)



FCM (q = 5)