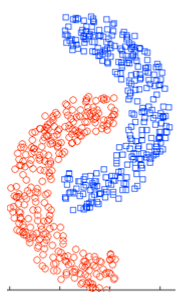
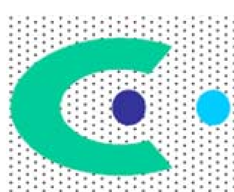
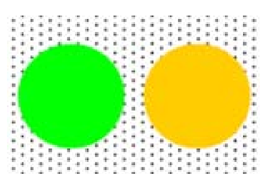
Outline

- What is Clustering?
- Types of Data in Cluster Analysis and Similarity Measures
- Some clustering Methods
  - K-means
  - K-medoids
  - Hierarchical clustering method
  - DBSCAN: a Density-based Algorithm
- Cluster Validity

Density-based Clustering

- 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.

Density-based Clustering

- Clustering based on density (local cluster criterion), such as density-connected points
- Major features:
  - Discover clusters of arbitrary shape
  - Handle noise
  - Do not need to specify k, but need density parameters
- Several interesting studies:
  - DBSCAN: Ester, et al. (KDD'96)
  - OPTICS: Ankerst, et al. (SIGMOD'99)
  - DENCLUE: Hinneburg & D. Kein (KDD'98)
  - CLIQUE: Agrawal, et al. (SIGMOD'98)

DBSCAN: Basic Concepts

- ***Eps***-neighborhood of point *p* in data set *D*:

  $N_{Eps}(p)=\{q \in D \mid dist(p,q) <= Eps\}$

  where ***Eps*** is called the radius of the neighborhood

- Density of ***Eps***-neighborhood of *p*:

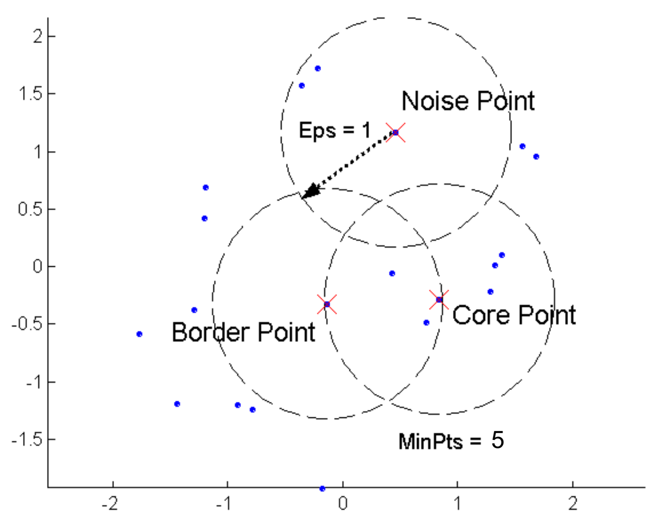
  the number of points in $N_{Eps}(p)$

DBSCAN: Basic Concepts

- A point is a **core point** if there are at least ***MinPts*** number of points in its ***Eps***-neighborhood.
  - These are points that are at the interior of a cluster
- A **border point** has fewer than ***MinPts*** points in its *Eps-neighborhood*, but is in the *Eps*-neighborhood of a core point.
  - These are points that are on or close to the border of a cluster
- A **noise point** is any point that is not a core point or a border point.
  - These are points that are outsider any cluster

44

DBSCAN: Core, Border and Noise Points



45

DBSCAN: Input Parameters

- ***Eps***:

  Maximum radius of the neighbourhood

- ***MinPts***:

  Minimum number of points in an *Eps*-neighbourhood of a core point

46

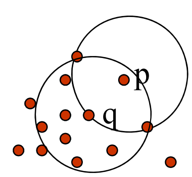
DBSCAN: Basic Concepts

- Directly density-reachable**:**
  - A point ***p*** is ***directly density-reachable*** from a point ***q*** wrt. ***Eps*** and ***MinPts*** if

    1) ***p*** belongs to $N_{Eps}(q)$

    2) ***q*** is a core point, that is:

    $$|N_{Eps}(q)| >= MinPts$$



MinPts = 5

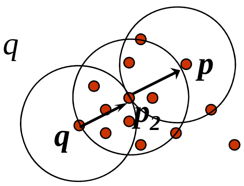
Eps = 1 cm

  - Asymmetric in general
    - Symmetric only when both *p* and *q* are core points.

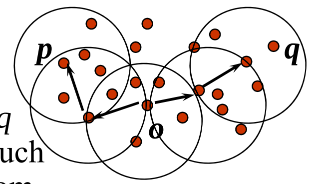
47

DBSCAN: Basic Concepts

- Density-reachable:
  - A point p is density-reachable from a point q wrt. *Eps* and *MinPts* if there is a chain of points $p_1, p_2, \ldots, p_n$, $p_1 = q$, $p_n = p$ such that p_{i+1} is directly density-reachable from p_i
  - Symmetric only when p and q are both core points



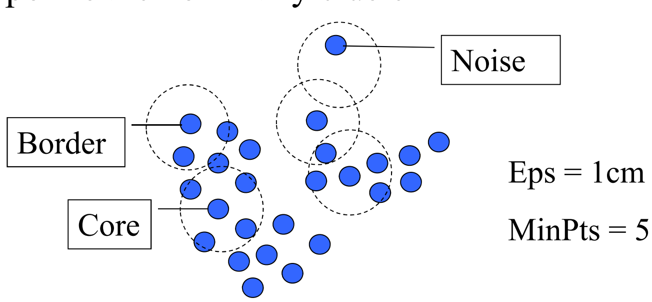
- Density-connected:
  - A point p is density-connected to a point q wrt. *Eps* and *MinPts* if there is a point o such that both p and q are density-reachable from o wrt. *Eps* and *MinPts*.
  - Symmetric



48

DBSCAN: Cluster

- *Density-based* notion of cluster: A *cluster* is defined as a maximal set of density-connected points
  - Each pair of points in a cluster are density-connected to each other (Connectivity)
  - Core points in other clusters are not density-connected to any core points in this cluster (Maximality)
- Noise points are not in any cluster



49

DBSCAN: The Algorithm

- Arbitrarily select an unprocessed point ***p***
- If ***p*** is a core point,

These points and p are in a cluster because they are density-connected through *p*.

  - Retrieve all points density-reachable from ***p*** wrt ***Eps*** and ***MinPts***.
  - A cluster is formed which includes all the points density-reachable from ***p***
  - Mark all the points in the cluster as "processed"
- If ***p*** is not a core point, no points are density-reachable from ***p*** and DBSCAN visits the next unprocessed point of the database.
- Continue the process until all of the core points have been processed.

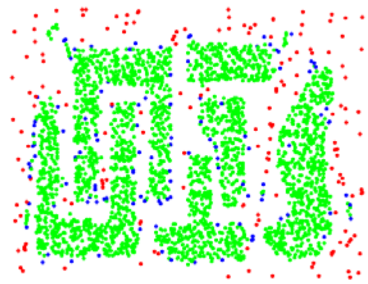
Note: Membership of a border point depends on the order of the points being processed if it is density-connected to core points in two or more clusters.

Another Version of DBSCAN (for simplicity, not efficiency)

- Label all points as core, border or noise points
- Eliminate noise points
- Put an edge between all core points that are within *Eps* of each other
- Make each group of connected core points into a separate cluster
- Assign each border point to one of the clusters of its associated core points.

DBSCAN: Core, Border and Noise Points

Original Points

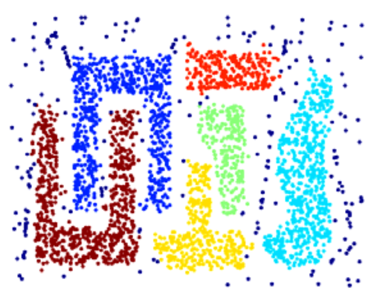


Point types: core, border and noise

52

When DBSCAN Works Well

Original Points

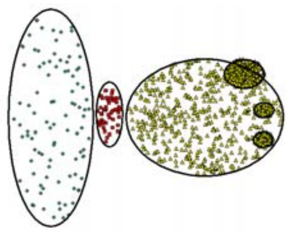


Clusters

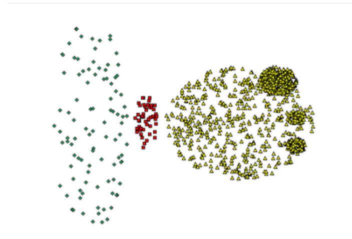
- **Resistant to Noise**
- **Can handle clusters of different shapes and sizes**

53

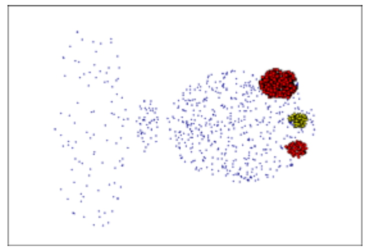
When DBSCAN Does NOT Work Well



Original Points



(MinPts=4, Eps=9.75).



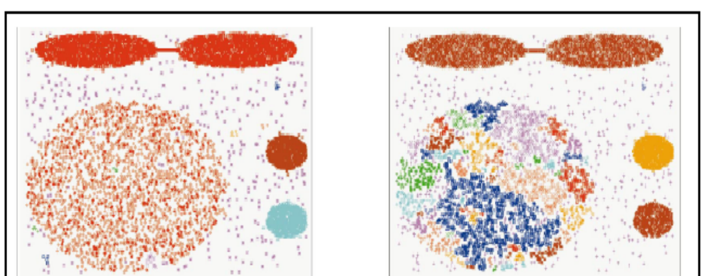
(MinPts=4, Eps=9.92)

• **Varying densities**

54

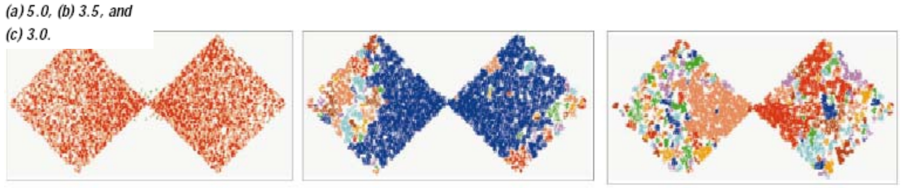
DBSCAN: Sensitive to Parameters

Figure 8. DBScan results for DS1 with MinPts at 4 and Eps at (a) 0.5 and (b) 0.4.



(a) (b)

Figure 9. DBScan results for DS2 with MinPts at 4 and Eps at (a) 5.0, (b) 3.5, and (c) 3.0.

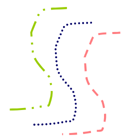


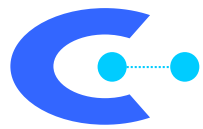
(a) (b) (c)

55

Comparing DBSCAN and K-means

- Cluster shapes
  - K-means: spheres
  - DBSCAN: arbitrary shape



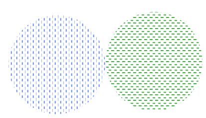


- Cluster sizes
  - K-means may have a problem when clusters are of different sizes
  - DBSCAN can handle clusters of different sizes
- Noise and outliers
  - K-means is sensitive to noise or outliers
  - DBSCAN is not strongly affected by noise or outliers

56

Comparing DBSCAN and K-means (Cont’d)

- Not well-separated clusters



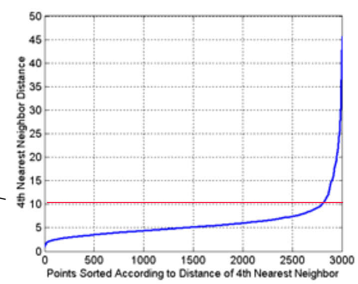
  - K-means can find clusters that are not well separated, even if they overlap
  - DBSCAN merges clusters that overlap
- Stability
  - K-means’ result depends on the random initialization of centroids
  - DBSCAN produces the same set of clusters from one run to another (except that the membership of some border points depends on the order of the points being processed)
- Number of clusters
  - For k-means, the number of clusters needs to be specified as a parameter
  - DBSCAN automatically determines the number of clusters. However, it has two other parameters: *Eps* and *MinPts*

57

DBSCAN: Determining Eps and MinPts

- Idea is that for points in a cluster, their k^{th} nearest neighbors are at roughly the same distance
- Noise points have the k^{th} nearest neighbor at farther distance
- So, plot sorted distance of every point to its k^{th} nearest neighbor (e.g., k=4)

Thus, Eps=10 for MinPts=4



58

Outline

- What is Clustering?
- Types of Data in Cluster Analysis and Similarity Measures
- Some clustering Methods
  - K-means
  - K-medoids
  - Hierarchical clustering method
  - DBSCAN: a Density-based Algorithm
- Cluster Validity

59

Cluster Validity

- For supervised classification we have a variety of measures to evaluate how good our model is
  - Accuracy, error rate, confusion matrix, misclassification cost
- For cluster analysis, the analogous question is how to evaluate the "goodness" of the resulting clusters?
- But "clusters are in the eye of the beholder"!
- Then why do we want to evaluate them?
  - To compare clustering algorithms
  - To compare two sets of clusters
  - To determine the 'correct' number of clusters.

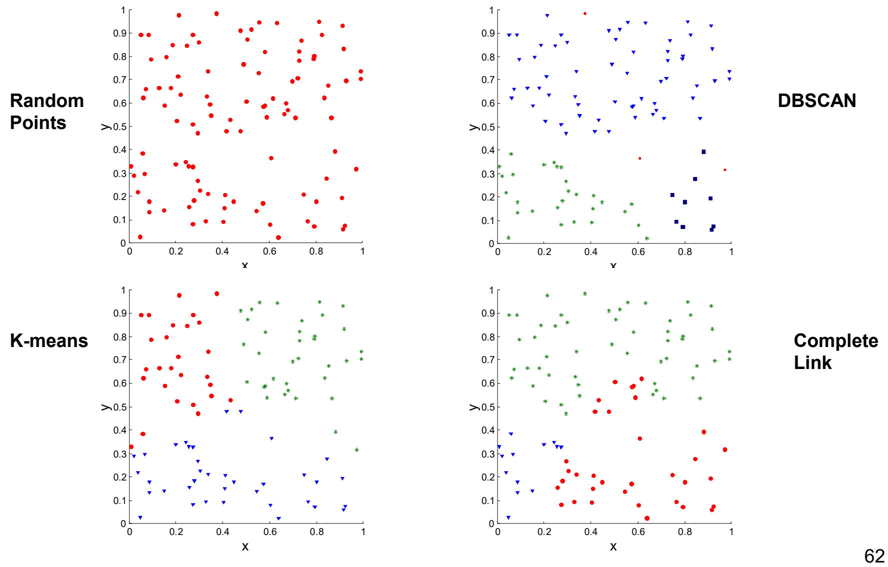
60

Different Aspects of Cluster Validation

- Determining whether an algorithm can identify the clustering tendency of a set of data, i.e., whether it can distinguish whether non-random structure actually exists in the data.
- Determining whether a clustering result is good:
  - Comparing the results of a cluster analysis to externally known results, e.g., to externally given class labels.
  - Evaluating how well the results of a cluster analysis fit the data *without* reference to external information.
    - Use only the data

61

Clusters found in Random Data



62

Measures of Cluster Validity

- Numerical measures that are applied to judge various aspects of cluster validity, are classified into the following two types.
  - External Index: Used to measure the extent to which cluster labels match externally supplied class labels.
    - E.g., HA index, entropy
  - Internal Index: Used to measure the goodness of a clustering structure *without* external information.
    - E.g., Sum of Squared Error (SSE)
- Sometimes these are referred to as criteria instead of indices
  - However, sometimes criterion is the general strategy and index is the numerical measure that implements the criterion.

63

HA Index: an External Index

HA index = $\frac{a+d}{a+b+c+d}$

- U is the true partition in the data set.
- V is the clustering result by some algorithm.
- a is the number of pairs of objects that are placed in the same class in U and in the same cluster in V
- b is the number of pairs of objects in the same class in U but not in the same cluster in V,
- c is the number of pairs of objects in the same cluster in V but not in the same class in U,
- d is the number of pairs of objects in different classes and different clusters in both partitions.

64

SSE: an Internal Index

- Sum of Squared Error (SSE):

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

v_i is the center of the cluster C_i

k is the number of clusters

- Considers only the compactness (i.e., the intra-cluster distances) of the clusters

65

DB: an Internal Index

- Davies-Bouldin index (DB) :

$$DB = \frac{1}{k}\sum_{i} \max_{j=1..k, j \neq i} (d_{ij}) \quad \text{where} \quad d_{ij} = \frac{\sigma_i + \sigma_j}{d(v_i, v_j)}$$

- k is the number of clusters,
- σ_i is the average distance between cluster points and the center in the ith cluster
- $d(v_i, v_j)$ is the distance between the ith and jth cluster centers.
- d_{ij} decreases when the clusters are more compact and when the distance between them is larger.
- For each cluster C_i, the $\max(d_{ij})$ identifies its "worst" neighbour. The DB index is the average of such value for all clusters.
- The DB index varies on the interval $[0,\infty)$ and is small when the clusters are *compact and well separated.*

66

Final Comment on Cluster Validity

"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."

Algorithms for Clustering Data, Jain and Dubes

67

Summary

- Cluster analysis groups objects based on their similarity and has wide applications
- We have looked at three clustering algorithms:
  - K-means
  - K-medoids
  - Hierarchical clustering
  - Density-based clustering
- Cluster Validity

68