Ann. Data. Sci. (2015) 2(2):165–193
DOI 10.1007/s40745-015-0040-1

A Comprehensive Survey of Clustering Algorithms

Dongkuan Xu[1,2] · Yingjie Tian[2,3]

Received: 25 May 2015 / Revised: 18 July 2015 / Accepted: 31 July 2015 /
Published online: 12 August 2015

Abstract Data analysis is used as a common method in modern science research, which is across communication science, computer science and biology science. Clustering, as the basic composition of data analysis, plays a significant role. On one hand, many tools for cluster analysis have been created, along with the information increase and subject intersection. On the other hand, each clustering algorithm has its own strengths and weaknesses, due to the complexity of information. In this review paper, we begin at the definition of clustering, take the basic elements involved in the clustering process, such as the distance or similarity measurement and evaluation indicators, into consideration, and analyze the clustering algorithms from two perspectives, the traditional ones and the modern ones. All the discussed clustering algorithms will be compared in detail and comprehensively shown in Appendix Table 22.

Keywords Clustering · Clustering algorithm · Clustering analysis · Survey · Unsupervised learning

✉ Yingjie Tian
tyj@ucas.ac.cn

Dongkuan Xu
xudongkuan14@mails.ucas.ac.cn

[1] School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing 100049, China

[2] Research Center on Fictitious Economy & Data Science, Chinese Academy of Sciences, Beijing 100190, China

[3] Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing 100190, China

1 Introduction

Clustering, considered as the most important question of unsupervised learning, deals with the data structure partition in unknown area and is the basis for further learning. The complete definition for clustering, however, isn't come to an agreement, and a classic one is described as follows [1]:

(1) Instances, in the same cluster, must be similar as much as possible;
(2) Instances, in the different clusters, must be different as much as possible;
(3) Measurement for similarity and dissimilarity must be clear and have the practical meaning;

The standard process of clustering can be divided into the following several steps [2]:

(1) Feature extraction and selection: extract and select the most representative features from the original data set;
(2) Clustering algorithm design: design the clustering algorithm according to the characteristics of the problem;
(3) Result evaluation: evaluate the clustering result and judge the validity of algorithm;
(4) Result explanation: give a practical explanation for the clustering result;

In the rest of this paper, the common similarity and distance measurements will be introduced in Sect. 2, the evaluation indicators for the clustering result will be listed in section 3, the traditional clustering algorithms and the modern ones will be analyzed systematically respectively in Sects. 4 and 5, and the final conclusion will be drawn in Sect. 6.

2 Distance and Similarity

Distance (dissimilarity) and similarity are the basis for constructing clustering algorithms. As for quantitative data features, distance is preferred to recognize the relationship among data. And similarity is preferred when dealing with qualitative data features [2].

The common used distance functions for quantitative data feature are summarized in Table 1.

The common used similarity functions for qualitative data feature are summarized in Table 2.

3 Evaluation Indicator

The main purpose of evaluation indicator is to test the validity of algorithm. Evaluation indicators can be divided into two categories, the internal evaluation indicators and the external evaluation indicators, in terms of the test data whether in the process of constructing the clustering algorithm.

The internal evaluation takes the internal data to test the validity of algorithm. It, however, can't absolutely judge which algorithm is better when the scores of two

Table 1 Distance functions

Name	Formula	Explanation
Minkowski distance	$\left(\sum_{l=1}^{d} \left\|x_{il} - x_{jl}\right\|^{n}\right)^{1/n}$	A set of definitions for distance: 1. City-block distance when n = 1 2. Euclidean distance when n = 2 3. Chebyshev distance when n → ∞
Standardized Euclidean distance	$\left(\sum_{l=1}^{d} \left\|\frac{x_{il}-x_{jl}}{s_l}\right\|^{2}\right)^{1/2}$	1. S stands for the standard deviation 2. A weighted Euclidean distance based on the deviation
Cosine distance	$1 - \cos\alpha = \frac{x_i^T x_j}{\\|x_i\\| \\|x_j\\|}$	1. Stay the same in face of the rotation change of data 2. The most commonly used distance in document area
Pearson correlation distance	$1 - \frac{Cov(x_i, x_j)}{\sqrt{D(x_i)}\sqrt{D(x_j)}}$	1. Cov stands for the covariance for and D stands for the variance 2. Measure the distance based on linear correlation
Mahalanobis distance	$\sqrt{(x_i - x_j)^T S^{-1} (x_i - x_j)}$	1. S is the covariance matrix inside the cluster 2. With high computation complexity

algorithms are not equal based on the internal evaluation indicators [5]. There are three commonly used internal indicators, summarized in Table 3.

The external evaluation, which is called as the gold standard for testing method, takes the external data to test the validity of algorithm. However, it turns out that the external evaluation is not completely correct recently [6]. There are six common used external evaluation indicators, summarized in Table 4.

In the following sections, especially in the analysis of time complexity, n stands for the number of total objects/data points, k stands for the number of clusters, s stands for the number of sample objects/data points, and t stands for the number of iterations.

4 Traditional Clustering Algorithms

The traditional clustering algorithms can be divided into 9 categories which mainly contain 26 commonly used ones, summarized in Table 5.

Table 2 Similarity functions

Name	Function formula or measure method	Explanation
Jaccard similarity	$J(A, B) = \frac{\|A \cap B\|}{\|A \cup B\|}$	1. Measure the similarity of two sets 2. |X| Stands for the number of elements of set X 3. Jaccard distance = 1 − Jaccard similarity
Hamming similarity	The minimum number of substitutions needed to change one data point into the other	The number is smaller, the similarity is more Hamming distance is the opposite of Hamming similarity Especially for the data of string
For data of mixed type	Map the feature into (0, 1) Transform the feature into dichotomous one $S_{ij} = \frac{1}{d}\sum_{l=1}^{d} s_{ijl}$ $S_{ij} = \left(\sum_{l=1}^{d} \eta_{ijl} s_{ijl}\right) / \left(\sum_{l=1}^{d} \eta_{ijl}\right)$	[3,4]

4.1 Clustering Algorithm Based on Partition

The basic idea of this kind of clustering algorithms is to regard the center of data points as the center of the corresponding cluster. K-means [7] and K-medoids [8] are the two most famous ones of this kind of clustering algorithms. The core idea of K-means is to update the center of cluster which is represented by the center of data points, by iterative computation and the iterative process will be continued until some criteria for convergence is met. K-mediods is an improvement of K-means to deal with discrete data, which takes the data point, most near the center of data points, as the representative of the corresponding cluster. The typical clustering algorithms based on partition also include PAM [9], CLARA [10], CLARANS [11].

For more information about this kind of clustering algorithms, you can refer to [12–14].

Analysis:

(1) Time complexity (Table 6):

(2) Advantages: relatively low time complexity and high computing efficiency in general;

Table 3 Evaluation indicators

Name	Formula or measure method	Explanation
Davies–Bouldin indicator	$DB = \frac{1}{k}\sum_{i=1}^{k}\max_{i\neq j}\left(\frac{\sigma_i+\sigma_j}{d(c_i,c_j)}\right)$	K stands for the number of clusters, C_x is the center of cluster x, σ_x is the average distance between any data in cluster x and C_x, $d\left(c_i, c_j\right)$is the distance between c_i and c_j
Dunn indicator	$D = \min_{1\leq i\leq n}\left\{\min_{1\leq j\leq n, i\neq j}\left\{\frac{d(i,j)}{\max_{1\leq k\leq n} d'(k)}\right\}\right\}$	1. Mainly for the data that has even density and distribution 2. $d\left(c_i, c_j\right)$is the distance between c_i and c_j, $d'\left(k\right)$ stands for the distance in cluster k
Silhouette coefficient	Evaluate the clustering result based on the average distance between a data point and other data points in the same cluster and average distance among different clusters	

(3) Disadvantages: not suitable for non-convex data, relatively sensitive to the outliers, easily drawn into local optimal, the number of clusters needed to be preset, and the clustering result sensitive to the number of clusters.;

(4) AP algorithm [15], which will be discussed in the section Clustering algorithm based on affinity propagation, can also be considered as one of this kind of clustering algorithm.

4.2 Clustering Algorithm Based on Hierarchy

The basic idea of this kind of clustering algorithms is to construct the hierarchical relationship among data in order to cluster [16]. Suppose that each data point stands for an individual cluster in the beginning, and then, the most neighboring two clusters are merged into a new cluster until there is only one cluster left. Or, a reverse process. Typical algorithms of this kind of clustering include BIRCH [17], CURE [18], ROCK [19], Chameleon [20]. BIRCH realizes the clustering result by constructing the feature tree of clustering, CF tree, of which one node stands for a subcluster. CF tree will dynamically grow when a new data point comes. CURE, suitable for large-scale clustering, takes random sampling technique to cluster sample separately and integrates the results finally. ROCK is an improvement of CURE for dealing with data of enumeration type, which takes the effect on the similarity from the data around the cluster into consideration. Chameleon, at first, divides the original data into clusters with smaller size based on the nearest neighbor graph, and then the clusters with small

Table 4 Evaluation indicators

Name	Formula or measure method	Explanation
Rand indicator	$RI = \frac{TP+TN}{TP+FP+FN+TN}$	1. TP is the number of true positives 2. TN is the number of true negatives 3. FP is the number of false positives 4. FN is the number of false negatives
F indicator	$F_\beta = \frac{\left(\beta^2+1\right)\cdot P\cdot R}{\beta^2\cdot P+R}$	1. $P = \frac{TP}{TP+FP}$ stands for the accuracy, $R = \frac{TP}{TP+FN}$ stands for the recall rate 2. TP, TN, FP, and FN are defined as before
Jaccard indicator	$J(A, B) = \frac{\|A\cap B\|}{\|A\cup B\|} = \frac{TP}{TP+FP+FN}$	1. Measure the similarity of two sets 2. |X| Stands for the number of elements of set X 3. TP, TN, FP, and FN are defined as before
Fowlkes–Mallows indicator	$FM = \sqrt{\frac{TP}{TP+FP}\cdot\frac{TP}{TP+FN}}$	TP, TN, FP, and FN are defined as before
Mutual information	To measure, based on information theory, how much information is shared by two clusters, between which the nonlinear correlation can be detected	
Confusion matrix	To figure out the difference between a cluster and a gold-standard cluster	

size are merged into a cluster with bigger size, based on agglomerative algorithm, until satisfied.

For more information about this kind of clustering algorithms, you can refer to [21,22].

Analysis:

(1) Time complexity (Table 7):

(2) Advantages: suitable for the data set with arbitrary shape and attribute of arbitrary type, the hierarchical relationship among clusters easily detected, and relatively high scalability in general;

(3) Disadvantages: relatively high in time complexity in general, the number of clusters needed to be preset.

Table 5 Traditional algorithms

Category	Typical algorithm
Clustering algorithm based on partition	K-means, K-medoids, PAM, CLARA, CLARANS
Clustering algorithm based on hierarchy	BIRCH, CURE, ROCK, Chameleon
Clustering algorithm based on fuzzy theory	FCM, FCS, MM
Clustering algorithm based on distribution	DBCLASD, GMM
Clustering algorithm based on density	DBSCAN, OPTICS, Mean-shift
Clustering algorithm based on graph theory	CLICK, MST
Clustering algorithm based on grid	STING, CLIQUE
Clustering algorithm based on fractal theory	FC
Clustering algorithm based on model	COBWEB, GMM, SOM, ART

Table 6 Time complexity

K-means	K-medoids	PAM	CLARA	CLARANS
O(knt)	O(k(n-k)^2)	O(k^3*n^2)	O(ks^2+k(n-k))	O(n^2)
Low	High	High	Middle	High

Table 7 Time complexity

BIRCH	CURE	ROCK	Chameleon
O(n)	O(s^2*s),	O(n^2*logn)	O(n^2)
Low	Low	High	High

4.3 Clustering Algorithm Based on Fuzzy Theory

The basic idea of this kind of clustering algorithms is that the discrete value of belonging label, {0, 1}, is changed into the continuous interval [0, 1], in order to describe the belonging relationship among objects more reasonably. Typical algorithms of this kind of clustering include FCM [23–25], FCS [26] and MM [27]. The core idea of FCM is to get membership of each data point to every cluster by optimizing the object function. FCS, different from the traditional fuzzy clustering algorithms, takes the multidimensional hypersphere as the prototype of each cluster, so as to cluster with the distance function based on the hypersphere. MM, based on the Mountain Function, is used to find the center of cluster.

For more information about this kind of clustering algorithms, you can refer to [28–30].

Table 8 Time complexity

FCM	FCS	MM
O(n)	(kernel)	O(vˆ2*n)
Low	High	Middle

Table 9 Time complexity

DBCLASD	GMM
O(n*logn)	O(nˆ2*kt)
Middle	High

Analysis:

1) Time complexity (Table 8):
2) The time complexity of FCS is high for the kernel involved in the algorithm;
3) Advantages: more realistic to give the probability of belonging, relatively high accuracy of clustering;
4) Disadvantages: relatively low scalability in general, easily drawn into local optimal, the clustering result sensitive to the initial parameter values, and the number of clusters needed to be preset.

4.4 Clustering Algorithm Based on Distribution

The basic idea is that the data, generated from the same distribution, belongs to the same cluster if there exists several distributions in the original data. The typical algorithms are DBCLASD [31] and GMM [32]. The core idea of DBCLASD, a dynamic incremental algorithm, is that if the distance between a cluster and its nearest data point satisfies the distribution of expected distance which is generated from the existing data points of that cluster, the nearest data point should belong to this cluster. The core idea of GMM is that GMM consists of several Gaussian distributions from which the original data is generated and the data, obeying the same independent Gaussian distribution, is considered to belong to the same cluster.

For more information about this kind of clustering algorithms, you can refer to [33,34].

Analysis:

(1) Time complexity (Table 9):
(2) Advantages: more realistic to give the probability of belonging, relatively high scalability by changing the distribution, number of clusters and so on, and supported by the well developed statistical science;
(3) Disadvantages: the premise not completely correct, involved in many parameters which have a strong influence on the clustering result and relatively high time complexity.

Table 10 Time complexity

DBSCAN	OPTICS	Mean-shift
O(n*logn)	O(n*logn)	(kernel)
Middle	Middle	High

4.5 Clustering Algorithm Based on Density

The basic idea of this kind of clustering algorithms is that the data which is in the region with high density of the data space is considered to belong to the same cluster [35]. The typical ones include DBSCAN [36], OPTICS [37] and Mean-shift [38]. DBSCAN is the most well known density-based clustering algorithm, which is generated from the basic idea of this kind of clustering algorithms directly. OPTICS is an improvement of DBSCAN and it overcomes the shortcoming of DBSCAN that being sensitive to two parameters, the radius of the neighborhood and the minimum number of points in a neighborhood. In the process of Mean-shift, the mean of offset of current data point is calculated at first, the next data point is figured out based on the current data point and the offset then, and last, the iteration will be continued until some criteria are met.

For more information about this kind of clustering algorithms, you can refer to [39–42].

Analysis:

(1) Time complexity (Table 10):
(2) The time complexity of Mean-shift is high for the kernel involved in the algorithm;
(3) Advantages: clustering in high efficiency and suitable for data with arbitrary shape;
(4) Disadvantages: resulting in a clustering result with low quality when the density of data space isn't even, a memory with big size needed when the data volume is big, and the clustering result highly sensitive to the parameters;
(5) DENCLUE algorithm [43], which will be discussed in the section Clustering algorithm for large-scale data, can also be considered as one of this kind of clustering algorithms.

4.6 Clustering Algorithm Based on Graph Theory

According to this kind of clustering algorithms, clustering is realized on the graph where the node is regarded as the data point and the edge is regarded as the relationship among data points. Typical algorithms of this kind of clustering are CLICK [44] and MST-based clustering [45]. The core idea of CLICK is to carry out the minimum weight division of the graph with iteration in order to generate the clusters. Generating the minimum spanning tree from the data graph is the key step to do the cluster analysis for the MST-based clustering algorithm.

For more detailed information about this kind of clustering algorithms, you can refer to [1,20,46–49].

Table 11 Time complexity

CLICK	MST
O(k*f(v, e))	O(e*logv)
Low	Middle

Table 12 Time complexity

STING	CLIQUE
O(n)	O(n+kˆ2)
Low	Low

Analysis:

(1) Time complexity (Table 11):
where v stands for the number of vertices, e stands for the number of edges, and f(v, e) stands for the time complexity of computing a minimum cut;
(2) Advantages: clustering in high efficiency, the clustering result with high accuracy;
(3) Disadvantages: the time complexity increasing dramatically with the increasing of graph complexity;
(4) SM algorithm [50] and NJW algorithm [51], which will be discussed in the section Clustering algorithm based on spectral graph theory, can also be considered as ones of this kind of clustering algorithms.

4.7 Clustering Algorithm Based on Grid

The basic idea of this kind of clustering algorithms is that the original data space is changed into a grid structure with definite size for clustering. The typical algorithms of this kind of clustering are STING [52] and CLIQUE [53]. The core idea of STING which can be used for parallel processing is that the data space is divided into many rectangular units by constructing the hierarchical structure and the data within different structure levels is clustered respectively. CLIQUE takes advantage of the grid-based clustering algorithms and the density-based clustering algorithms.

For more detailed information about this kind of clustering algorithms, you can refer to [41,54–57].

Analysis:

(1) Time complexity (Table 12):
(2) Advantages: low time complexity, high scalability and suitable for parallel processing and increment updating;
(3) Disadvantages: the clustering result sensitive to the granularity (the mesh size), the high calculation efficiency at the cost of reducing the quality of clusters and reducing the clustering accuracy;
4) Wavecluster algorithm [54], which will be discussed in the section Clustering algorithm for spatial data, can also be considered as ones of this kind of clustering algorithms.

4.8 Clustering Algorithm Based on Fractal Theory

Fractal stands for the geometry that can be divided into several parts which share some common characters with the whole [58]. The typical algorithm of this kind of clustering is FC [59] of which the core idea is that the change of any inner data of a cluster does not have any influence on the intrinsic quality of the fractal dimension.

For more detailed information about this kind of clustering algorithms, you can refer to [60–63].

Analysis:

(1) The time complexity of FC is O(n);
(2) Advantages: clustering in high efficiency, high scalability, dealing with outliers effectively and suitable for data with arbitrary shape and high dimension;
(3) Disadvantages: the premise not completely correct, the clustering result sensitive to the parameters.

4.9 Clustering Algorithm Based on Model

The basic idea is to select a particular model for each cluster and find the best fitting for that model. There are mainly two kinds of model-based clustering algorithms, one based on statistical learning method and the other based on neural network learning method.

The typical algorithms, based on statistical learning method, are COBWEB [64] and GMM [32]. The core idea of COBWEB is to build a classification tree, based on some heuristic criteria, in order to realize hierarchical clustering on the assumption that the probability distribution of each attribute is independent. The typical algorithms, based on neural network learning method, are SOM [65] and ART [66–69]. The core idea of SOM is to build a mapping of dimension reduction from the input space of high dimension to output space of low dimension on the assumption that there exists topology in the input data. The core idea of ART, an incremental algorithm, is to generate a new neuron dynamically to match a new pattern to create a new cluster when the current neurons are not enough. GMM has been discussed in the section Clustering algorithm based on distribution.

For more detailed information about this kind of clustering algorithms, you can refer to [70–75].

Analysis:

(1) Time complexity (Table 13):
(2) The time complexity of COBWEB is generally low, which depends on the distribution involved in the algorithm;

Table 13 Time complexity

COBWEB	GMM	SOM	ART
(distribution)	O(n^2*kt)	(layer)	(type+layer)
Low	High	High	Middle

(3) The time complexity of SOM is generally high, which depends on the layer construction involved in the algorithm;
(4) The time complexity of ART is generally middle, which depends on the type of ART and the layer construction involved in the algorithm;
(5) Advantages: diverse and well developed models providing means to describe data adequately and each model having its own special characters that may bring about some significant advantages in some specific areas;
(6) Disadvantages: relatively high time complexity in general, the premise not completely correct, and the clustering result sensitive to the parameters of selected models.

5 Modern Clustering Algorithms

The modern clustering algorithms can be divided into 10 categories which mainly contain 45 commonly used ones, summarized in Table 14.

Table 14 Modern algorithms

Category	Typical algorithm
Clustering algorithm based on kernel	kernel K-means, kernel SOM, kernel FCM, SVC, MMC, MKC
Clustering algorithm based on ensemble	Methods for generating the set of clusters: 4 types Consensus function: CSPA, HGPA, MCLA, VM, HCE, LAC, WPCK, sCSPA, sMCLA, sHBGPA
Clustering algorithm based on swarm intelligence	ACO_based(LF), PSO_based, SFLA_based, ABC_based
Clustering algorithm based on quantum theory	QC, DQC
Clustering algorithm based on spectral graph theory	SM, NJW
Clustering algorithm based on affinity propagation	AP
Clustering algorithm based on density and distance	DD
Clustering algorithm for spatial data	DBSCAN, STING, Wavecluster, CLARANS
Clustering algorithm for data stream	STREAM, CluStream, HPStream, DenStream
Clustering algorithm for large-scale data	K-means, BIRCH, CLARA, CURE, DBSCAN, DENCLUE, Wavecluster, FC

Table 15 Time complexity

kernel K-means	kernel SOM	kernel FCM	SVC	MMC	MKC
(kernel) High	(kernel) High	(kernel) High	(kernel) High	(kernel) High	(kernel) High

5.1 Clustering Algorithm Based on Kernel

The basic idea of this kind of clustering algorithms is that data in the input space is transformed into the feature space of high dimension by the nonlinear mapping for the cluster analysis. The typical algorithms of this kind of clustering include kernel K-means [76], kernel SOM [77], kernel FCM [78], SVC [79], MMC [80] and MKC [81]. The basic idea of kernel K-means, kernel SOM and kernel FCM is to take advantage of the kernel method and the original clustering algorithm, transforming the original data into a high dimensional feature space by nonlinear kernel function in order to carry out the original clustering algorithm. The core idea of SVC is to find the sphere with the minimum radius that can cover all the data point in the high dimensional feature space, then map the sphere back into the original data space to form the isoline, namely the border of clusters, covering the data, and the data in the closed isoline should belong to the same cluster. MMC tries to find the hyperplane with the maximum margin to cluster and it can be promoted for the multi-label clustering problem. MKC, an improvement of MMC, tries to find the best hyperplane based on several kernels to cluster. MMC and MKC share the limitation of computation to a degree.

For more detailed information about this kind of clustering algorithms, you can refer to [82–84].

Analysis:

(1) Time complexity (Table 15):
(2) The time complexity of this kind of clustering algorithms is generally high for the kernel involved in the algorithm;
(3) Advantages: more easy to cluster in the high dimensional feature space, suitable for data with arbitrary shape, able to analyze the noise and separate the overlapping clusters, and not needed to have the preliminary knowledge about the topology of data;
(4) Disadvantages: the clustering result sensitive to the type of kernel and its parameters, time complexity being high, and not suitable for large-scale data.

5.2 Clustering Algorithm Based on Ensemble

Clustering algorithm based on ensemble is also called ensemble clustering, of which the core idea is to generate a set of initial clustering results by a particular method and the final clustering result is got by integrating the initial clustering results. There are mainly 4 kinds of methods to get the set of initial clustering results as follows:

Table 16 Consensus functions

Name	Typical algorithm and application
Based on co-association matrix	[85]
Based on graph partition	CSPA, HGPA and MCLA [86]
Based on relabeling and voting	VM [88]
Based on the hybrid model	[89]
Based on information theory	[90]
Based on genetic algorithm	HCE [91]
Based on local adaptation	LAC [92]
Based on kernel method	WPCK [93]
Based on fuzzy theory	sCSPA, sMCLA and sHBGPA [94]

(1) For the same data set, employ the same algorithm with the different parameters or the different initial conditions [85];
(2) For the same data set, employ the different algorithms [86];
(3) For the subsets, carry out the clustering respectively [86];
(4) For the same data set, carry out the clustering in different feature spaces based on different kernels [87].

The initial clustering results are integrated by means of the consensus function. The consensus functions can be divided into the following 9 categories, summarized in Table 16:

For more detailed information about this kind of clustering algorithms, you can refer to [95].

Analysis:

(1) The time complexity of this kind of algorithm is based on the specific method and algorithms involved in the algorithm;
(2) Advantages: robust, scalable, able to be parallel and taking advantage of the strengths of the involved algorithms;
(3) Disadvantages: inadequate understanding about the difference among the initial clustering results, existing deficiencies of the design of the consensus function.

5.3 Clustering Algorithm Based on Swarm Intelligence

The basic idea of this kind of clustering algorithms is to simulate the changing process of the biological population. Typical algorithms include the 4 main categories: ACO_based [96,97], PSO_based [97,98], SFLA_based [99] and ABC_based [100]. The core idea of LF [101], the typical algprithm of the ACO_based, is that data is distributed randomly on the grid of two dimensions first, then the data is selected or not for further operation based on the decision of an ant and this process is iterated until a satisfactory clustering result is got. The PSO_based algorithms regard the data point as a particle. The initial clusters of particles is got by the other clustering algorithm first, then the clusters of particles is updated continuously based on the center

Table 17 Time complexity

ACO_based (LF)	PSO_based	SFLA_based	ABC_based
High	High	High	High

of clusters and the location and speed of each particle, until a satisfactory clustering result is got. The core idea of the SFLA_based algorithms is to simulate the information interaction of frogs and taking advantage of the local search and the global information interaction. The core idea of the ABC_based algorithms is to simulate the foraging behavior of three types of bee, of which the duty is to determine the food source, in a bee population and making use of the exchange of local information and global information for clustering.

For more detailed information about this kind of clustering algorithms, you can refer to [102–104].

Analysis:

(1) Time complexity (Table 17):
(2) The time complexity of this kind of algorithm is high, mainly for the large number of iterations;
(3) Advantages: algorithm with the character of overcoming being easily drawn into local optimal and getting the global optimal, easy to understand the algorithm;
(4) Disadvantages: low scalability, low operating efficiency and not suitable for high dimensional or large-scale data.

5.4 Clustering Algorithm Based on Quantum Theory

The clustering algorithm based on quantum theory is called quantum clustering, of which the basic idea is to study the distribution law of sample data in the scale space by studying the distribution law of particles in the energy field. The typical algorithms of this kind include QC [105,106] and DQC [107]. The core idea of QC (quantum clustering), suitable for high dimensional data, is to get the potential energy of each object by Schrodinger Equation using the iterative gradient descent algorithm, regard the object with low potential energy as the center of the cluster, and put the objects into different clusters by the defined distance function. DQC, an improvement of QC, adopts the time-based Schrodinger Equation in order to study the change of the original data set and the structure of the quantum potential energy function dynamically.

For more detailed information about this kind of clustering algorithms, you can refer to [108–110].

Analysis:

(1) Time complexity (Table 18):
(2) The time complexity of QC is high, for the process of solving the Schrodinger Equation and the large number of iterations;
(3) The time complexity of DQC which is more practical compared with DQ, is middle for the process of solving the Schrodinger Equation;

Table 18 Time complexity

QC	DQC
(Schrodinger Equation + a large number of iterations)	(Schrodinger Equation)
High	Middle

(4) Advantages: the number of parameters involved in this kind of algorithm being small, the determination of the center of a cluster based on the potential information of sample data;
(5) Disadvantages: the clustering result sensitive to the parameters of the algorithm, the algorithm model not able to describe the change law of data completely.

5.5 Clustering Algorithm Based on Spectral Graph Theory

The basic idea of this kind of clustering algorithms is to regard the object as the vertex and the similarity among objects as the weighted edge in order to transform the clustering problem into a graph partition problem. And the key is to find a method of graph partition making the weight of connection between different groups small as much as possible and the total weight of connection among the edges within the same group high as much as possible [111]. The typical algorithms of this kind of clustering can be mainly divided into two categories, recursive spectral and multiway spectral and the typical algorithms of this two categories are SM [50] and NJW [51] respectively. The core idea of SM which is usually used for image segmentation is to minimize Normalized Cut by heuristic method, based on the eigenvector. And NJW carries out the clustering analysis in the feature space constructed by the eigenvectors corresponding to the k largest eigenvalues of the Laplacian matrix.

For more detailed information about this kind of clustering algorithms, you can refer to [51,84,112–114].

Analysis:

(1) Time complexity (Table 19):
(2) The time complexity of SM is high, for the process of figuring out the eigenvectors and the heuristic method involved in the algorithm;
(3) The time complexity of NJW is high, for the process of figuring out the eigenvectors;
(4) Advantages: suitable for the data set with arbitrary shape and high dimension, converged to the global optimal, only the similarity matrix needed as the input, and not sensitive to the outliers;
(5) Disadvantages: the clustering result sensitive to the scaling parameter, time complexity relatively high, unclear about the construction of similarity matrix, the selection of eigenvector not optimized and the number of clusters needed to be preset.

Table 19 Time complexity

SM	NJW
(Eigenvector + heuristic method)	(Eigenvector)
High	High

5.6 Clustering Algorithm Based on Affinity Propagation

AP (affinity propagation clustering) is a significant algorithm, which was proposed in Science in 2007. The core idea of AP is to regard all the data points as the potential cluster centers and the negative value of the Euclidean distance between two data points as the affinity. So, the sum of the affinity of one data point for other data points is bigger, the probability of this data point to be the cluster center is higher. AP algorithm takes the greedy strategy which maximizes the value of the global function of the clustering network during every iteration [15].

For more detailed information about this kind of clustering algorithms, you can refer to [115–117].

Analysis:

(1) The time complexity of AP is O(nˆ2*logn);
(2) Advantages: simply and clear algorithm idea, insensitive to the outliers and the number of clusters not needed to be preset;
(3) Disadvantages: high time complexity, not suitable for very large data set, and the clustering result sensitive to the parameters involved in AP algorithm.

5.7 Clustering Algorithm Based on Density and Distance

DD (Density and distance-based clustering) is another significant clustering algorithm proposed in Science in 2014 [118], of which the core idea is novel. And the main characteristic of DD is for the description of the cluster center, which is shown as follows:

(1) with high local density: the number of data points near the cluster center within a certain scope must be big enough;
(2) away from other data points with high local density: cluster center must be away from other data points that could be the center of a cluster.

The core idea of DD is to figure out, based on the distance function, the local density of each data point and the shortest distance among each data point and other data points with higher local density in order to construct the decision graph first, select the cluster centers based on the decision graph then, and put the remaining data points into the nearest cluster with higher local density at last.

Analysis:

(1) The time complexity of DD is O(nˆ2);
(2) Advantages: simply and clear algorithm idea, suitable for the data set with arbitrary shape and insensitive to the outliers;

Table 20 Time complexity

DBSCAN	STING	Wavecluster	CLARANS
O(n*logn)	O(n)	O(n)	O(nˆ2)
Middle	Low	Low	High

(3) Disadvantages: relatively high time complexity, relatively strong subjectivity for the selection of the cluster center based on the decision graph and the clustering result sensitive to the parameters involved in DD algorithm.

5.8 Clustering Algorithm for Spatial Data

Spatial data refers to the data with the two dimensions, time and space, at the same time, sharing the characteristics of large in scale, high in speed and complex in information. The typical algorithms of this kind of clustering include DBSCAN [36], STING [52], Wavecluster [54] and CLARANS [11]. The core idea of Wavecluster which can be used for parallel processing is to carry out the clustering in the new feature space by applying the Wavelet Transform to the original data. And the core idea of CLARANS is to sample based on CLARA [10] and carry out clustering by PAM [9]. DBSCAN has been discussed in the section Clustering algorithm based on density and STING has been discussed in the section Clustering algorithm based on grid.

For more detailed information about this kind of clustering algorithms, you can refer to [119–122], ST-DBSCAN [123].

Time complexity (Table 20):

5.9 Clustering Algorithm for Data Stream

Data stream shares the characteristics of arriving based on sequence, large in scale and limited frequency of reading. The typical algorithms of this kind of clustering include STREAM [124], CluStream [125], HPStream [126], DenStream [127] and the latter three are incremental algorithms. STREAM, based on the idea of divide and conquer, deals with the data successively according to the sequence of data arriving in order to construct the hierarchical clustering structure. CluStream, which mainly deals with the shortcoming of STREAM that only describing the original data statically, regards data as a dynamic changing process. So CluStream can not only give the timely response for a request, but it also gives the clustering result in terms of different time granularities by figuring out the Micro-clusters online and offline. HPStream, an improvement of CluStream, takes the attenuation of data's influence over time into consideration and is more suitable for clustering data with high dimension. DenStream, which takes the core idea of the clustering algorithm based on density, is suitable for the nonconvex data set and can deal with outliers efficiently, compared with the algorithms mentioned above in this section.

For more detailed information about this kind of clustering algorithms, you can refer to [128–131], D-Stream [41,132].

Time complexity (Table 21):

The time complexity of CluStream, HPStream and DenStream is involved in the online and offline processes.

Table 21 Time complexity

STREAM	CluStream	HPStream	DenStream
O(kn)	(online and offline processes)		
Low	Low		

5.10 Clustering Algorithm for Large-Scale Data

Big data shares the characteristics of 4 V's, large in volume, rich in variety, high in velocity and doubt in veracity [133]. The main basic ideas of clustering for big data can be summarized in the following 4 categories:

(1) sample clustering [10,18];
(2) data merged clustering [17,134];
(3) dimension-reducing clustering [135,136];
(4) parallel clustering [114,137–139];

Typical algorithms of this kind of clustering are K-means [7], BIRCH [17], CLARA [10], CURE [18], DBSCAN [36], DENCLUE [43], Wavecluster [54] and FC [59].

For more detailed information about this kind of clustering algorithms, you can refer to [2,13,140,141].

The time complexity of DENCLUE is O(nlogn) and the complexities of K-means, BIRCH, CLARA, CURE, DBSCAN, Wavecluster and FC have been described before in other sections.

6 Conclusions

This paper starts at the basic definitions of clustering and the typical procedure, lists the commonly used distance (dissimilarity) functions, similarity functions, and evaluation indicators that lay the foundation of clustering, and analyzes the clustering algorithms from two perspectives, the traditional ones that contain 9 categories including 26 algorithms and the modern ones that contain 10 categories including 45 algorithms. The detailed and comprehensive comparisons of all the discussed clustering algorithms are summarized in Appendix Table 22.

The main purpose of the paper is to introduce the basic and core idea of each commonly used clustering algorithm, specify the source of each one, and analyze the advantages and disadvantages of each one. It is hard to present a complete list of all the clustering algorithms due to the diversity of information, the intersection of research fields and the development of modern computer technology. So 19 categories of the commonly used clustering algorithms, with high practical value and well studied, are selected and one or several typical algorithm(s) of each category is(are) discussed in detail so as to give readers a systematical and clear view of the important data analysis method, clustering.

Acknowledgments This work has been partially supported by grants form National Natural Science Foundations of China (Nos. 61472390, 11271361, 71331005, and 11226089), Major International (Regional) Joint Research Project (No. 71110107026).

Appendix

Table 22 The detailed and comprehensive comparisons of all the discussed clustering algorithms

Category	Typical algorithm	Complexity (time)	Scalability	For large-scale data	For high dimensional data	Shape of suitable data set	Sensitive to the sequence of inputting data	Sensitive to noise/outlier	References
Based on partition	***K-means***	Low 0(knt)	Middle	Yes	no	Convex	Highly	Highly	[7]
	K-medoids	High O(k(n-k)^2)	Low	No	No	Convex	Moderately	Little	[8]
	PAM	High 0(k^3*n^2)	Low	No	No	Convex	Moderately	Little	[9]
	CLARA	Middle 0(ks^2+k(n-k))	High	Yes	No	Convex	Moderately	Little	[10]
	CLARANS	High 0(n^2)	Middle	Yes	No	Convex	Highly	Little	[11]
	AP	*	*	*	*	*	*	*	[15]
Based on hierarchy	***BIRCH***	Low 0(n)	High	Yes	No	Convex	Moderately	Little	[17]
	CURE	Low 0(s^2*logs),	High	Yes	Yes	Arbitrary	Moderately	Little	[18]
	ROCK	High 0(n^2*logn)	Middle	No	Yes	Arbitrary	Moderately	Little	[19]
	Chameleon	High 0(n^2)	High	No	No	Arbitrary	Moderately	Little	[20]
Based on fuzzy theory	***FCM***	Low 0(n)	Middle	No	No	Convex	Moderately	Highly	[23–25]
	FCS	High (kernel)	Low	No	No	Arbitrary	Moderately	Highly	[26]
	MM	Middle 0(v^2*n)	Low	No	No	Arbitrary	Moderately	Little	[27]

Table 22 continued

Category	Typical algorithm	Complexity (time)	Scalability	For large-scale data	For high dimensional data	Shape of suitable data set	Sensitive to the sequence of inputting data	Sensitive to noise/outlier	References
Based on distribution	***DBCLASD***	Middle 0(n*logn)	Middle	Yes	Yes	Arbitrary	Little	Little	[31]
	GMM	High 0(nˆ2*kt)	High	No	No	Arbitrary	Highly	Little	[32]
Based on density	***DBSCAN***	Middle 0(n*logn)	Middle	Yes	No	Arbitrary	Moderately	Little	[36]
	OPTICS	Middle 0(n*logn)	Middle	Yes	No	Arbitrary	Little	Little	[37]
	Mean-shift	High (kernel)	Low	No	No	Arbitrary	Little	Little	[38]
	DENCLUE	*	*	*	*	*	*	*	[43]
Based on graph theory	***CLICK***	Low 0(k*f(v,e))	High	Yes	No	Arbitrary	Highly	Highly	[44]
	MST	Middle 0(e*logv)	High	Yes	No	Arbitrary	Highly	Highly	[45]
	SM	*	*	*	*	*	*	*	[50]
	NJW	*	*	*	*	*	*	*	[51]

Table 22 continued

Category	Typical algorithm	Complexity (time)	Scalability	For large-scale data	For high dimensional data	Shape of suitable data set	Sensitive to the sequence of inputting data	Sensitive to noise/out lier	References
Based on grid	***STING***	Low 0(n)	High	Yes	Yes	Arbitrary	Little	Little	[52]
	CLIQUE	Low 0(n+kˆ2)	High	No	Yes	Convex	Little	Moderately	[53]
	Wavecluster	*	*	*	*	*	*	*	[54]
Based on fractal theory	***FC***	Low 0(n)	High	Yes	Yes	Arbitrary	Highly	Little	[59]
Based on model	***COBWEB***	Low (distribution)	Middle	Yes	No	Arbitrary	Little	Moderately	[64]
	GMM	*	*	*	*	*	*	*	[32]
	SOM	High (layer)	Low	No	Yes	Arbitrary	Little	Little	[65]
	ART	Middle (type+layer)	High	Yes	No	Arbitrary	Highly	Highly	[66–69]
Based on kernel	***kernel K-means***	High (kernel)	Middle	No	No	Arbitrary	Moderately	Little	[76]
	kernel SOM	High (kernel)	High	No	No	Arbitrary	Little	Little	[77]
	kernel FCM	High (kernel)	Middle	No	No	Arbitrary	Moderately	Little	[78]
	SVC	High (kernel)	Low	No	No	Arbitrary	Little	Little	[79]
	MMC	High (kernel)	Low	No	No	Arbitrary	Little	Little	[80]
	MKC	High (kernel)	Low	No	No	Arbitrary	Little	Little	[81]
Based on ensemble	NA	NA	NA	NA	NA	NA	NA	NA	[85–94]

Table 22 continued

Category	Typical algorithm	Complexity (time)	Scalability	For large-scale data	For high dimensional data	Shape of suitable data set	Sensitive to the sequence of inputting data	Sensitive to noise/out lier	References
Based on swarm intelligence	*ACO_based (LF)*	High (iterations)	Low	No	No	Arbitrary	Little	Highly	[101]
	PSO_based	High (iterations)	Low	No	No	Arbitrary	Moderately	moderately	[98]
	SFLA_based	High (iterations)	Low	No	No	Arbitrary	Moderately	moderately	[97,99]
	ABC_based	High (iterations)	Low	No	No	Arbitrary	Moderately	moderately	[100]
Based on quantum theory	*QC*	High (Schrodinger Equation +iterations)	Middle	No	No	Convex	Little	Little	[105,106]
	DQC	Middle (Schrodinger Equation)	Middle	No	No	Convex	Little	Little	[107]
Spectral clustering	*SM*	High (eigenvector +heuristics)	Middle	No	Yes	Arbitrary	Little	Little	[50]
	NJW	High (eigenvector)	Middle	No	Yes	Arbitrary	Little	Little	[51]
Based on affinity propagation	*AP*	High 0(nˆ2*logn)	Low	No	No	Convex	Moderately	Little	[15]
Based on density and distance	*DD*	High 0(nˆ2)	Low	No	No	Arbitrary	Little	Little	[118]

Table 22 continued

Category	**Typical algorithm**	**Complexity (time)**	**Scalability**	**For large-scale data**	**For high dimensional data**	**Shape of suitable data set**	**Sensitive to the sequence of inputting data**	**Sensitive to noise/outlier**	**References**
For spatial data	*DBSCAN*	*	*	*	*	*	*	*	[36]
	STING	*	*	*	*	*	*	*	[52]
	Wavecluster	Low 0(n)	High	Yes	No	Arbitrary	Little		[54]
	CLARANS	*	*	*	*	*	*	*	[11]
For data stream	*STREAM*	Low 0(kn)	Middle	Yes	No	Convex	Highly	Highly	[124]
	CluStream	Low (online+offline)	High	Yes	No	Convex	Highly	Highly	[125]
	HPStream		High	Yes	Yes	Convex	Highly	Highly	[126]
	DenStream		High	Yes	No	Arbitrary	Highly	Little	[127]
For large-scale data	*K-means*	*	*	*	*	*	*	*	[7]
	BIRCH	*	*	*	*	*	*	*	[17]
	CLARA	*	*	*	*	*	*	*	[10]
	CURE	*	*	*	*	*	*	*	[18]
	DBSCAN	*	*	*	*	*	*	*	[36]
	DENCLUE	Middle 0(nlogn)	Middle	Yes	Yes	Arbitrary	Moderately	Little	[43]
	Wavecluster	*	*	*	*	*	*	*	[54]
	FC	*	*	*	*	*	*	*	[59]

Label **NA** in the row of algorithm *based on ensemble* indicates that the evaluation value depends on the specific selected method/model/algorithm
Label * indicates that this algorithm has been or will be discussed in other section

References

1. Jain A, Dubes R (1988) Algorithms for clustering data. Prentice-Hall, Inc, Upper Saddle River
2. Xu R, Wunsch D (2005) Survey of clustering algorithms. IEEE Trans Neural Netw 16:645–678
3. Everitt B, Landau S, Leese M (2001) Clustering analysis, 4th edn. Arnold, London
4. Gower J (1971) A general coefficient of similarity and some of its properties. Biometrics 27:857–871
5. Estivill-Castro V (2002) Why so many clustering algorithms: a position paper. ACM SIGKDD Explor Newsl 4:65–75
6. Färber I, Günnemann S, Kriegel H, Kröger P, Müller E, Schubert E, Seidl T, Zimek A (2010) On using class-labels in evaluation of clusterings. In MultiClust: 1st international workshop on discovering, summarizing and using multiple clusterings held in conjunction with KDD, Washington, DC
7. MacQueen J (1967) Some methods for classification and analysis of multivariate observations. Proc Fifth Berkeley Symp Math Stat Probab 1:281–297
8. Park H, Jun C (2009) A simple and fast algorithm for K-medoids clustering. Expert Syst Appl 36:3336–3341
9. Kaufman L, Rousseeuw P (1990) Partitioning around medoids (program pam). Finding groups in data: an introduction to cluster analysis. Wiley, Hoboken
10. Kaufman L, Rousseeuw P (2008) Finding groups in data: an introduction to cluster analysis, vol 344. Wiley, Hoboken. doi:10.1002/9780470316801
11. Ng R, Han J (2002) Clarans: a method for clustering objects for spatial data mining. IEEE Trans Knowl Data Eng 14:1003–1016
12. Boley D, Gini M, Gross R, Han E, Hastings K, Karypis G, Kumar V, Mobasher B, Moore J (1999) Partitioning-based clustering for web document categorization. Decis Support Syst 27:329–341
13. Jain A (2010) Data clustering: 50 years beyond K-means. Pattern Recognit Lett 31:651–666
14. Velmurugan T, Santhanam T (2011) A survey of partition based clustering algorithms in data mining: an experimental approach. Inf Technol J 10:478–484
15. Frey BJ, Dueck D (2007) Clustering by passing messages between data points. Science 315(5814):972–976
16. Johnson S (1967) Hierarchical clustering schemes. Psychometrika 32:241–254
17. Zhang T, Ramakrishnan R, Livny M (1996) BIRCH: an efficient data clustering method for very large databases. ACM SIGMOD Rec 25:103–104
18. Guha S, Rastogi R, Shim K (1998) CURE: an efficient clustering algorithm for large databases. ACM SIGMOD Rec 27:73–84
19. Guha S, Rastogi R, Shim K (1999) ROCK: a robust clustering algorithm for categorical attributes. In: Proceedings of the 15th international conference on data engineering, pp 512-521
20. Karypis G, Han E, Kumar V (1999) Chameleon: hierarchical clustering using dynamic modeling. Computer 32:68–75
21. Murtagh F (1983) A survey of recent advances in hierarchical clustering algorithms. Comput J 26:354–359
22. Carlsson G, Mémoli F (2010) Characterization, stability and convergence of hierarchical clustering methods. J Mach Learn Res 11:1425–1470
23. Dunn J (1973) A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. J Cybern 3:32–57
24. Bezdek J (1981) Pattern recognition with fuzzy objective function algorithms. Plenum, New York
25. Bezdek J, Ehrlich R, Full W (1984) FCM: the fuzzy c-means clustering algorithm. Comput Geosci 10:191–203
26. Dave R, Bhaswan K (1992) Adaptive fuzzy c-shells clustering and detection of ellipses. IEEE Trans Neural Netw 3:643–662
27. Yager R, Filev D (1994) Approximate clustering via the mountain method. IEEE Trans Syst Man Cybern 24:1279–1284
28. Yang M (1993) A survey of fuzzy clustering. Math Comput Model 18:1–16
29. Baraldi A, Blonda P (1999) A survey of fuzzy clustering algorithms for pattern recognition. I. IEEE Trans Syst Man Cybern Part B 29:778–785
30. Höppner F (1999) Fuzzy cluster analysis: methods for classification, data analysis and image recognition. Wiley, Hoboken

31. Xu X, Ester M, Kriegel H, Sander J (1998) A distribution-based clustering algorithm for mining in large spatial databases. In: Proceedings of the fourteenth international conference on data engineering, pp 324-331
32. Rasmussen C (1999) The infinite Gaussian mixture model. Adv Neural Inf Process Syst 12:554–560
33. Preheim S, Perrotta A, Martin-Platero A, Gupta A, Alm E (2013) Distribution-based clustering: using ecology to refine the operational taxonomic unit. Appl Environ Microbiol 79:6593–6603
34. Jiang B, Pei J, Tao Y, Lin X (2013) Clustering uncertain data based on probability distribution similarity. IEEE Trans Knowl Data Eng 25:751–763
35. Kriegel H, Kröger P, Sander J, Zimek A (2011) Densitybased clustering. Wiley Interdiscip Rev 1:231–240
36. Ester M, Kriegel H, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the second ACM SIGKDD international conference on knowledge discovery and data mining, pp 226–231
37. Ankerst M, Breunig M, Kriegel H, Sander J (1999) OPTICS: ordering points to identify the clustering structure. In: Proceedings on 1999 ACM SIGMOD international conference on management of data, vol 28, pp 49–60
38. Comaniciu D, Meer P (2002) Mean shift: a robust approach toward feature space analysis. IEEE Trans Pattern Anal Mach Intell 24:603–619
39. Januzaj E, Kriegel H, Pfeifle M (2004) Scalable density-based distributed clustering. In: Proceedings of the 8th european conference on principles and practice of knowledge discovery in databases, pp 231–244
40. Kriegel H, Pfeifle M (2005) Density-based clustering of uncertain data. In: Proceedings of the eleventh ACM SIGKDD international conference on knowledge discovery in data mining, pp 672–677
41. Chen Y, Tu L (2007) Density-based clustering for real-time stream data. In: Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining, pp 133–142
42. Duan L, Xu L, Guo F, Lee J, Yan B (2007) A local-density based spatial clustering algorithm with noise. Inf Syst 32:978–986
43. Hinneburg A, Keim D (1998) An efficient approach to clustering in large multimedia databases with noise. In Proceedings of the 4th ACM SIGKDD international conference on knowledge discovery and data mining 98: 58–65
44. Sharan R, Shamir R (2000) CLICK: a clustering algorithm with applications to gene expression analysis. In: Proc international conference intelligent systems molecular biolgy, pp 307–316
45. Jain A, Murty M, Flynn P (1999) Data clustering: a review. ACM Comput Surv (CSUR) 31:264–323
46. Ben-Dor A, Shamir R, Yakhini Z (1999) Clustering gene expression patterns. J Comput Biol 6:281–297
47. Hartuv E, Shamir R (2000) A clustering algorithm based on graph connectivity. Inf Process Lett 76:175–181
48. Estivill-Castro V, Lee I (2000) Amoeba: hierarchical clustering based on spatial proximity using delaunay diagram. In: Proceedings of the 9th international symposium on spatial data handling, Beijing
49. Cherng J, Lo M (2001) A hypergraph based clustering algorithm for spatial data sets. In: Proceedings of the 2001 IEEE international conference on data mining, pp 83–90
50. Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell 22:888–905
51. Ng A, Jordan M, Weiss Y (2002) On spectral clustering: analysis and an algorithm. Adv Neural Inf Process Syst 2:849–856
52. Wang W, Yang J, Muntz R (1997) STING: a statistical information grid approach to spatial data mining. In VLDB, pp 186–195
53. Agrawal R, Gehrke J, Gunopulos D, Raghavan P (1998) Automatic subspace clustering of high dimensional data for data mining applications. In: Proceedings 1998 ACM sigmod international conference on management of data, vol 27, pp 94–105
54. Sheikholeslami G, Chatterjee S, Zhang A (1998) Wavecluster: A multi-resolution clustering approach for very large spatial databases. In: VLDB, pp 428–439
55. Ma E, Chow T (2004) A new shifting grid clustering algorithm. Pattern Recognit 37:503–514
56. Park N, Lee W (2004) Statistical grid-based clustering over data streams. ACM SIGMOD Rec 33:32–37

57. Pilevar A, Sukumar M (2005) GCHL: a grid-clustering algorithm for high-dimensional very large spatial data bases. Pattern Recognit Lett 26:999–1010
58. Mandelbrot B (1983) The fractal geometry of nature. Macmillan, London
59. Barbará D, Chen P (2000) Using the fractal dimension to cluster datasets. In: Proceedings of the sixth ACM SIGKDD international conference on knowledge discovery and data mining, pp 260–264
60. Zhang A, Cheng B, Acharya R (1996) A fractal-based clustering approach in large visual database systems. In Representation and retrieval of visual media in, multimedia systems, pp 49–68
61. Menascé D, Abrahao B, Barbará D, Almeida V, Ribeiro F (2002) Fractal characterization of web workloads. In: Proceedings of the " Web Engineering" Track of WWW2002, pp 7–11
62. Barry R, Kinsner W (2004) Multifractal characterization for classification of network traffic. Conf Electr Comput Eng 3:1453–1457
63. Al-Shammary D, Khalil I, Tari Z (2014) A distributed aggregation and fast fractal clustering approach for SOAP traffic. J Netw Comput Appl 41:1–14
64. Fisher D (1987) Knowledge acquisition via incremental conceptual clustering. Mach Learn 2:139–172
65. KohonenKohonen T (1990) The self-organizing map. Proc IEEE 78:1464–1480
66. Carpenter G, Grossberg S (1987) A massively parallel architecture for a self-organizing neural pattern recognition machine. Comput Vis Gr Image Process 37:54–115
67. Carpenter G, Grossberg S (1988) The ART of adaptive pattern recognition by a self-organizing neural network. Computer 21:77–88
68. Carpenter G, Grossberg S (1987) ART 2: self-organization of stable category recognition codes for analog input patterns. Appl Opt 26:4919–4930
69. Carpenter G, Grossberg S (1990) ART 3: hierarchical search using chemical transmitters in self-organizing pattern recognition architectures. Neural Netw 3:129–152
70. Meilă M, Heckerman D (2001) An experimental comparison of model-based clustering methods. Mach Learn 42:9–29
71. Fraley C, Raftery A (2002) Model-based clustering, discriminant analysis, and density estimation. J Am Stat Assoc 97:611–631
72. McLachlan G, Bean R, Peel D (2002) A mixture model-based approach to the clustering of microarray expression data. Bioinformatics 18:413–422
73. Medvedovic M, Sivaganesan S (2002) Bayesian infinite mixture model based clustering of gene expression profiles. Bioinformatics 18:1194–1206
74. Zhong S, Ghosh J (2003) A unified framework for model-based clustering. J Mach Learn Res 4:1001–1037
75. McNicholas P, Murphy T (2010) Model-based clustering of microarray expression data via latent Gaussian mixture models. Bioinformatics 26:2705–2712
76. Schölkopf B, Smola A, Müller K (1998) Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput 10:1299–1319
77. MacDonald D, Fyfe C (2000) The kernel self-organising map. Proc Fourth Int Conf Knowl-Based Intell Eng Syst Allied Technol 1:317–320
78. Wu Z, Xie W, Yu J (2003) Fuzzy c-means clustering algorithm based on kernel method. In: Proceedings of the fifth ICCIMA, pp 49–54
79. Ben-Hur A, Horn D, Siegelmann H, Vapnik V (2002) Support vector clustering. J Mach Learn Res 2:125–137
80. Xu L, Neufeld J, Larson B, Schuurmans D (2004) Maximum margin clustering. In: Advances in neural information processing systems, pp 1537–1544
81. Zhao B, Kwok J, Zhang C (2009) Multiple kernel clustering. In SDM, pp 638–649
82. Müller K, Mika S, Rätsch G, Tsuda K, Schölkopf B (2001) An introduction to kernel-based learning algorithms. IEEE Trans Neural Netw 12:181–201
83. Girolami M (2002) Mercer kernel-based clustering in feature space. IEEE Trans Neural Netw 13:780–784
84. Filippone M, Camastra F, Masulli F, Rovetta S (2008) A survey of kernel and spectral methods for clustering. Pattern Recognit 41:176–190
85. Fred A, Jain A (2005) Combining multiple clusterings using evidence accumulation. IEEE Trans Pattern Anal Mach Intell 27:835–850
86. Strehl A, Ghosh J (2003) Cluster ensembles—a knowledge reuse framework for combining multiple partitions. J Mach Learn Res 3:583–617

87. Fern X, Brodley C (2003) Random projection for high dimensional data clustering: a cluster ensemble approach. ICML 3:186–193
88. Dimitriadou E, Weingessel A, Hornik K (2001) Voting-merging: an ensemble method for clustering. In: ICANN, pp 217–224
89. Topchy A, Jain A, Punch W (2004) A mixture model for clustering ensembles. In: Proceedings of the SIAM international conference on data mining, pp 379
90. Topchy A, Jain A, Punch W (2005) Clustering ensembles: models of consensus and weak partitions. IEEE Trans Pattern Anal Mach Intell 27:1866–1881
91. Yoon H, Ahn S, Lee S, Cho S, Kim J (2006) Heterogeneous clustering ensemble method for combining different cluster results. In: Data mining for biomedical applications, pp 82–92
92. Domeniconi C, Gunopulos D, Ma S, Yan B, Al-Razgan M, Papadopoulos D (2007) Locally adaptive metrics for clustering high dimensional data. Data Min Knowl Discov 14:63–97
93. Vega-Pons S, Correa-Morris J, Ruiz-Shulcloper J (2010) Weighted partition consensus via kernels. Pattern Recognit 43:2712–2724
94. Punera K, Ghosh J (2008) Consensus-based ensembles of soft clusterings. Appl Artif Intell 22:780–810
95. Vega-Pons S, Ruiz-Shulcloper J (2011) A survey of clustering ensemble algorithms. Int J Pattern Recognit Artif Intell 25:337–372
96. Handl J, Meyer B (2007) Ant-based and swarm-based clustering. Swarm Intell 1:95–113
97. Abraham A, Das S, Roy S (2008) Swarm intelligence algorithms for data clustering. In: Soft computing for knowledge discovery and data mining, pp 279–313
98. Van der Merwe D, Engelbrecht A (2003) Data clustering using particle swarm optimization. Congr Evol Comput 1:215–220
99. Amiri B, Fathian M, Maroosi A (2009) Application of shuffled frog-leaping algorithm on clustering. Int J Adv Manuf Technol 45:199–209
100. Karaboga D, Ozturk C (2011) A novel clustering approach: artificial bee colony (ABC) algorithm. Appl Soft Comput 11:652–657
101. Lumer E, Faieta B (1994) Diversity and adaptation in populations of clustering ants. Proc Third Int Conf Simul Adapt Behav 3:501–508
102. Shelokar P, Jayaraman V, Kulkarni B (2004) An ant colony approach for clustering. Anal Chim Acta 509:187–195
103. Karaboga D, Akay B (2009) A survey: algorithms simulating bee swarm intelligence. Artif Intell Rev 31:61–85
104. Xu R, Xu J, Wunsch D (2012) A comparison study of validity indices on swarm-intelligence-based clustering. IEEE Trans Syst Man Cybern Part B 42:1243–1256
105. Horn D, Gottlieb A (2001) Algorithm for data clustering in pattern recognition problems based on quantum mechanics. Phys Rev Lett 88:018702
106. Horn D, Gottlieb A (2001) The method of quantum clustering. In: Advances in neural information processing systems, pp 769–776
107. Weinstein M, Horn D (2009) Dynamic quantum clustering: a method for visual exploration of structures in data. Phys Rev E 80:066117
108. Horn D (2001) Clustering via Hilbert space. Phys A 302:70–79
109. Horn D, Axel I (2003) Novel clustering algorithm for microarray expression data in a truncated SVD space. Bioinformatics 19:1110–1115
110. Aïmeur E, Brassard G, Gambs S (2007) Quantum clustering algorithms. In: ICML, pp 1–8
111. Von Luxburg U (2007) A tutorial on spectral clustering. Stat Comput 17:395–416
112. Yu S, Shi J (2003) Multiclass spectral clustering. In: Proceedings of the ninth IEEE international conference on computer vision, pp 313–319
113. Verma D, Meila M (2003) A comparison of spectral clustering algorithms. University of Washington Tech Rep UWCSE030501 1: 1–18
114. Chen W, Song Y, Bai H, Lin C, Chang E (2011) Parallel spectral clustering in distributed systems. IEEE Trans Pattern Anal Mach Intell 33:568–586
115. Lu Z, Carreira-Perpinan M (2008) Constrained spectral clustering through affinity propagation. In: IEEE conference on computer vision and pattern recognition, pp 1–8
116. Givoni I, Frey B (2009) A binary variable model for affinity propagation. Neural Comput 21:1589–1600

117. Shang F, Jiao L, Shi J, Wang F, Gong M (2012) Fast affinity propagation clustering: a multilevel approach. Pattern Recognit 45:474–486
118. Rodriguez A, Laio A (2014) Clustering by fast search and find of density peaks. Science 344:1492–1496
119. Ng R, Han J (1994) Efficient and effective clustering methods for spatial data mining. In: VLDB, pp 144–155
120. Sander J, Ester M, Kriegel H, Xu X (1998) Density-based clustering in spatial databases: the algorithm gdbscan and its applications. Data Min Knowl Discov 2:169–194
121. Harel D, Koren Y (2001) Clustering spatial data using random walks. In: Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining, pp 281–286
122. Zaïane O, Lee C (2002) Clustering spatial data when facing physical constraints. In: Proceedings of the IEEE international conference on data mining, pp 737–740
123. Birant D, Kut A (2007) ST-DBSCAN: an algorithm for clustering spatial-temporal data. Data Knowl Eng 60:208–221
124. O'callaghan L, Meyerson A, Motwani R, Mishra N, Guha S (2002) Streaming-data algorithms for high-quality clustering. In: ICDE, p 0685
125. Aggarwal C, Han J, Wang J, Yu P (2003) A framework for clustering evolving data streams. In: VLDB, pp 81–92
126. Aggarwal C, Han J, Wang J, Yu P (2004) A framework for projected clustering of high dimensional data streams. In: VLDB, pp 852–863
127. Cao F, Ester M, Qian W, Zhou A (2006) Density-based clustering over an evolving data stream with noise. SDM 6:328–339
128. Guha S, Mishra N, Motwani R, O'Callaghan L (2000) Clustering data streams. In: Proceedings of the 41st annual symposium on foundations of computer science, pp 359–366
129. Barbará D (2002) Requirements for clustering data streams. ACM SIGKDD Explor Newsl 3:23–27
130. Guha S, Meyerson A, Mishra N, Motwani R, O'Callaghan L (2003) Clustering data streams: theory and practice. IEEE Trans Knowl Data Eng 15:515–528
131. Beringer J, Hüllermeier E (2006) Online clustering of parallel data streams. Data Knowl Eng 58:180–204
132. Silva J, Faria E, Barros R, Hruschka E, de Carvalho A, Gama J (2013) Data stream clustering: a survey. ACM Comput Surv 46:13
133. Leskovec J, Rajaraman A, Ullman JD (2014) Mining massive datasets. Cambridge University Press, Cambridge
134. Steinbach M, Karypis G, Kumar V (2000) A comparison of document clustering techniques. KDD Workshop Text Min 400:525–526
135. Parsons L, Haque E, Liu H (2004) Subspace clustering for high dimensional data: a review. ACM SIGKDD Explor Newsl 6:90–105
136. Kriegel H, Kröger P, Zimek A (2009) Clustering high-dimensional data: a survey on subspace clustering, pattern-based clustering, and correlation clustering. ACM Trans Knowl Discov Data 3:1
137. Judd D, McKinley P, Jain A (1996) Large-scale parallel data clustering. In: Proceedings of the 13th international conference on pattern recognition, vol 4, pp 488–493
138. Tasoulis D, Vrahatis M (2004) Unsupervised distributed clustering. In: Parallel and distributed computing and networks, pp 347–351
139. Zhao W, Ma H, He Q (2009) Parallel k-means clustering based on mapreduce. In: Cloud computing, pp 674–679
140. Herwig R, Poustka A, Müller C, Bull C, Lehrach H, O'Brien J (1999) Large-scale clustering of cDNA-fingerprinting data. Genome Res 9:1093–1105
141. Hinneburg A, Keim D (2003) A general approach to clustering in large databases with noise. Knowl Inf Syst 5:387–415