DBSCAN

**DBSCAN** (for Density-Based Spatial Clustering of Applications with Noise) is a data clustering algorithm proposed by [Martin Ester](http://en.wikipedia.org/w/index.php?title=Martin_Ester&action=edit&redlink=1), [Hans-Peter Kriegel](http://en.wikipedia.org/wiki/Hans-Peter_Kriegel), [Jörg Sander](http://en.wikipedia.org/w/index.php?title=J%C3%B6rg_Sander&action=edit&redlink=1)and [Xiaowei Xu](http://en.wikipedia.org/w/index.php?title=Xiaowei_Xu&action=edit&redlink=1) in 1996.[[1]](http://en.wikipedia.org/wiki/DBSCAN#cite_note-1) It is a density-based clustering algorithm because it finds a number of clusters starting from the estimated density distribution of corresponding nodes.

**Basic Idea**

DBSCAN's definition of a cluster is based on the notion of density reachability. Basically, a point q is directly density-reachable from a point p if it is not farther away than a given distance \varepsilon (i.e., is part of its \varepsilon-neighborhood) and if p is surrounded by sufficiently many points such that one may consider p and q to be part of a cluster. *q* is called density-reachable (note the distinction from "directly density-reachable") from p if there is a sequence p_1,\ldots,p_n of points with p_1 = p and p_n = q where each p_{i+1} is directly density-reachable fromp_i.

The relation of density-reachable is not symmetric. q might lie on the edge of a cluster, having insufficiently many neighbors to count as dense itself. This would halt the process of finding a path that stops with the first non-dense point. By contrast, starting the process with q would lead to p (though the process would halt there, pbeing the first non-dense point). Due to this asymmetry, the notion of density-connected is introduced: two points p and q are density-connected if there is a point o such that both p and q are density-reachable fromo. Density-connectedness is symmetric.

**Density connectedness**

A cluster, which is a subset of the points of the database, satisfies two properties:

1. All points within the cluster are mutually density-connected.
2. If a point is density-connected to any point of the cluster, it is part of the cluster as well.

**Complexities**

DBSCAN visits each point of the database, possibly multiple times (e.g., as candidates to different clusters). For practical considerations, however, the time complexity is mostly governed by the number of region Query invocations. DBSCAN executes exactly one such query for each point, and if an indexing structure is used that executes such a neighborhood query in O(log n), an overall runtime complexity of O(n \cdot \log n) is obtained. Without the use of an accelerating index structure, the run time complexity is O(n^2). Often the distance matrix of size (n^2-n)/2 is materialized to avoid distance recomputations. This however also needs O(n^2) memory.

**Advantages**

1. DBSCAN does not require one to specify the number of clusters in the data a priori, as opposed to k-means.
2. DBSCAN can find arbitrarily shaped clusters. It can even find a cluster completely surrounded by (but not connected to) a different cluster. Due to the MinPts parameter, the so-called single-link effect (different clusters being connected by a thin line of points) is reduced.
3. DBSCAN has a notion of noise.
4. DBSCAN requires just two parameters and is mostly insensitive to the ordering of the points in the database. (However, points sitting on the edge of two different clusters might swap cluster membership if the ordering of the points is changed, and the cluster assignment is unique only up to isomorphism.)

**Disadvantages**

1. The quality of DBSCAN depends on the distance measure used in the function regionQuery (P,\varepsilon). The most common distance metric used is Euclidean distance. Especially for high-dimensional data, this metric can be rendered almost useless due to the so-called "Curse of dimensionality", making it difficult to find an appropriate value for\varepsilon. This effect, however, is also present in any other algorithm based on Euclidean distance.
2. DBSCAN cannot cluster data sets well with large differences in densities, since the minPts-\varepsilon combination cannot then be chosen appropriately for all clusters.