Unsupervised Learning

Malaviya National Institute of Technology

October 18, 2015

Supervised Learning vs. Unsupervised Learning

Supervised Learning: discover patterns in the data that relate data attributes with a target (class) attribute. These patterns are then utilized to predict the values of the target attribute in future data instances.

Unsupervised Learning: the data have no target attribute. We want to explore the data to find some intrinsic structures in them.

Clustering

Clustering is a technique for finding similarity groups in data, called clusters. I.e., it groups data instances that are similar to (near) each other in one cluster and data instances that are very different (far away) from each other into different clusters.

Clustering is often called an unsupervised learning task as no class values denoting an a priori grouping of the data instances are given, which is the case in supervised learning.

In fact, association rule mining is also unsupervised.

Clustering Example

Few Applications

Groups people of similar sizes together to make small, medium and large T-Shirts. Tailor-made for each person: too expensive One-size-fits-all: does not fit all.

In marketing, segment customers according to their similarities To do targeted marketing.

Given a collection of text documents, we want to organize them according to their content similarities, To produce a topic hierarchy.

In image processing and retrieval, similar images according to specific criteria may be grouped, image segmentation.

Clustering Aspects

A clustering algorithm can be:

- Partition Based Clustering
- Hierarchical Clustering
- Density Based Clustering

- A distance (similarity, or dissimilarity) function
- Clustering quality
  - Inter-clusters distance is maximized
  - Intra-clusters distance is minimized
- The quality of a clustering result depends on the algorithm, the distance function, and the application.

Distance & Similarity Measures I

- Manhattan Distance
- Euclidean Distance
- Mahalanobis Distance
- Cosine Similarity

Text	laughing	player	dirty	music
Text1	1	1	0	1
Text2	2	0	1	1

$$\frac{1.2+1.0+0.1+1.1}{\sqrt{1^2+1^2+0^2+1^2}\sqrt{2^2+0^2+1^2+1^2}} \cong 0.72$$

Distance & Similarity Measures II

- Jaccard Similarity & Hamming Distance
  Let x = 010101001, y = 010011000
  Hamming distance = 3
  Jaccard coefficient: J= (number of matching presences) / (number of attributes not involved in 00 matches)
  J=(f11)/(f01 + f10 + f11)
  f01 = 1 the number of attributes where x was 0 and y was 1
  f10 = 2 the number of attributes where x was 1 and y was 0
  f00 = 5 the number of attributes where x was 0 and y was 0
  f11 = 2 the number of attributes where x was 1 and y was 1
  J= (2) / (1 + 2 + 2) J = 2/5 = 0.4
- Levenshtein (Edit) Distance
  LD(BIOLOGY, BIOLOGIA)=2
  BIOLOGY $\longrightarrow$ BIOLOGI (substitution)
  BIOLOGI $\longrightarrow$ BIOLOGIA (insertion)

K-means Clustering

- Partition Based Clustering
- The k-means algorithm partitions the given data into k clusters.
  - Each cluster has a cluster center, called centroid
  - k is specified by the user
- The quality of a clustering result depends on the algorithm, the distance function, and the application.

K-means Clustering Algorithm

Algorithm 1 Algorithm k-means(k,D)

1: Randomly choose k data points (seeds) to be the initial centroids, cluster
2: **repeat**
3: **for** Every data point $x \in D$ **do**
4: compute the distance from x to each centroid
5: assign x to the closest centroid
6: **end for**
7: recompute the centroids using current grouping of data points
8: **until** the stopping criterion is met

Stopping or Convergence Conditions

- no (or minimum) re-assignments of data points to different clusters
- no (or minimum) change of centroids
- minimum decrease in the sum of squared error (SSE)

$$SSE = \sum_{k=1}^{N} \sum_{x \in C_j} dist(x, m_j)^2 \quad (1)$$

- C_j is the j^{th} cluster, m_j is the centroid of cluster C_j (the mean vector of all the data points in C_j), and $dist(x, m_j)$ is the distance between data point x and centroid m_j.

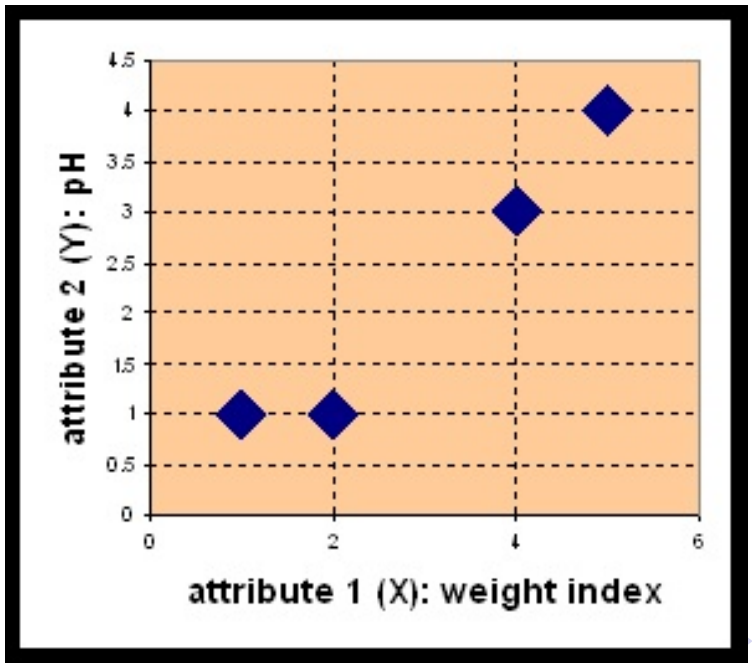
Strengths and Weaknesses

- simple, easy and most popular
- The algorithm is only applicable if the mean is defined.
- For categorical data, k-mode-the centroid is represented by most frequent values.
- The user needs to specify k.
- The algorithm is sensitive to outliers.
  - Outliers are data points that are very far away from other data points.
  - Outliers could be errors in the data recording or some special data points with very different values.

Example: K-means I

- K=2, Euclidean distance.

Object	Attribute 1 (X) Weight Index	Attribute 2 (Y) PH Index
A	1	1
B	2	1
C	4	3
D	5	4

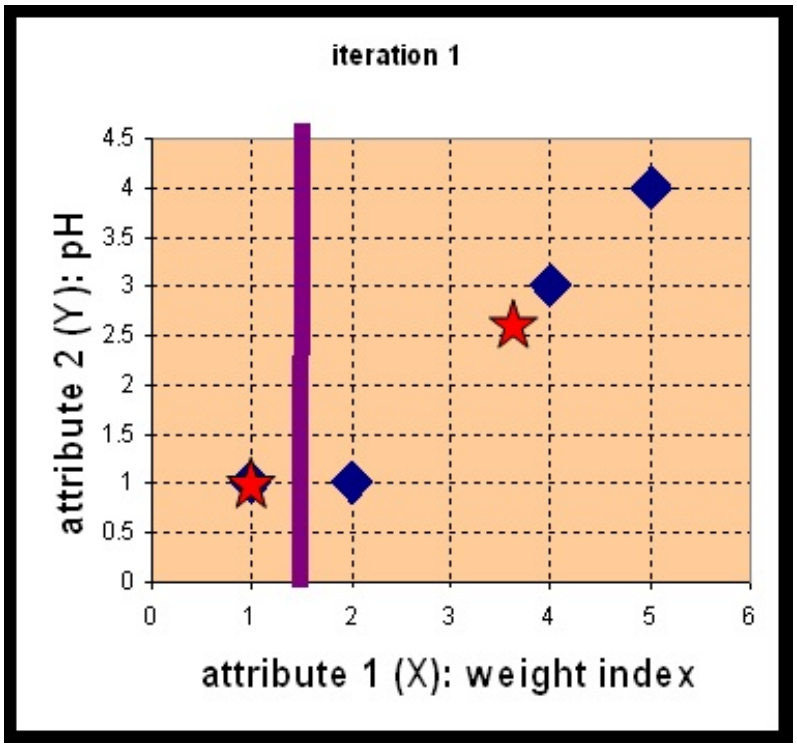


Example: K-means II

- Let C_1=$\{1,1\}$ and C_2=$\{2,1\}$

Distance Table.

Cluster	A	B	C	D
C_1=$\{1,1\}$ Group-1	0	1	3.61	5
C_2=$\{2,1\}$ Group-2	1	0	2.83	4.24

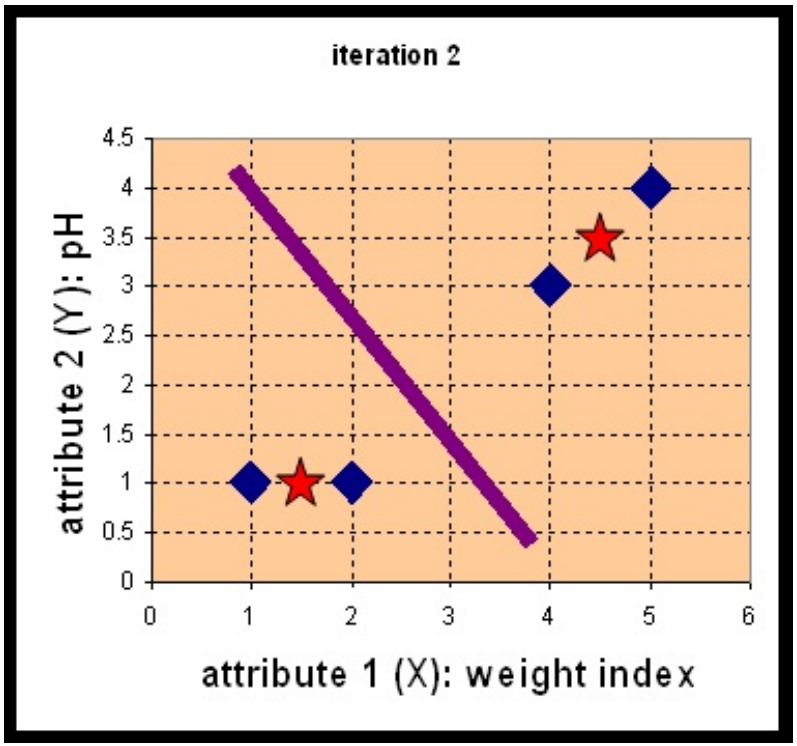


Example: K-means III

- Updated centroids C_1={1,1} and C_2={3.66,2.66}

Distance Table.

Cluster	A	B	C	D
C_1={1,1} Group-1	0	1	3.61	5
C_2={3.66,2.66} Group-2	3.14	2.36	0.47	1.89



Example: K-means IV

- Updated centroids C_1={1.5,1} and C_2={4.5,3.5}

Distance Table.

Cluster	A	B	C	D
C_1={1.5,1} Group-1	0.5	0.5	3.20	4.61
C_2={4.5,3.5} Group-2	4.30	3.54	0.71	0.71

No Change in centroids.

Other Partition Based Clustering Methods

- K-medoid.
- K-nearest neighbour.

Hierarchical Clustering

- Produce a nested sequence of clusters, a tree, also called Dendrogram.
- Agglomerative (bottom up) clustering: It builds the dendrogram (tree) from the bottom level, and
  - merges the most similar (or nearest) pair of clusters
  - stops when all the data points are merged into a single cluster (i.e., the root cluster).
- Divisive (top down) clustering: It starts with all data points in one cluster, the root.
  - splits the root into a set of child clusters. Each child cluster is recursively divided further
  - stops when only singleton clusters of individual data points remain, i.e., each cluster with only a single point

Agglomerative Clustering

- It is more popular then divisive methods.
- At the beginning, each data point forms a cluster (also called a node).
- Merge nodes/clusters that have the least distance.
- Go on merging
- Eventually all nodes belong to one cluster

Agglomerative Clustering Algorithm

Algorithm 2 Agglomerative(D)

1: Make each data point in the dataset D as cluster
2: Compute all pair distances of each data point
3: **repeat**
4: Find two clusters that are nearest to each other
5: Merge the two clusters to form a new cluster C
6: Compute the distance from C to all other clusters
7: **until** their is only one cluster left

Example: Agglomerative Clustering I

	A	B	C	D	E	F
A	0.00	0.71	5.66	3.61	4.24	3.20
B	0.71	0.00	4.95	2.92	3.54	2.50
C	5.66	4.95	0.00	2.24	1.41	2.50
D	3.61	2.92	2.24	0.00	1.00	0.50
E	4.24	3.54	1.41	1.00	0.00	1.12
F	3.20	2.50	2.50	0.50	1.12	0.00

Example: Agglomerative Clustering II

	A	B	C	D,F	E
A	0.00	0.71	5.66	3.20	4.24
B	0.71	0.00	4.95	2.50	3.54
C	5.66	4.95	0.00	2.24	1.41
D,F	3.20	2.50	2.24	0.00	1.00
E	4.24	3.54	1.41	.00	0.00

Example: Agglomerative Clustering III

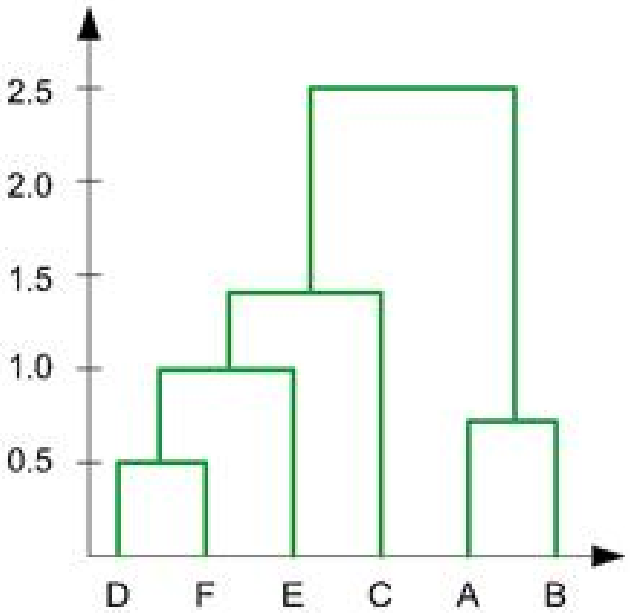
	A,B	C	D,F	E
A,B	0.00	4.95	2.50	3.54
C	4.95	0.00	2.24	1.41
D,F	2.50	2.24	0.00	1.00
E	3.54	1.41	1.00	0.00

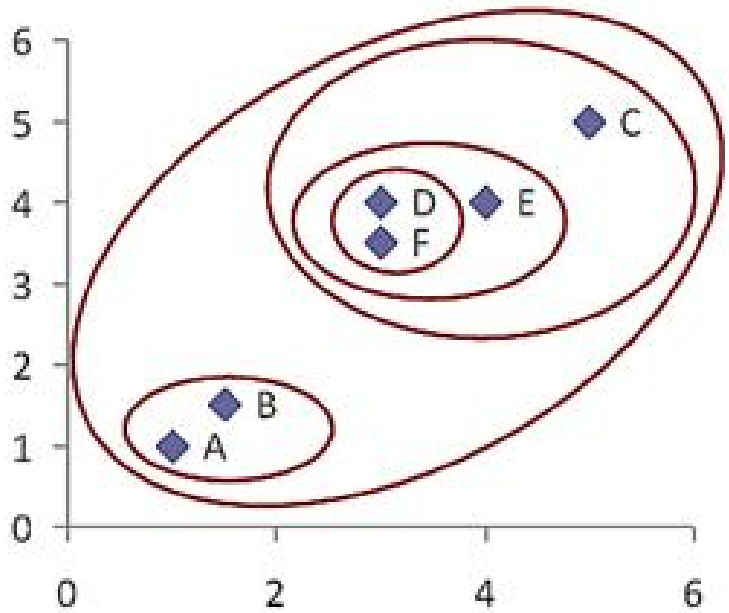
Example: Agglomerative Clustering IV

	A,B	C	(D,F),E
A,B	0.00	4.95	2.50
C	4.95	0.00	1.41
(D,F),E	2.50	1.41	0.00

	A,B	((D,F),E),C
A,B	0.00	2.50
((D,F),E),C	2.50	0.00

Example: Agglomerative Clustering V



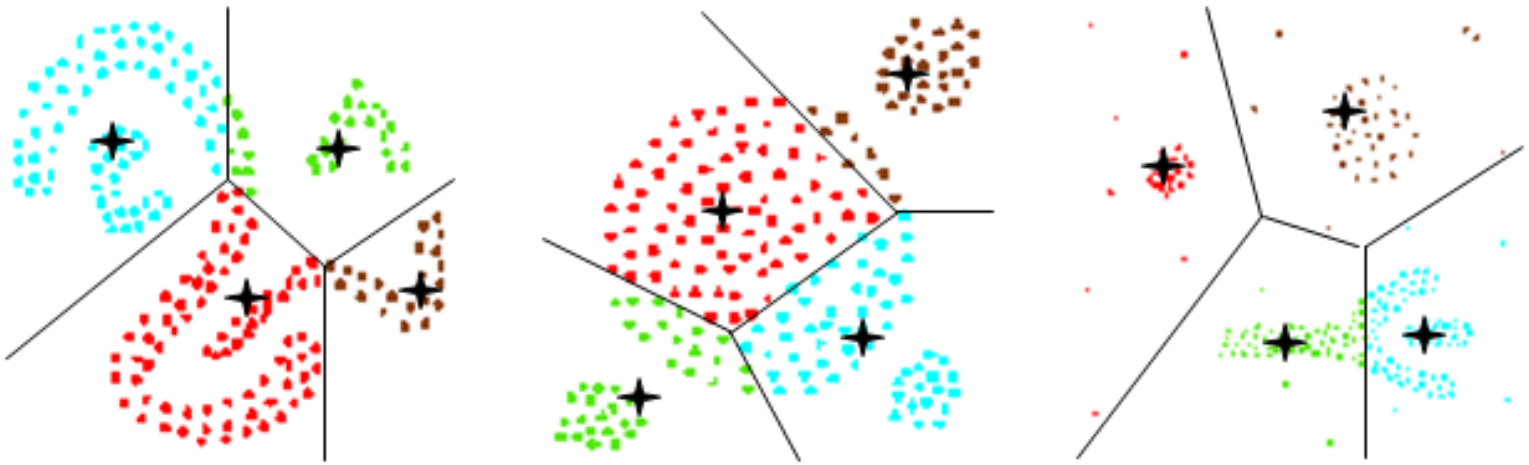


Measuring the Distance of two Clusters

- Single link: The distance between two clusters is the distance between two closest data points in the two clusters, one data point from each cluster. It can find arbitrarily shaped clusters, but It may cause the undesirable "chain effect" by noisy points.
- Complete link: The distance between two clusters is the distance of two furthest data points in the two clusters. It is sensitive to outliers because they are far away.
- Average link: In this method, the distance between two clusters is the average distance of all pair-wise distances between the data points in two clusters.
- Centroids: In this method, the distance between two clusters is the distance between their centroids.

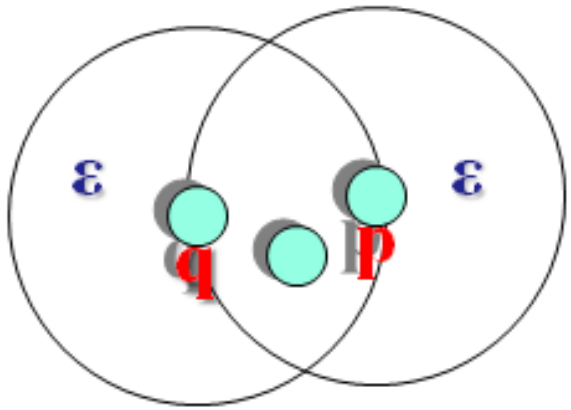
Density Based Clustering, why?

Density-based spatial clustering of applications with noise (DBSCAN)
K-medoid result for k=4



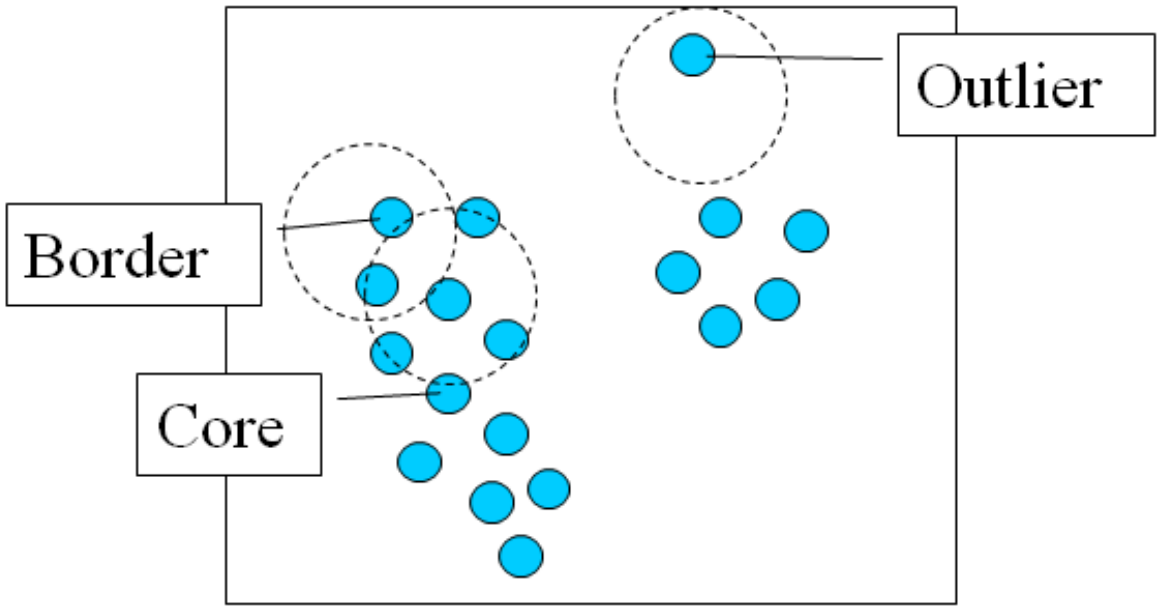
Density Based Clustering: Basics

- For any point in a cluster, the local point density around that point has to exceed some threshold
- The set of points from one cluster is spatially connected
- Local point density at a point p defined by two parameters
  - ϵ - radius for the neighborhood of point p:
    N_ϵ(p) := {q in data set D | $dist(p, q) \leq \epsilon$}
  - MinPts - minimum number of points in the given neighborhood N(p)



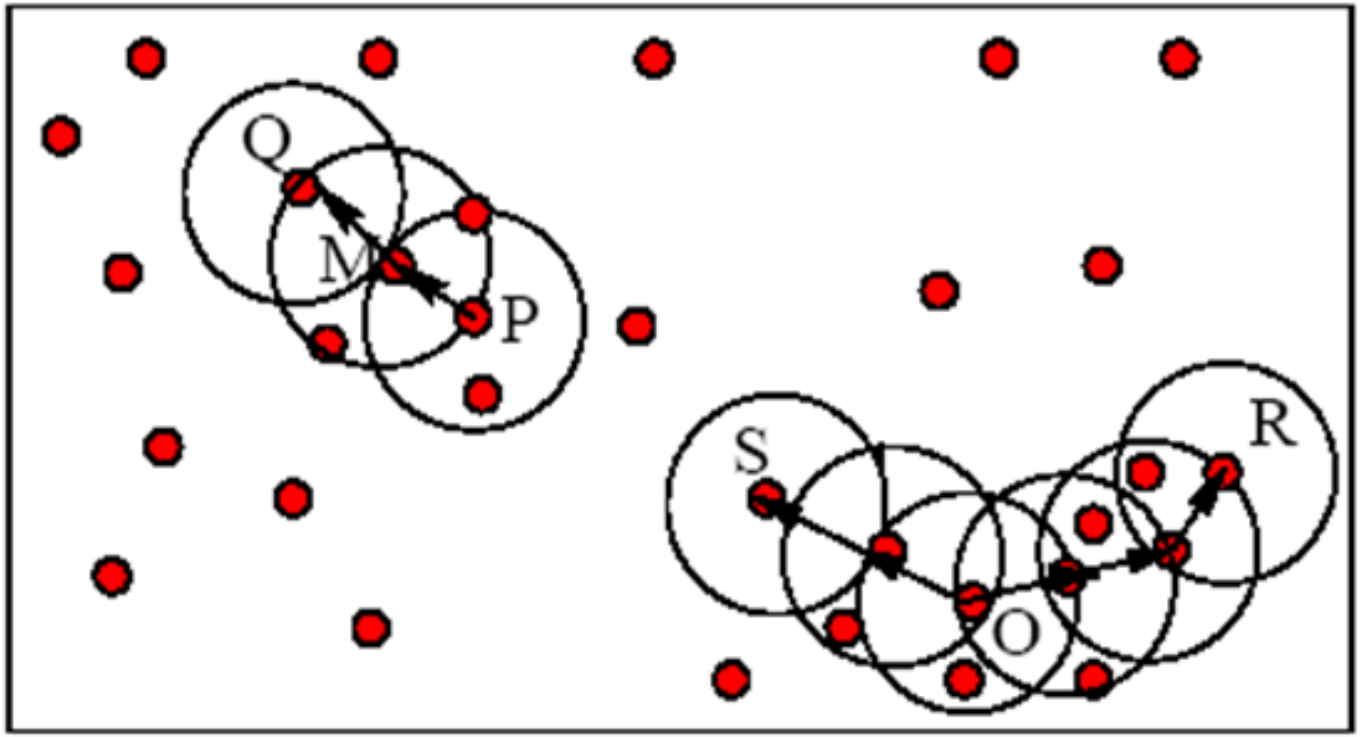
Density of p is high (MinPts = 4) Density of q is low (MinPts = 4)

Core, Border & Outlier



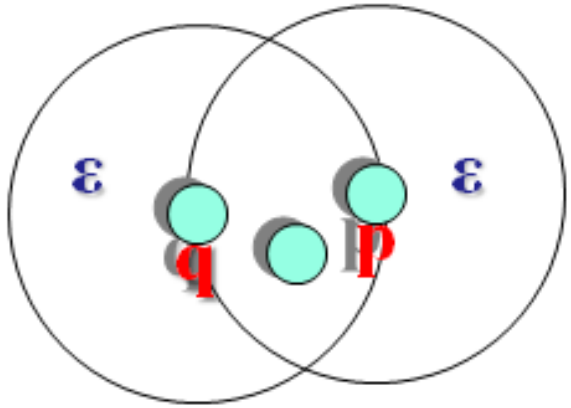
- A point is a core point if it has more than a specified number of points (MinPts) within Eps These are points that are at the interior of a cluster.
- A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point.
- A noise point is any point that is not a core point nor a border point.

Core, Border & Outlier: Example



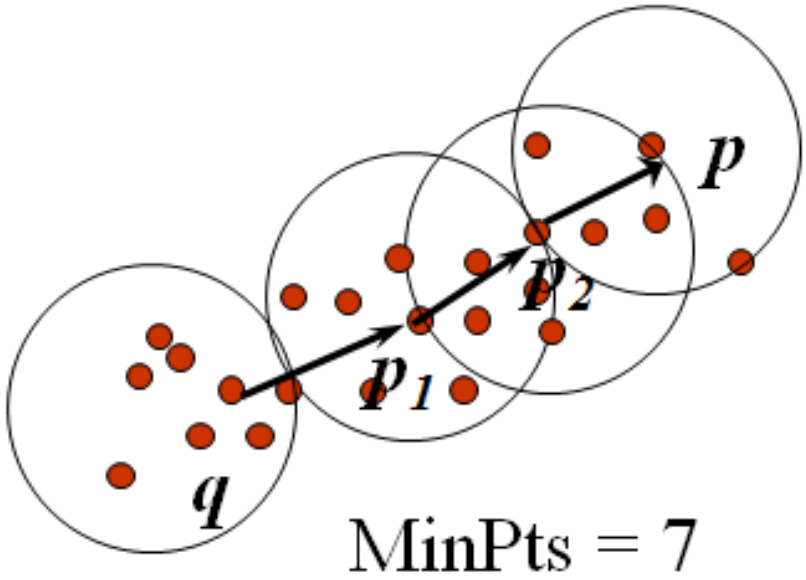
- M, P, O, and R are core objects since each is in an Eps neighborhood containing at least 3 points.

Density-Reachability



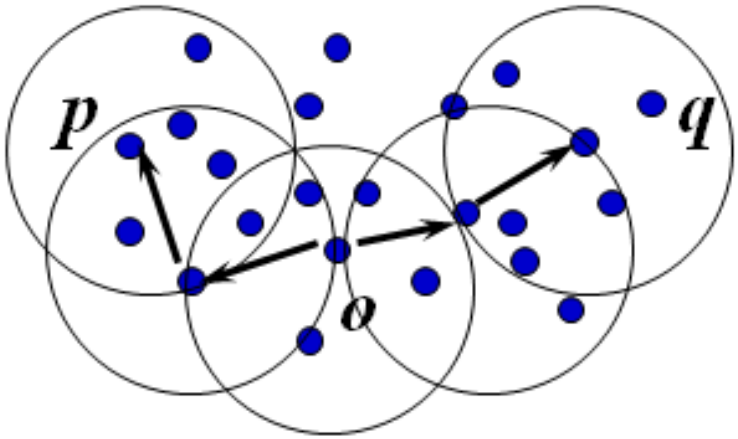
- Directly density-reachable: An object q is directly density-reachable from object p if p is a core object and q is in p's ϵ-neighborhood.
- q is directly density-reachable from p.
- p is not directly density- reachable from q?
- Density-reachability is asymmetric.

Density-Reachability



- A point p is directly density-reachable from p2.
- p2 is directly density-reachable from p1.
- p1 is directly density-reachable from q.
- p $\rightarrow$ p2$\rightarrow$ p1$\rightarrow$ q form a chain.
- p is (indirectly) density-reachable from q.
- q is not density- reachable from p?

Density-Connected



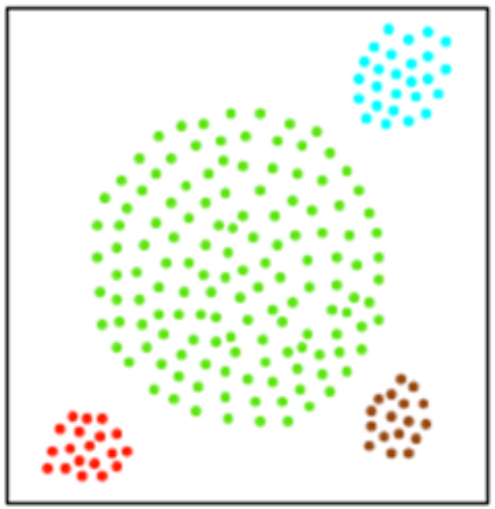
- Density-reachable is not symmetric
- not good enough to describe clusters.
- Density-Connected: A pair of points p and q are density-connected if they are commonly density-reachable from a point o.
- Density-connectivity is symmetric.

DBSCAN Algorithm

Algorithm 3 DBSCAN(D)

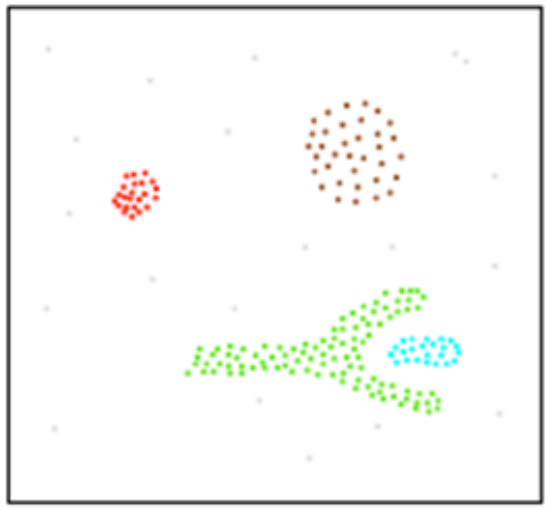
1: select a point p.
2: Retrieve all points density-reachable from p wrt ϵ and MinPts.
3: If p is a core point, a cluster is formed.
4: If p is a border point, no points are density-reachable from p and DBSCAN visits the next point of the database.
5: Continue the process until all of the points have been processed.

Example: DBSCAN



database 1

database 2



database 3

How to choose a Clustering Algorithm

- Every algorithm has limitations and works well with certain data distributions.
- It is very hard, if not impossible, to know what distribution the application data follow. The data may not fully follow any "ideal" structure or distribution required by the algorithms.
- One also needs to decide how to standardize the data, to choose a suitable distance function and to select other parameter values.
- Run several algorithms using different distance functions and parameter settings, and then carefully analyze and compare the results.
- The interpretation of the results must be based on insight into the meaning of the original data together with knowledge of the algorithms used. Clustering is highly application dependent and to certain extent subjective (personal preferences).

The End