

# **CLUSTERING Methods**

**Prof. Sukhendu Das  
Deptt. of CS&E,  
IIT Madras,  
Chennai, INDIA.**

**Email: [sdas@cse.iitm.ac.in](mailto:sdas@cse.iitm.ac.in)**

# What is Cluster Analysis?

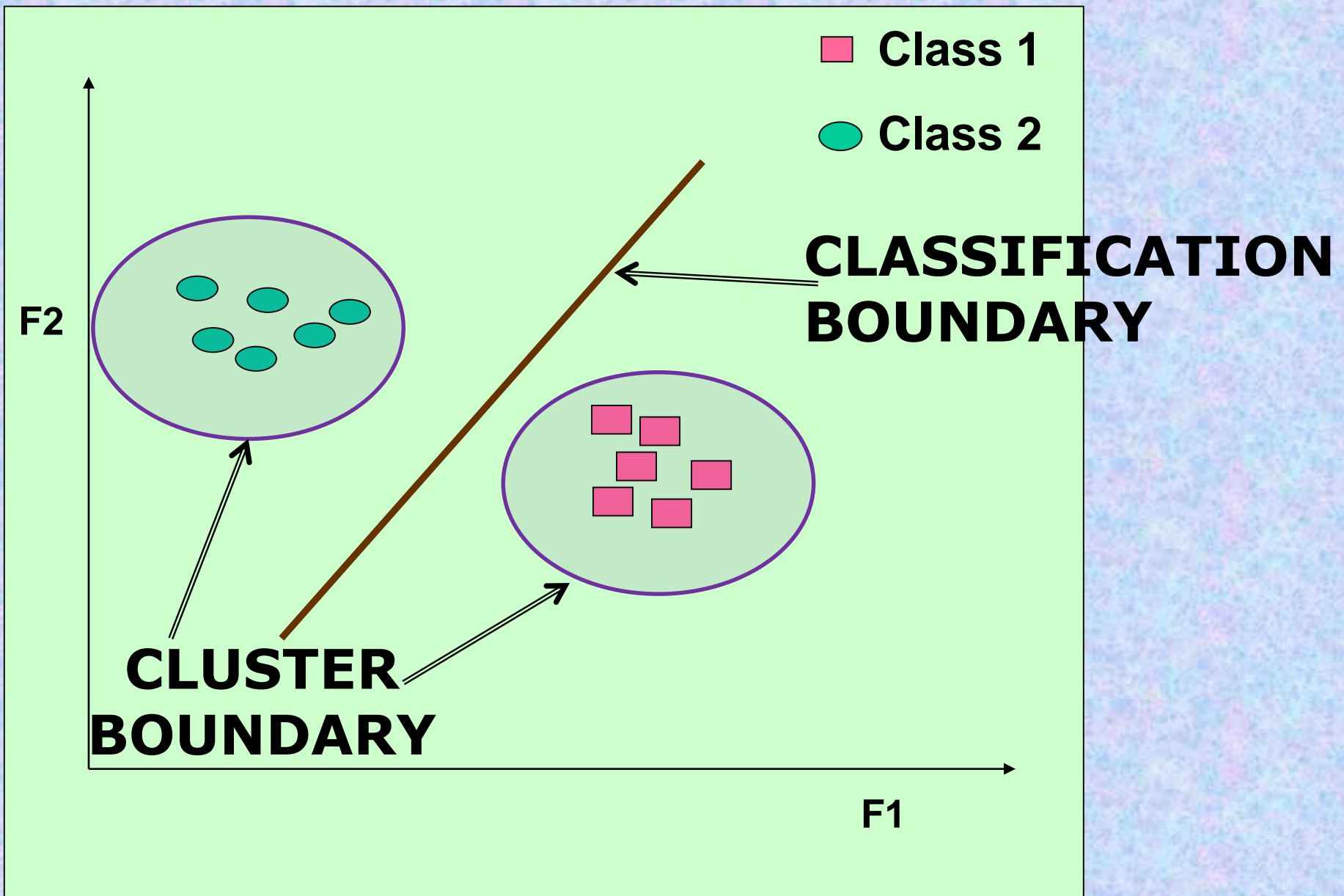
---

- Cluster: A collection of data objects
  - similar (or related) to one another within the same group
  - dissimilar (or unrelated) to the objects in other groups
- Cluster analysis (or *clustering*, *data segmentation*, ...)
  - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
- **Unsupervised learning**: no predefined classes (i.e., *learning by observations* vs. learning by examples: supervised)
- Typical applications
  - As a **stand-alone tool** to get insight into data distribution
  - As a **preprocessing step** for other algorithms

# Clustering: Application Examples

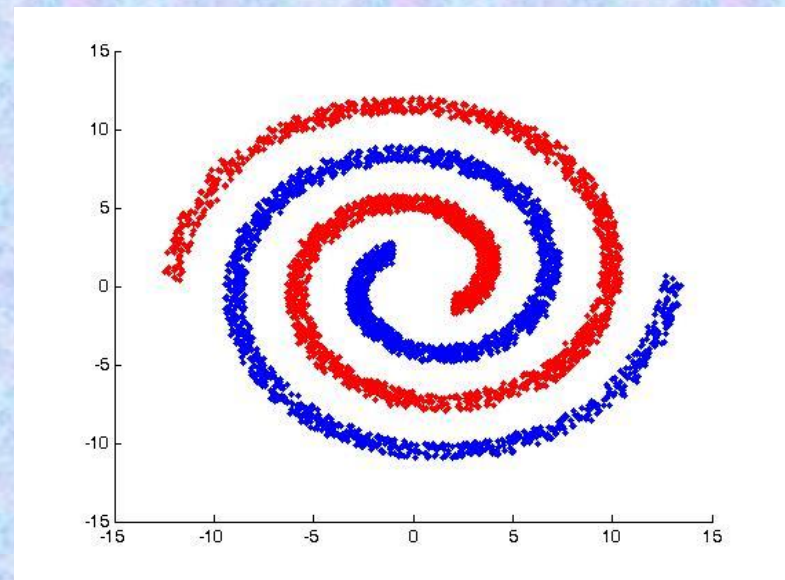
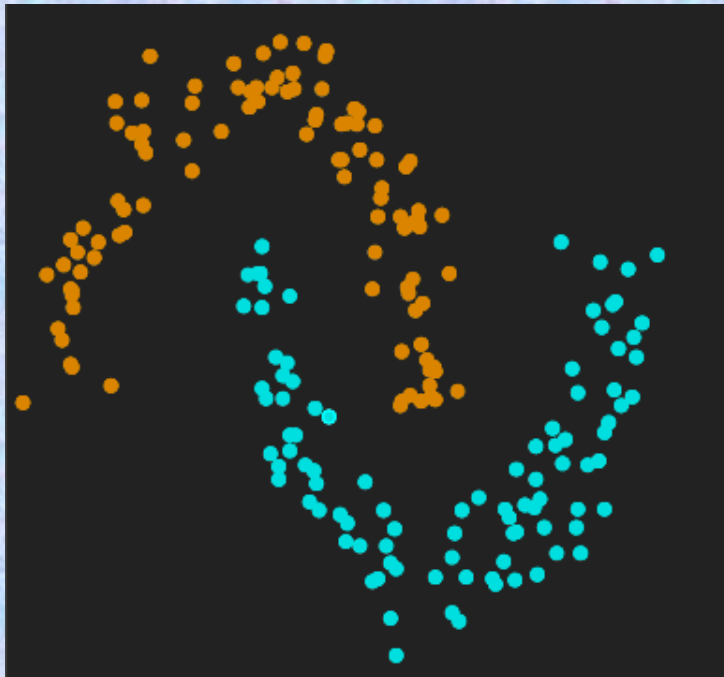
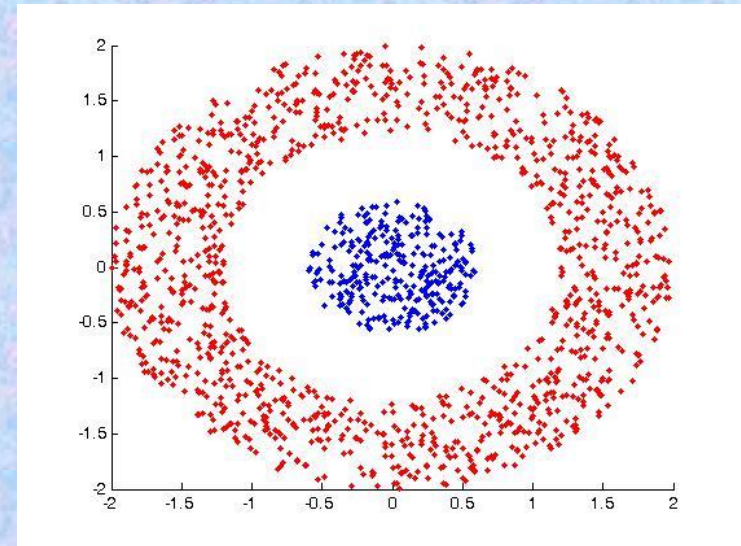
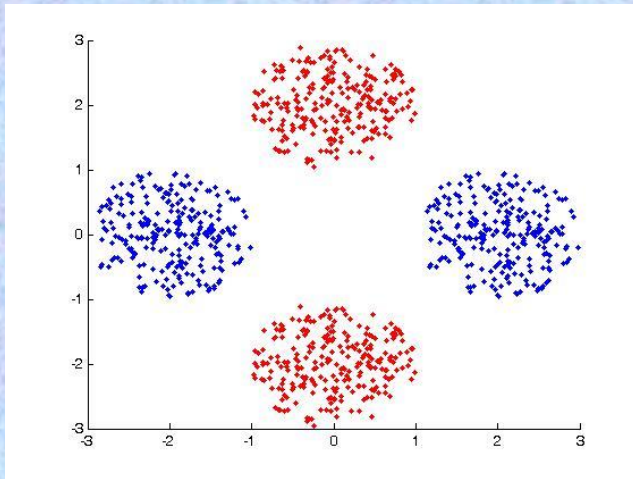
---

- **Biology:** taxonomy of living things: kingdom, phylum, class, order, family, genus and species
- **Information retrieval:** document clustering
- **Land use:** Identification of areas of similar land use in an earth observation database
- **Marketing:** Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- **City-planning:** Identifying groups of houses according to their house type, value, and geographical location
- **Earth-quake studies:** Observed earth quake epicenters should be clustered along continent faults
- **Climate:** understanding earth climate, find patterns of atmospheric and ocean
- **Economic Science:** market research



**Sample points in a two-dimensional feature space**

# Complex cases of classification and clustering



## **CLUSTERING**

**Data Points have  
no labels**

## **CLASSIFICATION**

**Most data points  
have labels**



# **CLUSTERING      METHODS OF      CLASSIFICATION** **AND**

- **REPRESENTATIVE POINTS**
- **Split & MERGE**
- **LINKAGE**
- **SOM**
- **MODEL-BASED**
- **VECTOR QUANTIZATION**

# Quality: What Is Good Clustering?

---

- A good clustering method will produce high quality clusters
  - high intra-class similarity: **cohesive** within clusters
  - low inter-class similarity: **distinctive** between clusters
- The quality of a clustering method depends on
  - the similarity measure used by the method
  - its implementation, and
  - Its ability to discover some or all of the hidden patterns



# Considerations for Cluster Analysis

---

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

# Major Clustering Approaches (I)

---

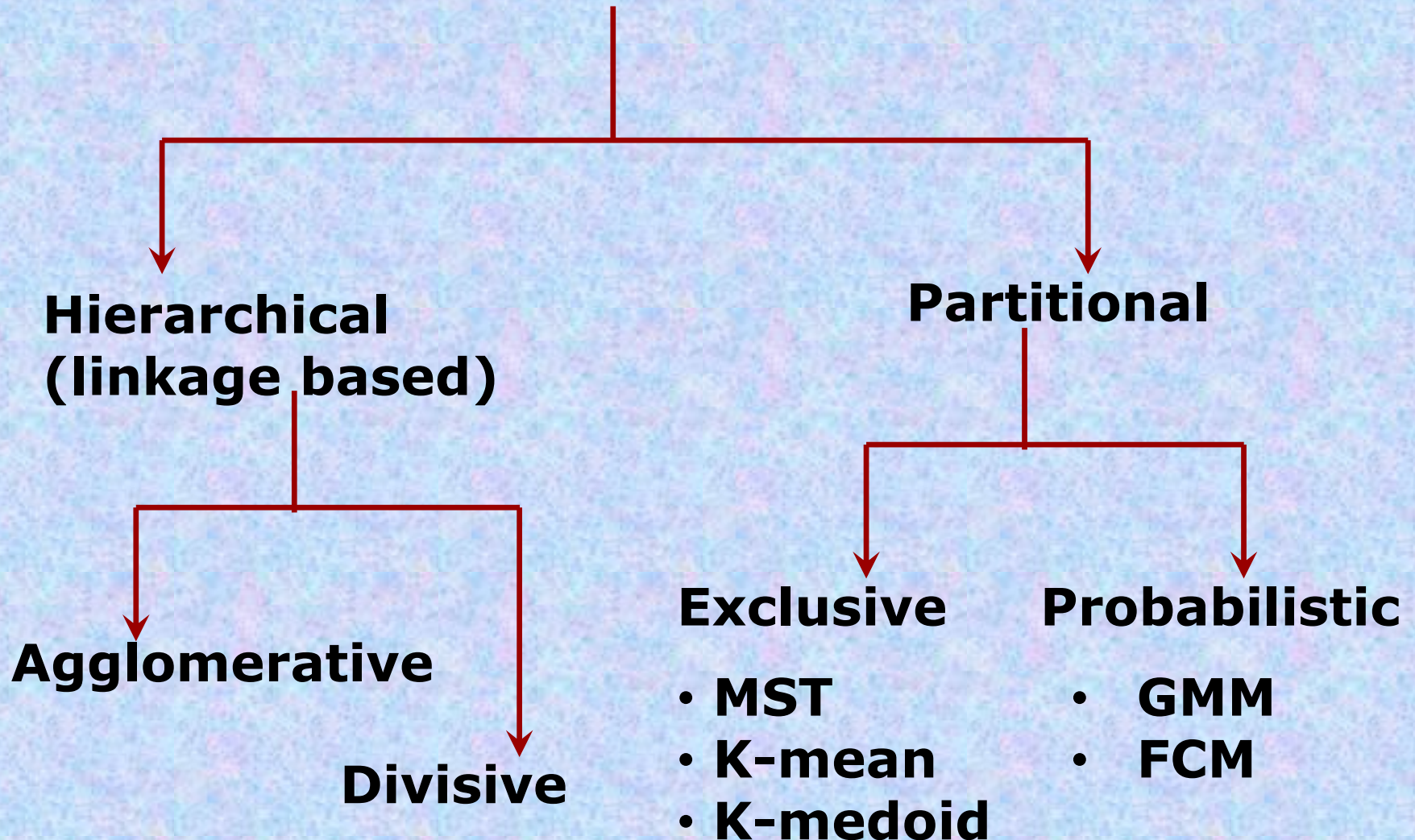
- **Partitioning approach:**
  - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square errors
  - Typical methods: k-means, k-medoids, CLARANS
- **Hierarchical approach:**
  - Create a hierarchical decomposition of the set of data (or objects) using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

# Major Clustering Approaches (II)

---

- **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: p-Cluster
- **User-guided or constraint-based:**
  - Clustering by considering user-specified or application-specific constraints
  - Typical methods: COD (obstacles), constrained clustering
- **Link-based clustering:**
  - Objects are often linked together in various ways
  - Massive links can be used to cluster objects: SimRank, LinkClus

# **GENERAL CATEGORIES** **of CLUSTERING DATA**



## **Alternative view of Algorithms for CLUSTERING**

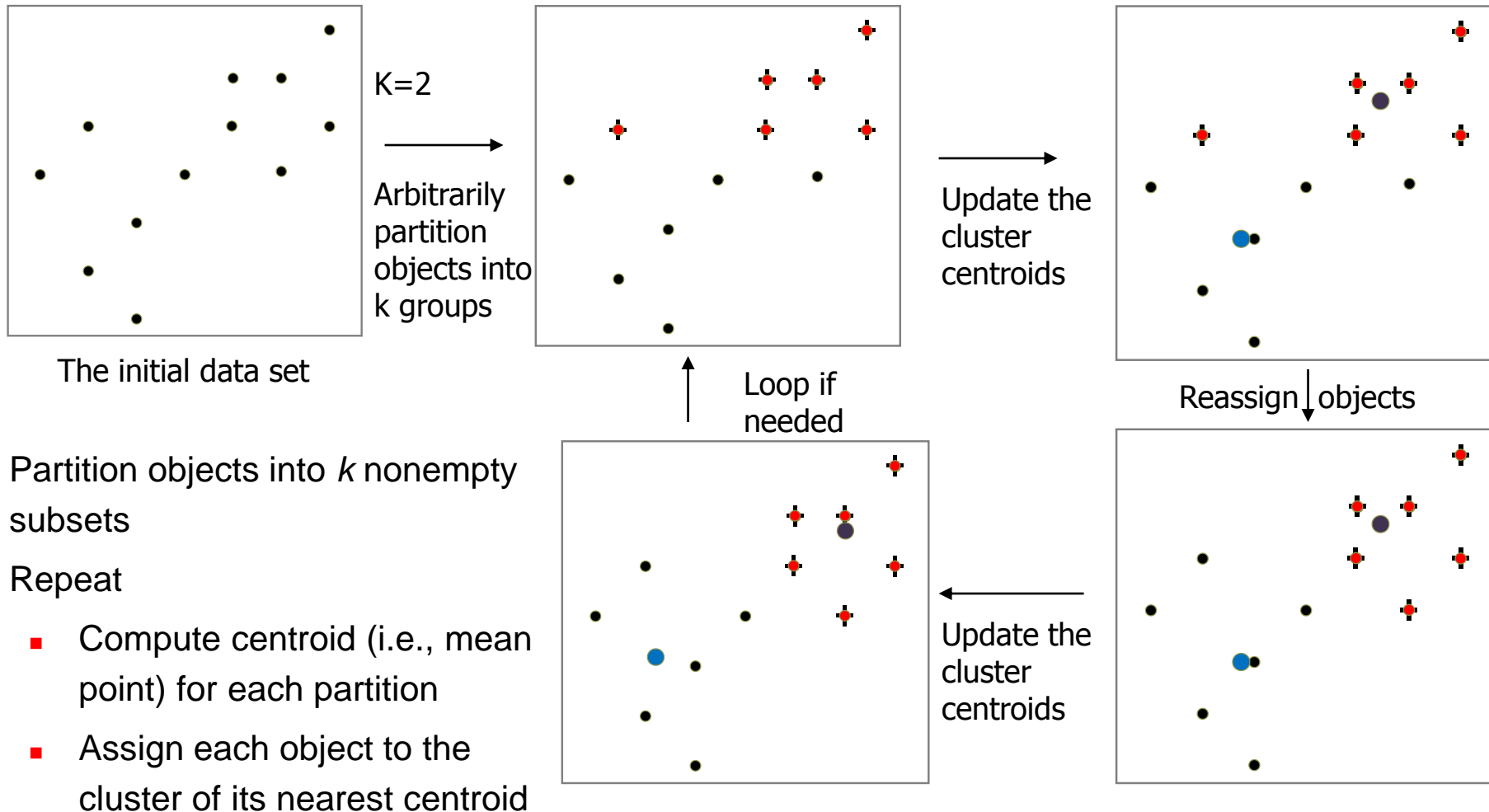
- **Unsupervised Learning/Classification:**
  - **K-means; K-medoid**
- **Density Estimation :**
  - (i) **Parametric**
    - **Gaussian**
    - **MOG (Mixture of Gaussians)**
    - **Dirichlet, Beta etc.**
    - **Branch and Bound Procedure**
    - **Piecewise Quadratic Boundary**
    - **Nearest Mean Classifier**
    - **MLE (maximum Likelihood Estimate)**

- **Density Estimation :**
  - (ii) **Non-Parametric**

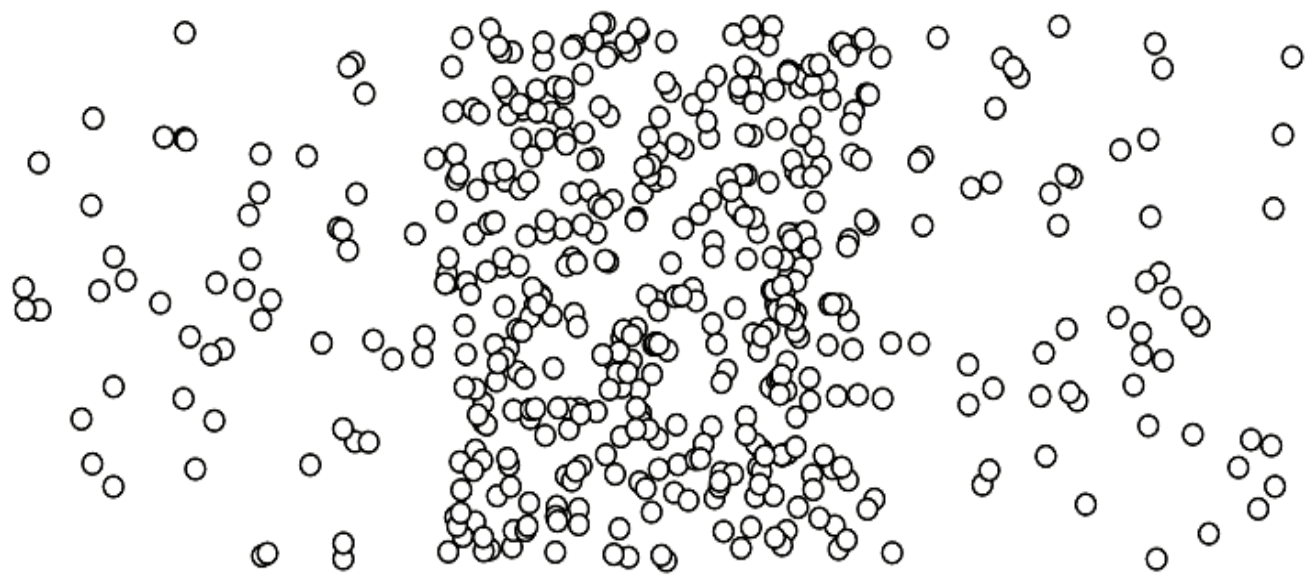
- **Histogram**
- **Neighborhood**
- **Kernel Methods**
- **Graph Theoretic**
- **Iterative Valley Seeking**



# An Example of *K-Means* Clustering



- Partition objects into  $k$  nonempty subsets
- Repeat
  - Compute centroid (i.e., mean point) for each partition
  - Assign each object to the cluster of its nearest centroid
- Until no change



# FCM - Fuzzy C-Means Clustering

# FCM

- A method of clustering which allows one piece of data to belong to two or more clusters.
- Objective function to be minimized:

$$J_m = \sum_{i=1}^N \sum_{j=1}^c u_{ij}^m \|x_i - \mu_j\|^2, \quad 1 \leq m < \infty$$

Where

- $u_{ij}$  is the degree of membership of  $x_j$  in the cluster  $j$ .
- $x_i$  is d-dimensional observation
- $\mu_j$  is d-dimensional center of cluster  $j$

# Update

- FCM is an iterative optimization approach.
- At each step, the membership  $u_{ij}$  and the cluster centers  $\mu_j$  are updated as follows:

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

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

# Termination Criterion

- Iteration stops, when

$$\max_{ij} \left\{ \left| u_{ij}^{(k+1)} - u_{ij}^{(k)} \right| \right\} < \epsilon$$

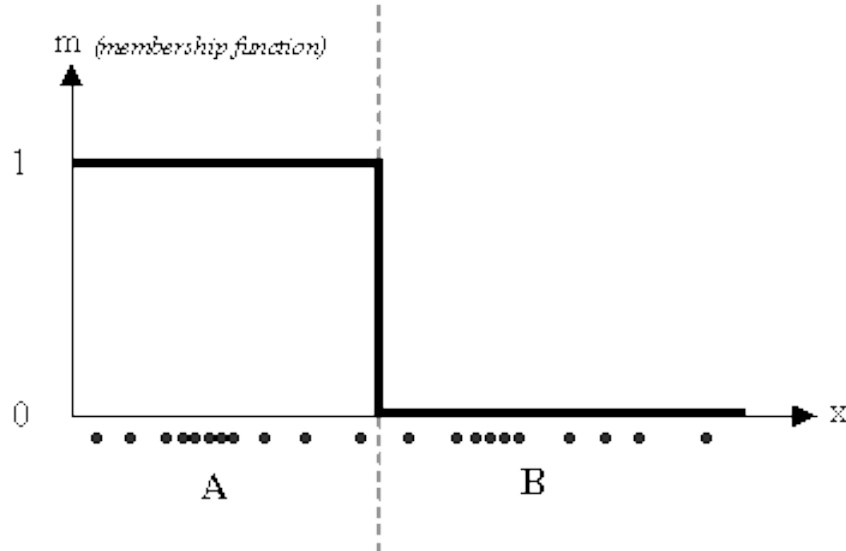
Where  $k$  is the iteration number.

$\epsilon$  is between 0 and 1

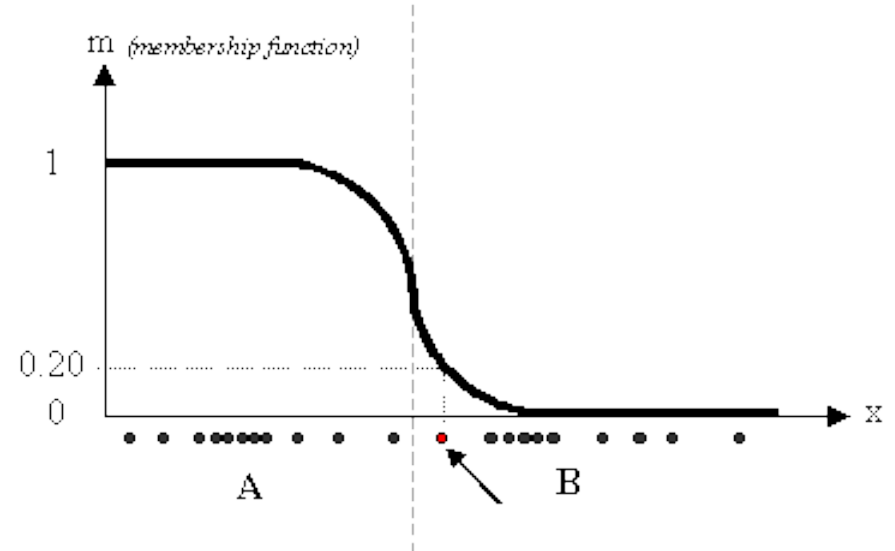


# K-means Vs FCM

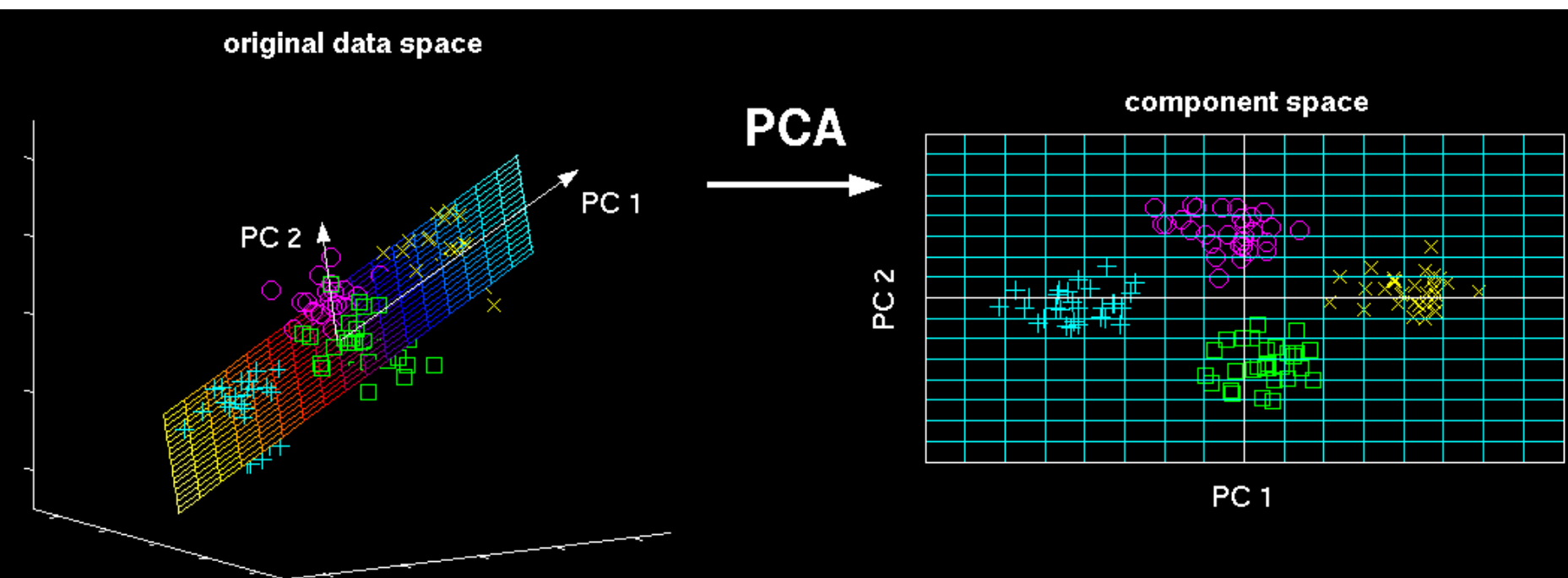
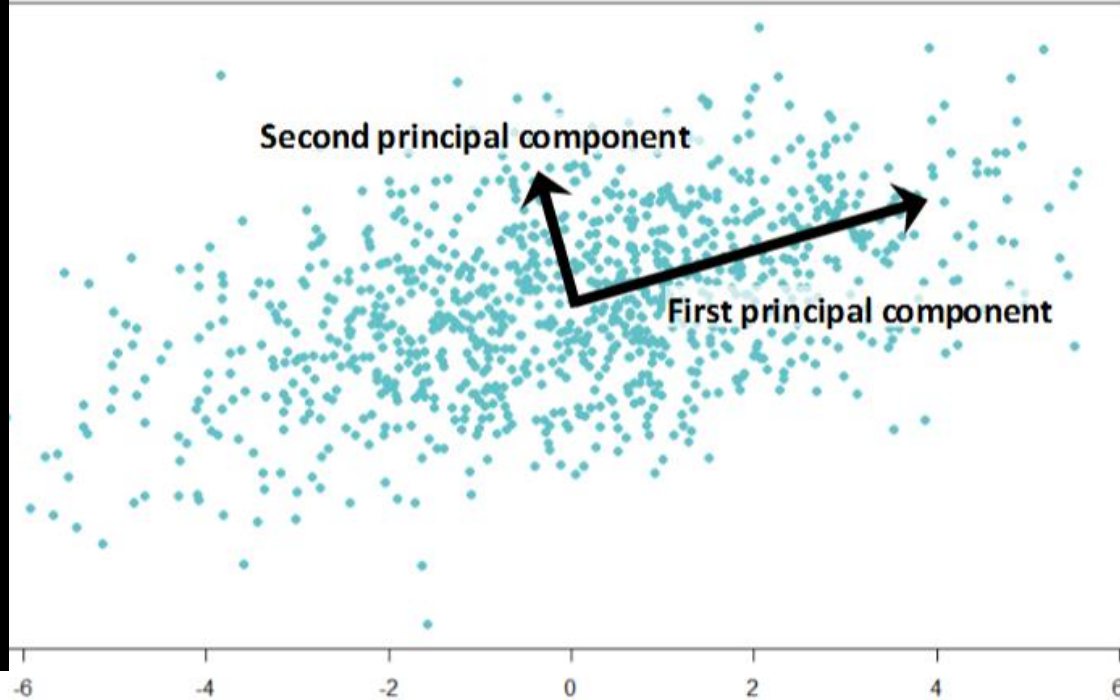
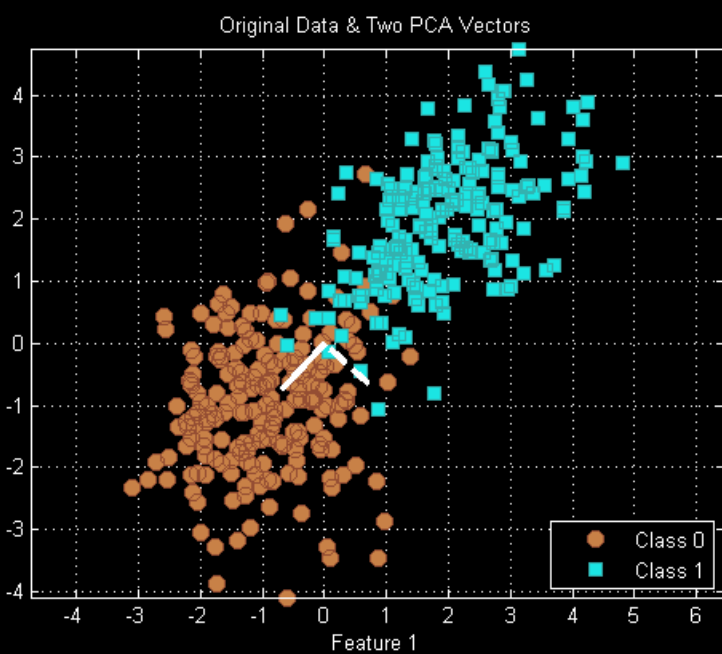
K-means



FCM



**Read about K-medoids**



# Hierarchical Clustering

# Hierarchical Clustering

- Builds hierarchy of clusters
- Types:
  - Bottom Up - *Agglomerative*
    - *Starts by considering each observation as a cluster of it's own*
    - *Clusters are merged as we move up the hierarchy*
  - Top Down - *Divisive*
    - *Starts by considering all observations in one cluster*
    - *Clusters are divided as we move down the hierarchy*

# Distance Functions

Certain mathematical properties are expected of any distance measure, or *metric*:

1.  $d(x, y) \geq 0$  for all  $x, y$ .
2.  $d(x, y) = 0$  iff  $x = y$ .
3.  $d(x, y) = d(y, x)$  (symmetry)
4.  $d(x, y) \leq d(x, z) + d(z, y)$  for all  $x, y$ , and  $z$ . (triangle inequality)

*Euclidean distance*  $d(x, y) = \sqrt{\sum_{i=1}^d |x_i - y_i|^2}$  is probably the most commonly used metric. Note that it weights all features/dimensions “equally”.

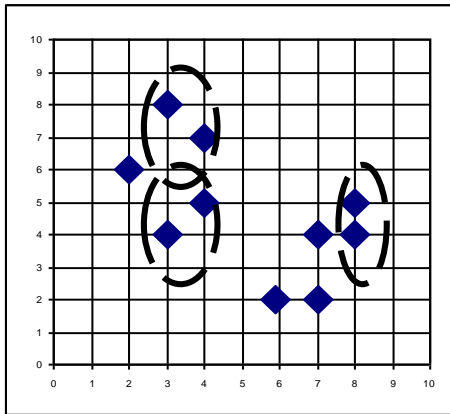
# Some commonly used Metrics

- Euclidean distance
- Squared Euclidean distance
- Manhattan distance
- Maximum distance
- Mahalanobis distance



# Agglomerative clustering

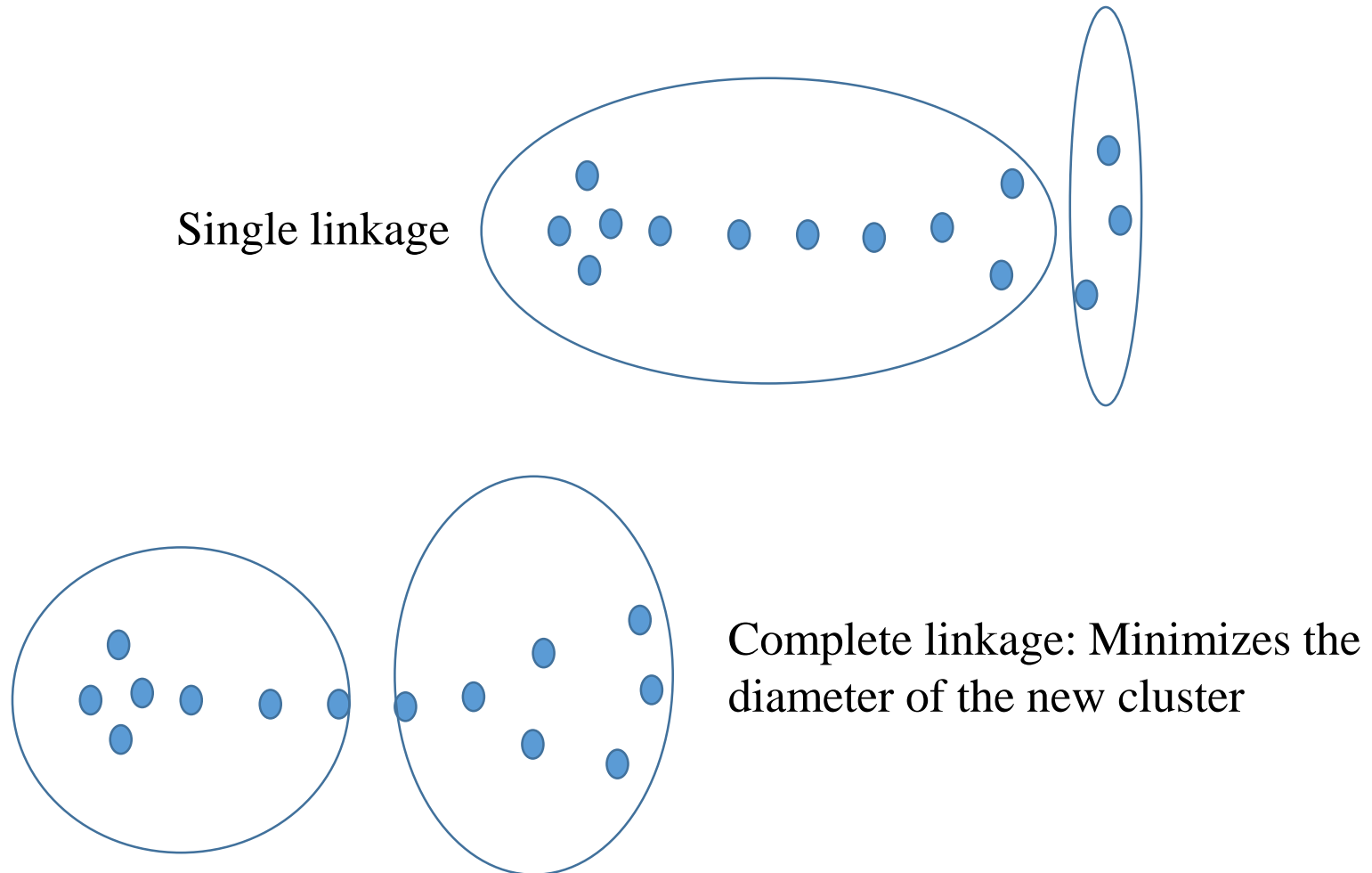
- Each node/object is a cluster initially
- Merge clusters that have the **least** dissimilarity
  - Ex: single-linkage, complete-linkage, etc.
- Go on in a non-descending fashion
- Eventually, all nodes belong to the same cluster



# Linkage Criteria

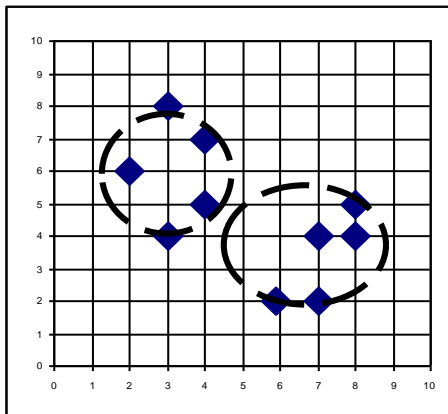
- Determines the distance between sets of observations as a function of the pairwise distances between observations.
- Some commonly used criterias:
  - *Single Linkage*: Distance between two clusters is the **smallest** pairwise distance between two observations/nodes, each belonging to different clusters.
  - *Complete Linkage*: Distance between two clusters is the **largest** pairwise distance between two observations/nodes, each belonging to different clusters.
  - *Mean or average linkage clustering*: Distance between two clusters is the **average** of all the pairwise distances, each node/observation belonging to different clusters.
  - *Centroid linkage clustering*: Distance between two clusters is the **distance between their centroids**.

# Single Linkage vs. Complete Linkage



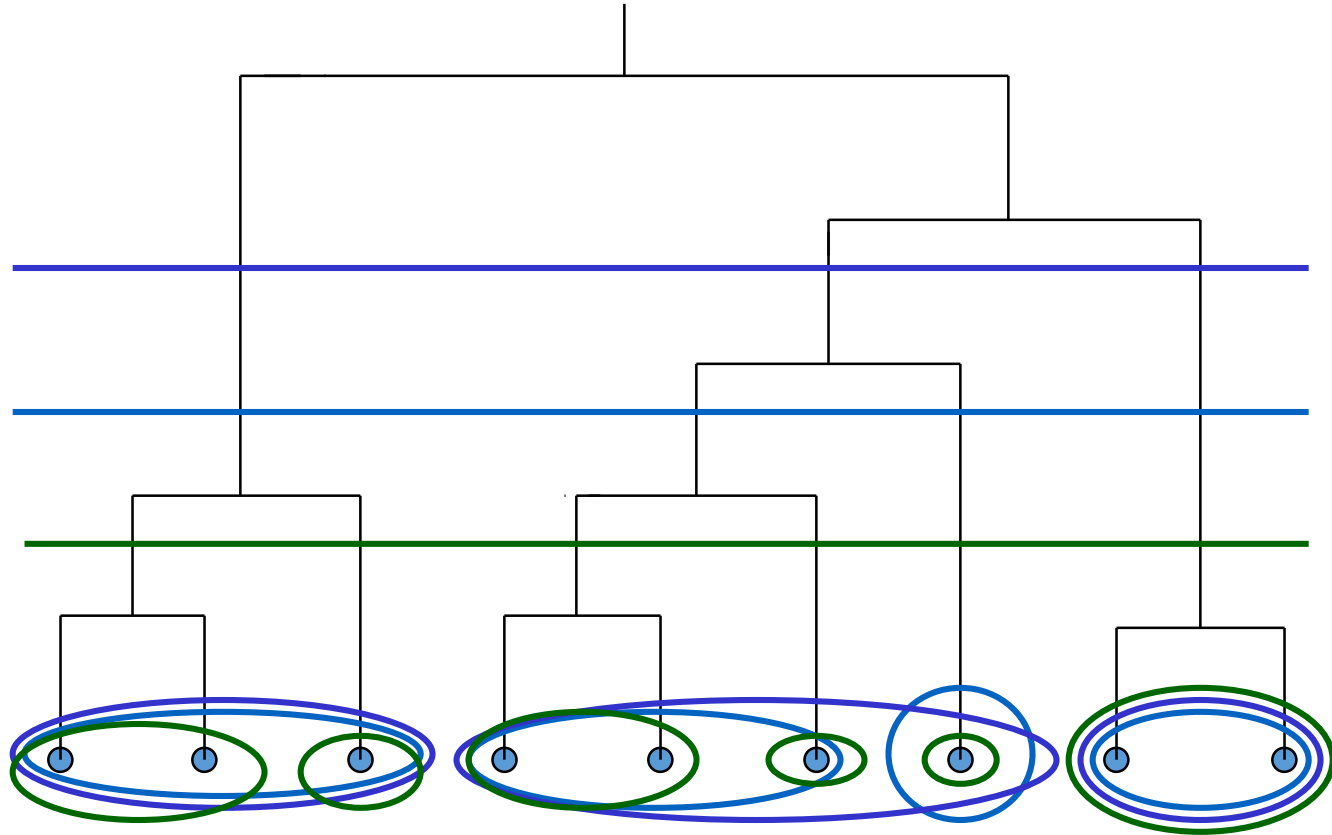
# Divisive Clustering

- Initially, all data is in the same cluster
- The largest cluster is split until every object is separate.



# What are the true number of clusters?

- Decompose data objects into a several levels of nested partitioning (tree of clusters), called a dendrogram.
- A clustering of the data objects is obtained by cutting the dendrogram at the desired level, then each connected component forms a cluster.



# DBSCAN : Density Based Spatial Clustering of Applications with Noise

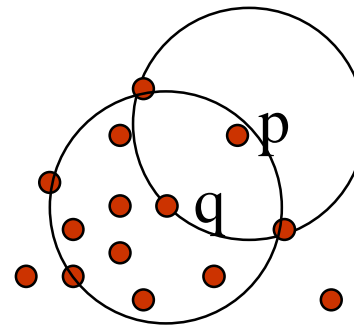


# Density-Based Clustering Methods

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

# Density-Based Clustering: Basic Concepts

- Two parameters:
  - *Eps*: Maximum radius of the neighborhood
  - *MinPts*: Minimum number of points in an *Eps*-neighborhood of that point
- $N_{Eps}(p)$ :  $\{q \text{ belongs to } D \mid \text{dist}(p, q) \leq Eps\}$
- **Directly density-reachable**: A point  $p$  is directly density-reachable from a point  $q$  w.r.t. *Eps*, *MinPts* if
  - $p$  belongs to  $N_{Eps}(q)$
  - core point condition:
$$|N_{Eps}(q)| \geq MinPts$$

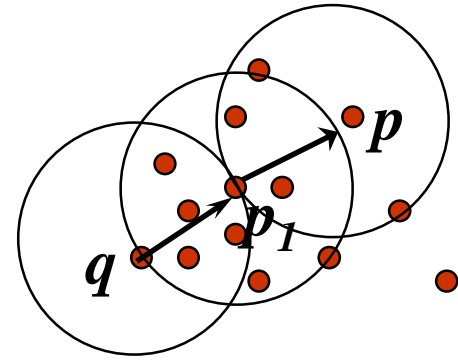


MinPts = 5  
Eps = 1 cm

# Density-reachable & Density-connected

- Density-reachable:

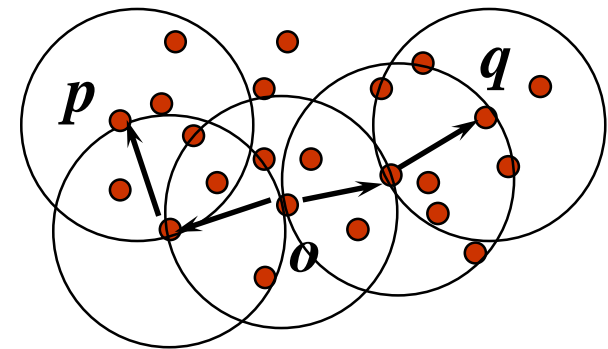
- A point  $p$  is **density-reachable** from a point  $q$  if there is a chain of points  $p_1, \dots, p_n, p_1 = q, p_n = p$  such that  $p_{i+1}$  is directly density-reachable from  $p_i$



- This is not symmetric

- Density-connected

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



# DBSCAN

- A set of points  $C$  is a cluster, if
  - For any two points  $p, q \in C$ ,  $p$  and  $q$  are density-connected
  - There does not exist any pair of points,  $p \in C$  and  $s \notin C$  such that  $p$  and  $s$  are density-connected.

Border points are points that are reachable from any of the core points. For a border point  $p$

$$|N_{Eps}(p)| < MinPts$$

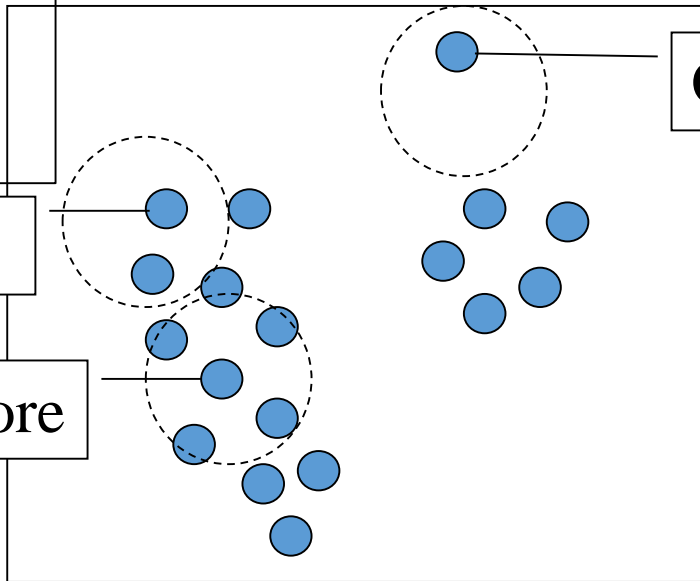
Border

Core

Outlier

Eps = 1cm

MinPts = 5



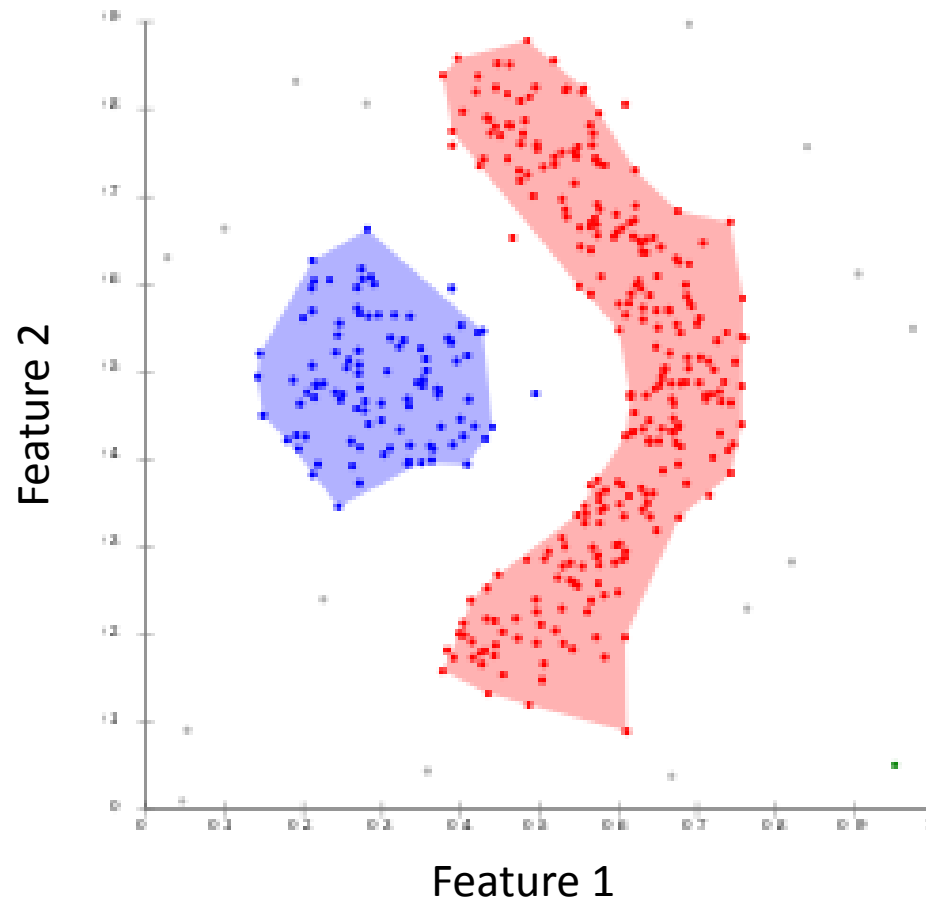
# Algorithm

- Select a point  $p$
- Retrieve all points directly density-reachable from  $p$  wrt.  $Eps$  and  $MinPts$ .
- If  $p$  is not a core point,  $p$  is marked as noise
- Else a cluster is initiated.
  - $p$  is marked as classified with a cluster ID
  - $seedSet =$  all directly reachable points from  $p$ .
  - For each point  $p_i$  in  $seedSet$  till it is empty
    - If  $p_i$  is a noise point, assign  $p_i$  to the current cluster ID
    - If  $p_i$  is unclassified, identify if it is a core point. If yes, then add all directly reachable point to seed set and add  $p_i$  to cluster ID
    - Delete  $p_i$  from  $seedSet$

# DBSCAN: Properties

- Can discover clusters of arbitrary shapes
- Complexity
  - Time
    - $O(n^2)$
    - $O(n \log^{d-1} n)$  with range tree. But requires more storage
      - $d$  dimensions
- Weakness:
  - Parameter sensitive

# DBSCAN - non-linearly separable clusters



# How to pick the initial centroids?

I'll Choose

Randomly

Farthest Point

## What kind of data would you like

Uniform Points

Gaussian Mixture

Smiley Face

Density Bars

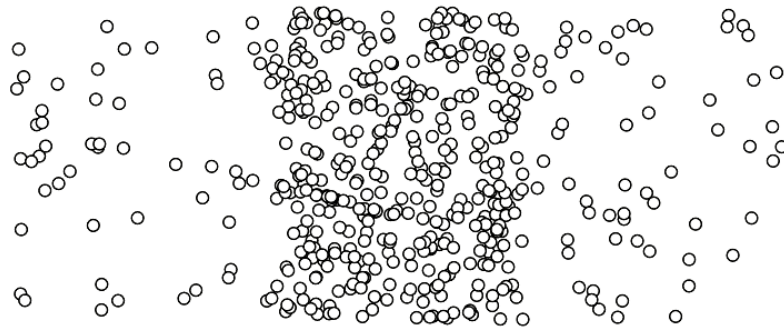
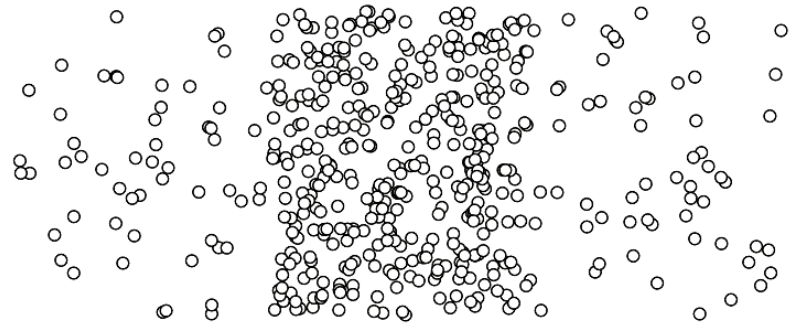
Packed Circles

Pimpled Smiley

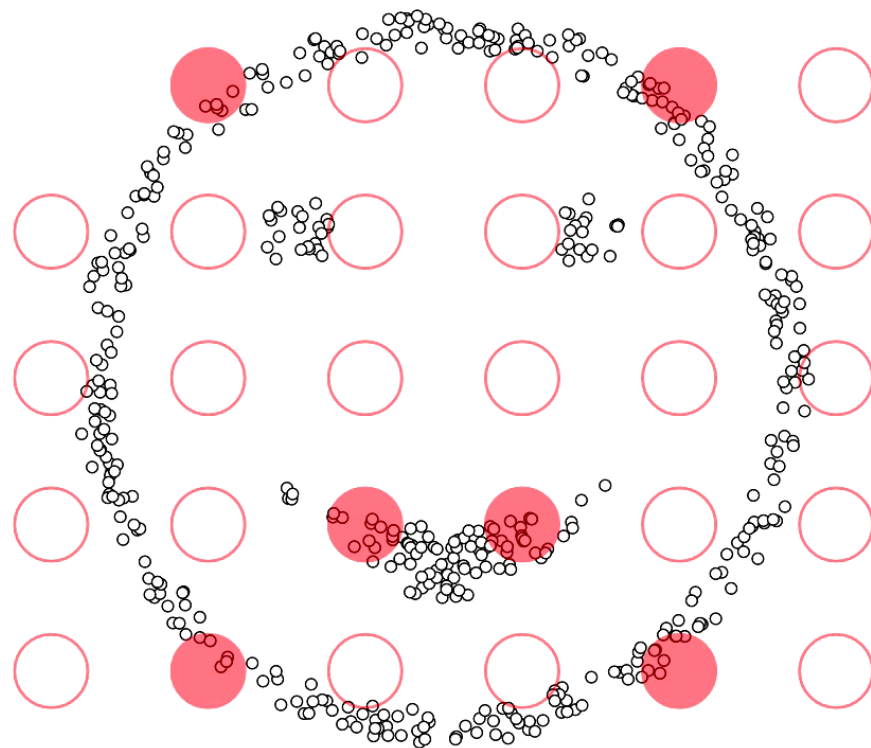
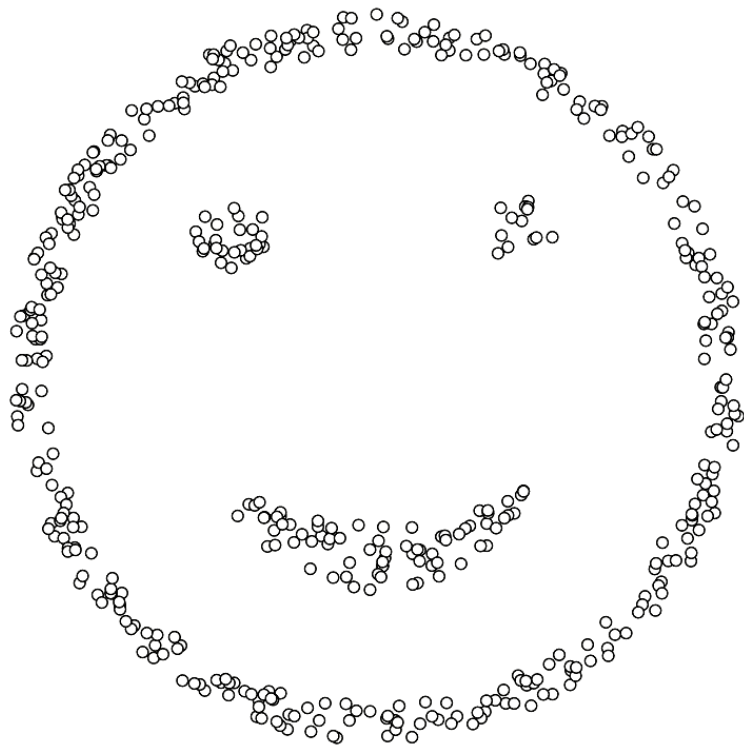
DBSCAN Rings

Example A

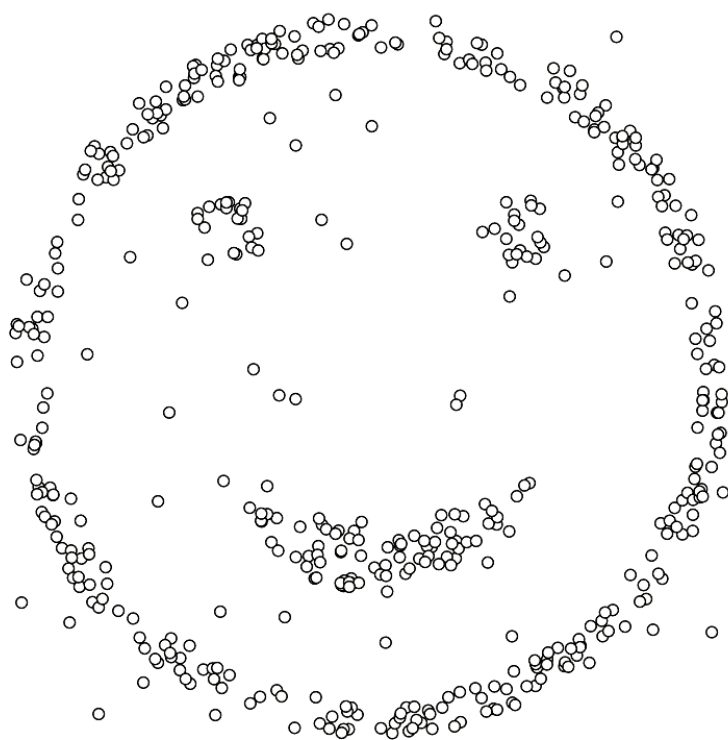




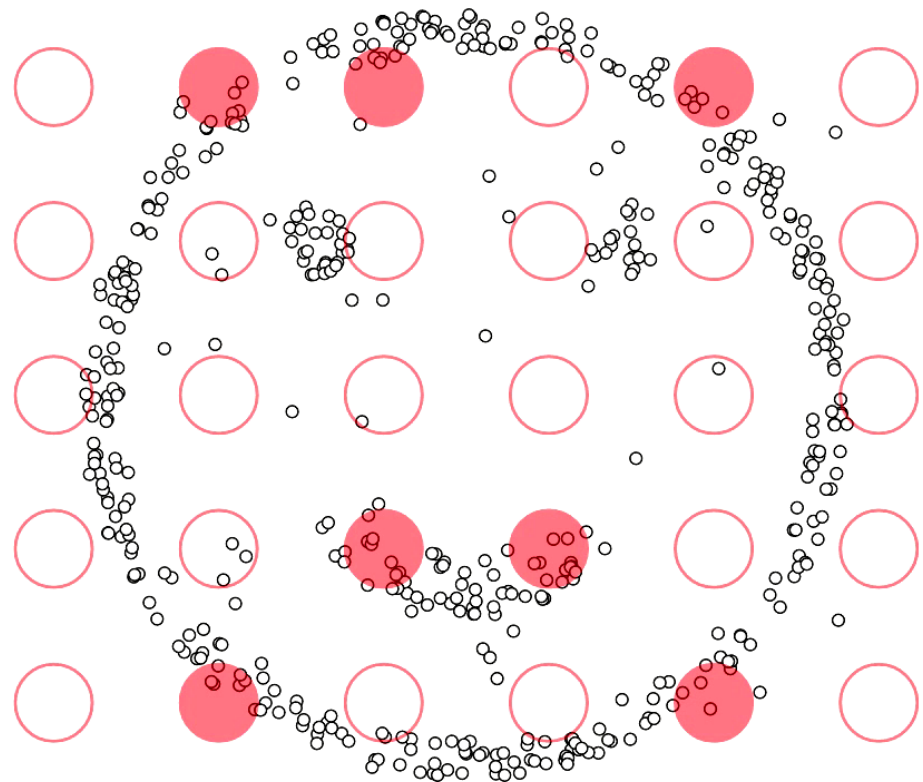
epsilon = 1.00  
minPoints = 4

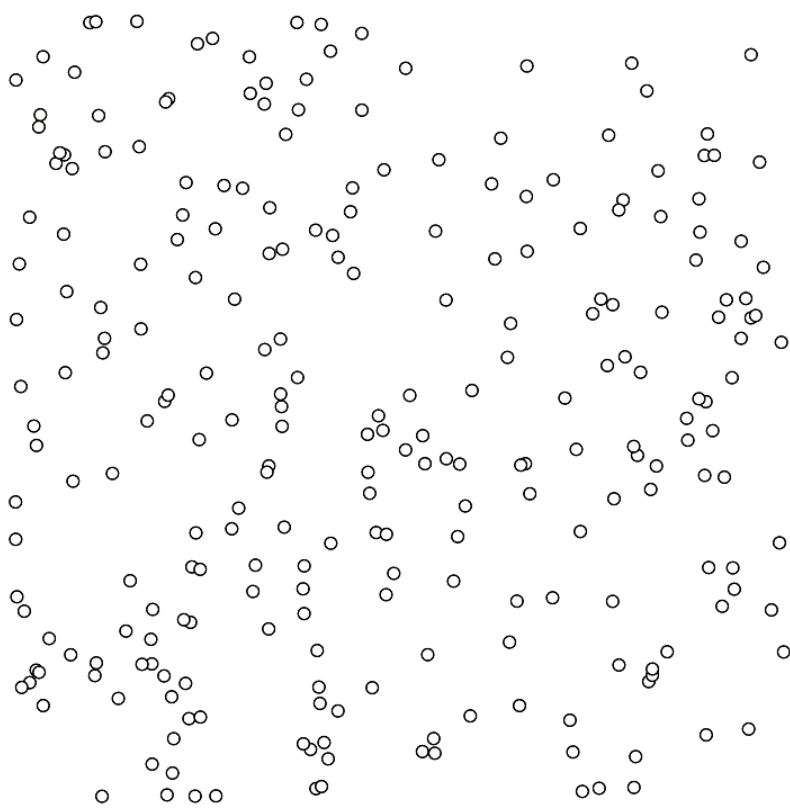


epsilon = 1.00  
minPoints = 4



epsilon = 1.00  
minPoints = 4





at kind of data would you like?

Uniform Points

Gaussian Mixture

Smiley Face

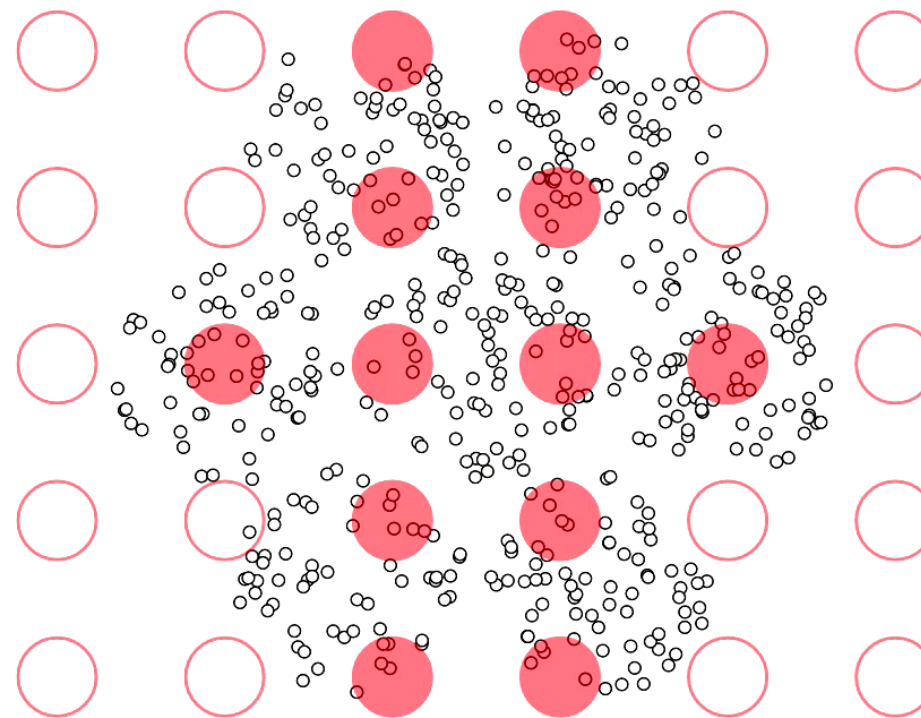
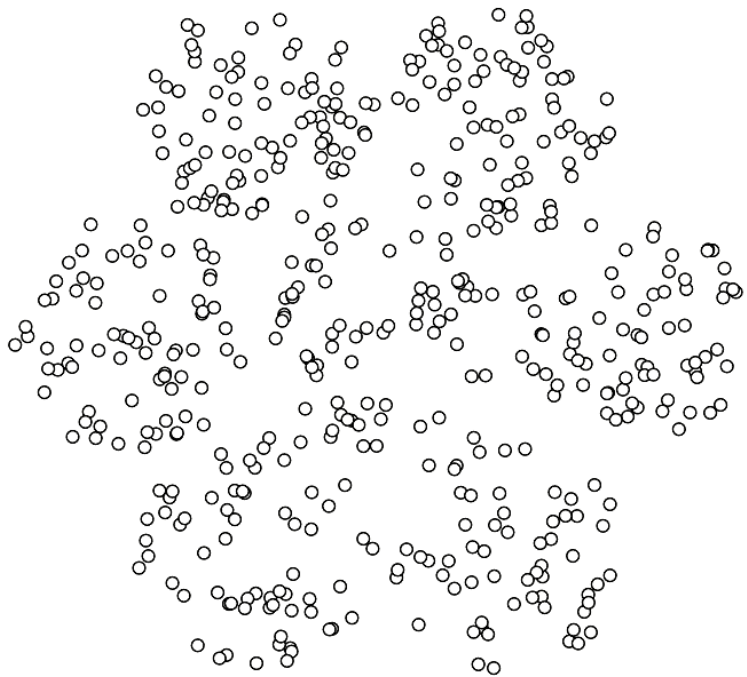
Density Bars

Packed Circles

Pimpled Smiley

DBSCAN Rings

Example A



epsilon = 1.00  
minPoints = 4

## **REFERENCES**

**K. Fukunaga; Introduction to Statistical Pattern Recognition, Second Edition, Academic Press, Morgan Kaufmann, 1990.**

**Richard O. Duda, Peter E. Hart, and David G. Stork. "Pattern Classification." Wiley, 1973.**

**Christopher M. Bishop, "Pattern Recognition and Machine Learning." Springer, 2006.**

**M. R. Anderberg. Cluster Analysis for Applications. Academic Press, 1973.**

**A. K. Jain and R. C. Dubes. Algorithms for Clustering Data. Prentice Hall, 1988.**

**L. Kaufman and P. J. Rousseeuw. Finding Groups in Data: an Introduction to Cluster Analysis. John Wiley & Sons, 1990.**

**G. J. McLachlan and K.E. Bkasford. Mixture Models: Inference and Applications to Clustering. John Wiley and Sons, 1988.**

# Demo

## Visualizing DBSCAN Clustering

Link: <https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/>

# References (1)

---

- R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. SIGMOD'98
- M. R. Anderberg. Cluster Analysis for Applications. Academic Press, 1973.
- M. Ankerst, M. Breunig, H.-P. Kriegel, and J. Sander. Optics: Ordering points to identify the clustering structure, SIGMOD'99.
- Beil F., Ester M., Xu X.: "Frequent Term-Based Text Clustering", KDD'02
- M. M. Breunig, H.-P. Kriegel, R. Ng, J. Sander. LOF: Identifying Density-Based Local Outliers. SIGMOD 2000.
- M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases. KDD'96.
- M. Ester, H.-P. Kriegel, and X. Xu. Knowledge discovery in large spatial databases: Focusing techniques for efficient class identification. SSD'95.
- D. Fisher. Knowledge acquisition via incremental conceptual clustering. Machine Learning, 2:139-172, 1987.
- D. Gibson, J. Kleinberg, and P. Raghavan. Clustering categorical data: An approach based on dynamic systems. VLDB'98.
- V. Ganti, J. Gehrke, R. Ramakrishan. CACTUS Clustering Categorical Data Using Summaries. KDD'99.



# References (2)

---

- D. Gibson, J. Kleinberg, and P. Raghavan. Clustering categorical data: An approach based on dynamic systems. In Proc. VLDB'98.
- S. Guha, R. Rastogi, and K. Shim. Cure: An efficient clustering algorithm for large databases. SIGMOD'98.
- S. Guha, R. Rastogi, and K. Shim. ROCK: A robust clustering algorithm for categorical attributes. In *ICDE'99*, pp. 512-521, Sydney, Australia, March 1999.
- A. Hinneburg, D. I. A. Keim: An Efficient Approach to Clustering in Large Multimedia Databases with Noise. KDD'98.
- A. K. Jain and R. C. Dubes. Algorithms for Clustering Data. Printice Hall, 1988.
- G. Karypis, E.-H. Han, and V. Kumar. CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling. *COMPUTER*, 32(8): 68-75, 1999.
- L. Kaufman and P. J. Rousseeuw. Finding Groups in Data: an Introduction to Cluster Analysis. John Wiley & Sons, 1990.
- E. Knorr and R. Ng. Algorithms for mining distance-based outliers in large datasets. VLDB'98.

# References (3)

- G. J. McLachlan and K.E. Bkasford. Mixture Models: Inference and Applications to Clustering. John Wiley and Sons, 1988.
- R. Ng and J. Han. Efficient and effective clustering method for spatial data mining. VLDB'94.
- L. Parsons, E. Haque and H. Liu, Subspace Clustering for High Dimensional Data: A Review, SIGKDD Explorations, 6(1), June 2004
- E. Schikuta. Grid clustering: An efficient hierarchical clustering method for very large data sets. Proc. 1996 Int. Conf. on Pattern Recognition,.
- G. Sheikholeslami, S. Chatterjee, and A. Zhang. WaveCluster: A multi-resolution clustering approach for very large spatial databases. VLDB'98.
- A. K. H. Tung, J. Han, L. V. S. Lakshmanan, and R. T. Ng. Constraint-Based Clustering in Large Databases, ICDT'01.
- A. K. H. Tung, J. Hou, and J. Han. Spatial Clustering in the Presence of Obstacles, ICDE'01
- H. Wang, W. Wang, J. Yang, and P.S. Yu. Clustering by pattern similarity in large data sets, SIGMOD' 02.
- W. Wang, Yang, R. Muntz, STING: A Statistical Information grid Approach to Spatial Data Mining, VLDB'97.
- T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH : An efficient data clustering method for very large databases. SIGMOD'96.
- Xiaoxin Yin, Jiawei Han, and Philip Yu, "[LinkClus: Efficient Clustering via Heterogeneous Semantic Links](#)", in Proc. 2006 Int. Conf. on Very Large Data Bases (VLDB'06), Seoul, Korea, Sept. 2006.