Analyzing Subspace Clustering Approaches for High Dimensional Data

Parul Agarwal and Shikha Mehta

1 Introduction

A huge body of knowledge can be exploited through clustering techniques. Clustering algorithms make use of knowledge present in input data. It uncovers the hidden patterns existing in data. Clustering is one of the effective data mining techniques for grouping similar data objects [1]. The grouping is made on the basis of attributes. With the tremendous growth in the technology of information sciences, huge data is being produced. Such data are sensor data, web data, bioinformatics data and many more. The large volume of data not only includes a large number of rows but also a large number of attributes. The datasets with many columns/features are termed as high dimensional data. Clustering high dimensional data poses a number of computational challenges [2]. Traditional clustering algorithms' efficacy degrades on high dimensional data. This problem is termed as "curse of dimensionality." This is because traditional clustering algorithms find clusters in all dimension space. However, clusters might exist in a few subsets of dimensions. All dimensions might not be important for all clusters. Considering all dimensions for clustering might hide the relevant dimensions. Another challenge is the distance measure. Distance between data objects becomes meaningless in high dimensional data [3]. All objects appear equidistant from each other. Most of the traditional clustering algorithms use distance measure for grouping data objects. New techniques are developed to find relevant dimensions and overcome challenges of high dimensional clustering.

Relevant dimensions are selected through feature selection methods and, subsequently, selected features to participate in complete clustering method. However,

P. Agarwal (✉) · S. Mehta
Department of Computer Science and Information Technology, Jaypee Institute
of Information Technology, Noida, India
e-mail: parul.agarwal@jiit.ac.in; shikha.mehta@jiit.ac.in

S. Awasthi et al. (eds.), *Artificial Intelligence for a Sustainable Industry 4.0*,
https://doi.org/10.1007/978-3-030-77070-9_11

clusters might exist in different subspaces. Dimensions participating in one cluster might be irrelevant for another cluster. Hence to find clusters in different subspaces as shown in Fig. 1 (each circle represents one cluster), subspace clustering methods are employed [4]. It is an extended version of traditional clustering algorithms. Subspace clustering algorithms confine the search in a certain manner, i.e., either top-down or bottom-up approach so that it is able to determine clusters existing in various subspaces. Further, subspace clusters can be non-overlapping or overlapping in dimensions/objects.

The significance of subspace clustering is intensifying from last two decades. Figure 2 shows the percentage of papers published on various methods of high

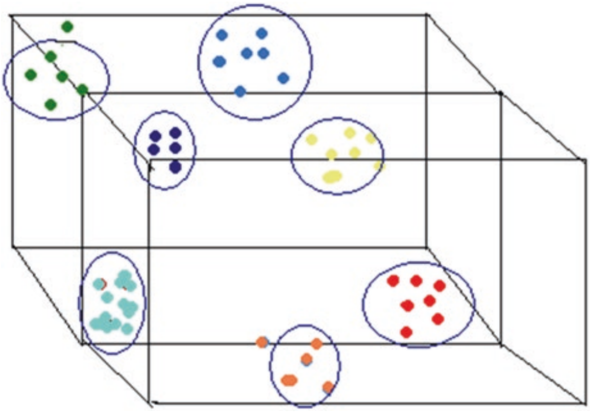

Fig. 1 Representation of clusters in subspaces

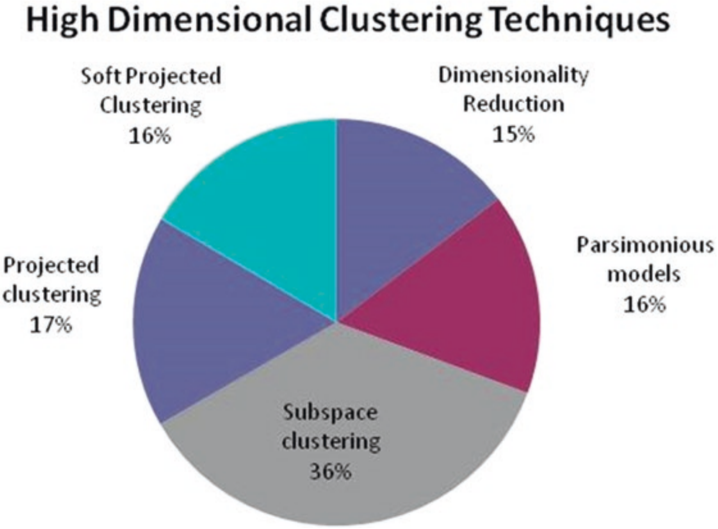

Fig. 2 Papers published on different techniques of high dimensional clustering

dimensional clustering from the year 2000 onwards. The main techniques for handling high dimensional clustering [5, 6] are as follows:

(a) Dimensionality reduction
(b) Parsimonious models
(c) Projected clustering
(d) Soft projected clustering
(e) Subspace clustering

It was observed that only 15% of work is done in high dimensional clustering through dimensionality reduction approach. Sixteen percent of papers are published for clustering high dimensional data using soft projected clustering and parsimonious models. Seventeen percent of the work is dedicated to projected clustering. The large amount of work for high dimensional clustering is being done by subspace clustering approach. This proves that subspace clustering methods are gaining considerable attention in current research.

This study is entailed to answer the critical research questions identified below.

(i) What are the major challenges faced by traditional clustering algorithms to cluster high dimensional data?
(ii) What search techniques are being used in subspace clustering to determine subspaces?
(iii) What are the evaluation measures for comparing subspace clustering algorithms?
(iv) What is the current scenario of subspace clustering?
(v) What are the research gaps in the literature and the future prospects of subspace clustering?

This chapter presents a combined review of approximately all subspace clustering algorithms belonging to different classes. Various evaluation measures required for comparing the clusters and subspaces are also presented. Additionally, statistical data of subspace clustering approaches published in different years and different repositories are provided. This chapter is targeted to researchers planning to work in subspace clustering area. It provides a roadmap of research in subspace clustering approach for high dimensional data. The chapter also presents the application areas, identifies gaps in present work, and suggests future opportunities for research in this field.

The chapter is divided into following subsections: Sect. 2 presents challenges in subspace clustering, Sect. 3 gives various classifications of subspace clustering approaches, Sect. 4 presents evaluation measure for comparing subspace clustering algorithms, and Sects. 5 and 6 depict the literature survey and empirical assessment of subspace clustering algorithms, respectively. Applications and future prospects are illustrated in Sect. 7, and Sect. 8 finally concludes the chapter.

2 Challenges in Subspace Clustering

The major goal of clustering is not only to find similar groups of data points but to find high-quality groups within a reasonable time. In cases where clusters exist in different subsets of dimensions, it is essential for clustering algorithms to determine effective clusters along with relevant dimensions. Subspace clustering algorithms have proved to be efficient for extracting clusters from high dimensional datasets [7]. However, a number of challenges persist which needs to be focused before developing an efficient subspace clustering algorithm. These challenges are as follows:

(a) It is hard to determine the subsets of dimensions where data points are similar. This is because a number of dimensions are large and possible combinations are huge.
(b) It is quite difficult to determine the distribution of data within subspaces. If data is near the cluster center and far from another subspace center, then clustering is easy; otherwise it is difficult to cluster within subspaces.
(c) There may be overlapping subspaces that mean few dimensions may be common in few subspaces. Clustering becomes even more complicated in case of overlapping subspaces.
(d) There are possibilities of noisy data in the dataset. That means some data points might not belong to any subspace or if they are part of any subspace, they are not part of a cluster of particular subspace. Handling of such data becomes a challenge for subspace clustering.
(e) It is difficult to understand which clustering algorithm and subspace strategy are appropriate for a given problem. As the number of subspaces and the dimension of each subspace is unknown, the problem becomes intricate.

Due to the above challenges encountered in subspace clustering approaches, there has been a scope of improvement in these algorithms. Subspace clustering not only determines the clusters in the dataset but also the subspaces in which these clusters are present. The next section presents the review of subspace clustering classification.

3 Classification of Subspace Clustering Approaches

In order to determine a group of similar data points in different subspaces, subspace clustering algorithms are employed. This section discusses the various categories of subspace clustering approaches on the basis of different parameters. Figure 3 depicts the categories of subspace clustering algorithms on the basis of parameters required in algorithms for forming clusters from data objects [8]. These are as follows:

Cell-based Subspace Clustering: This approach is also called a grid-based approach. It makes use of an approximate number of cells required to form a cluster.

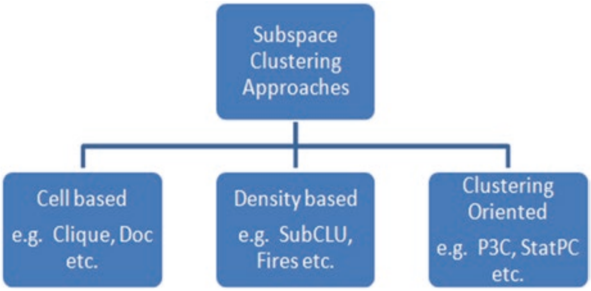

Fig. 3 Classification of subspace clustering approaches on the basis of cluster definitions

The cluster description starts with a minimum width "w" of a number of cells. Each cell contains a minimum threshold number of objects. The cells of a cluster are either of fixed grid size or variable in number forming hypercube of width "w." The cells participating in clustering uses subsets of dimensions of datasets. Hence the relevant dimensions to a particular cluster are determined. Irrelevant dimensions, not participating in clustering, expand on other cells. Few cell-based algorithms are CLIQUE, DOC, MINECLUS, SCHISM, etc.

Density-based Subspace Clustering: This approach is able to determine the clusters of arbitrary shapes. It is dependent upon the density of data objects lying in datasets. This approach separates the dense region from sparse region. Density is determined through distance measure. The parameters used in algorithms are the least number of points "minpts" required to form a cluster and "epsilon" distance among the neighboring points. The dense region is formed by counting the number of points "minpts" within "epsilon" neighboring distance. Any region not satisfying the "minpts" and "epsilon" properties is not able to form a cluster and is termed as sparse region. Some density-based subspace clustering algorithms are FIRES, INCY, SUBCLU, etc.

Clustering-Oriented-Based Subspace Clustering: This approach does not provide any requirements for cluster formation. It is not dependent on any cluster definition or on input parameters to form a cluster. As the name suggests it gives statistical orientation properties of total clusters formed. It means it defines properties of resultant clusters formed like a number of clusters formed, average dimensionality per cluster, etc. Clustering oriented approach is more suitable to datasets of varied distributions. Some algorithms of this approach are STATPC, P3C, PROCLUS, etc.

Subspace clustering approaches confine their search in such a manner that clusters existing in different subspaces are extracted [3]. The search either proceeds from single-dimensional to full dimensional dataset (bottom-up) or full dimensional to single dimensional dataset (top-down). Each search method is defined as follows:

Bottom-Up Subspace Search Method: It is a grid-based method which starts from single dimension. This method follows an a priori approach [1] to determine relevant dimensions. The method begins with forming similar groups in single dimensions on the basis of density threshold parameters and grid size. The data objects participating in single dimension will also participate in multi-dimensional grids. This approach detects the noise and also determines overlapping subspace clusters. However, it may find redundant subspaces or clusters across the dataset. Some algorithms of bottom-up approach are CLIQUE, ENCLUS, DOC, etc. (Fig. 4).

Top-Down Subspace Search Method: It is an iterative method which starts from entire dimensions of dataset. Initially, each dimension is assigned equal weights and clustering begins. After clustering, the weights of each dimension for each cluster are updated. In the next iteration, updated weights are used, clustering proceeds, and again weights are updated. Dimensions with highest weights for a cluster are relevant dimensions. This is expensive method which requires many iterations. Input parameters used in this approach are number of clusters and size of subspaces. Both the parameters are difficult to decide before clustering. This method finds non-overlapping clusters. Some algorithms are PROCLUS, FINDIT, MAFIA, etc.

Data objects of clusters determined from subspace clustering approaches may or may not be aligned along the axis [9, 10]. On this basis, subspace clustering approaches are divided into two categories as shown in Fig. 5.

Axis Aligned: Clusters determined along the parallel axis to data space are axis-aligned. These subspace clusters could be determined with low computational complexity. Number of subspaces determined from this approach are fixed in number, e.g., CLIQUE and DOC.

Non-axis Aligned: the subspace clusters determined in the arbitrary orientation of data space are non-axis aligned. Clusters may be expressed in a better way using

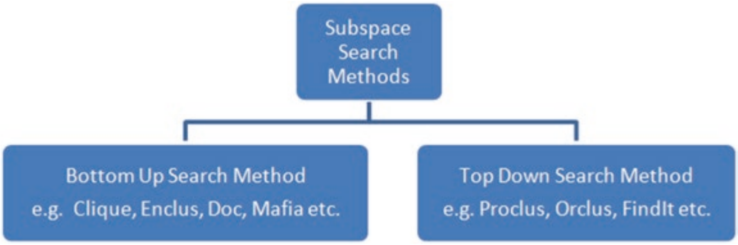
Subspace Search Methods
Bottom Up Search Method
e.g. Clique, Enclus, Doc, Mafia etc.
Top Down Search Method
e.g. Proclus, Orclus, FindIt etc.

Fig. 4 Classification of subspace clustering approaches on basis of search methods

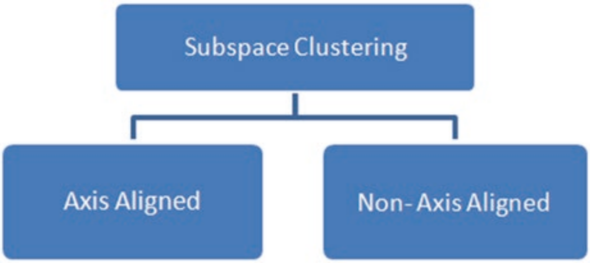

Fig. 5 Classification of subspace clustering approaches on basis of axis alignment

this approach. However, computation complexity is quite high in finding clusters in arbitrary orientation. Subspaces may go infinite in number, e.g., Orclus.

4 Evaluation Measures for Subspace Clustering Algorithms

This section describes the various systematic evaluation measures used for comparing objects and subspaces of clusters formed. However, there are no standard criteria defined for comparing the subspaces or clusters formed from subspace clustering approach. In literature, researchers have employed different evaluation measures for performance assessment of subspace clustering algorithms. A common ground of comparing subspace clustering algorithms is lacking. This is because true cluster labels along with relevant dimensions are lacking in datasets. The paper presents thorough evaluation measures shown in Fig. 6, required for comparing subspace clustering algorithms [8, 9].

Object-Based Validation Measures: This type of validation considers the data objects participating in the clustering process. The various measures are as follows:

F1_Measure – It is the harmonic mean of precision and recall values. This measure ensures that actual cluster (found cluster from the algorithm) should mask maximum objects of true cluster (already given class in dataset) and unmask the objects of different clusters. This is computed from the confusion matrix [1]. Let TP is true positive, i.e., objects of actual cluster are same of true cluster; TN is true negative, i.e., objects of different true clusters are mapped to different actual

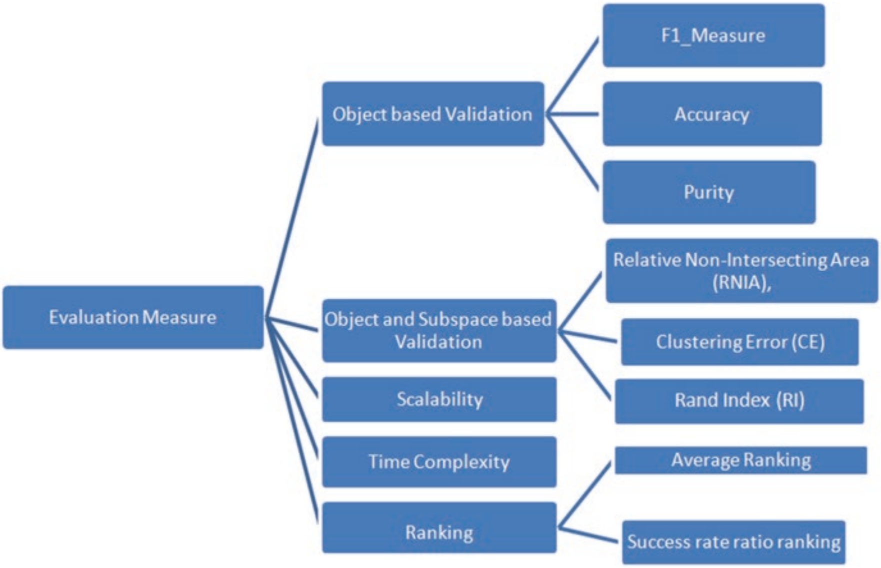

Fig. 6 Evaluation measures for subspace clustering approaches

clusters; FN false negative, i.e., objects of same true cluster belong to different actual cluster; FP false positive, i.e., objects of different true cluster belong to same actual cluster. Precision is minimum mapping of objects from other clusters, while recall is maximum mapping of objects from same true cluster. It is given by Eqs. 1, 2, and 3:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

$$\text{F1\_Measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

Accuracy: Accuracy is the ratio of a number of objects of the actual cluster correctly mapped with objects of true cluster by total objects. It can be calculated using confusion matrix through following Eq. 4:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4}$$

Purity: This measures the purity or homogeneity of actual clusters determined from clustering algorithm with respect to true clusters. It can be calculated as follows:

$$\text{Purity} = \frac{\text{TP}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{5}$$

Object and Subspace-Based Validation Measures: This type of validation considers not only the data objects but the relevant subsets of dimensions participating in the clustering process. These measures actually evaluate how well clusters are formed in various subspaces. In these measures, an object of a dataset is assumed to be divided into number of sub-objects which spans across all the dimensions of datasets. Hence a subspace cluster consists of relevant dimensions along with sub-objects. An object shared between the subspace clusters with disjoint dimensions will have different sub-objects. Thus, subspace clusters having same objects with different relevant dimensions do not overlap. The various measures are as follows:

Relative Non-Intersecting Area (RNIA): This measure ensures the maximum number of sub-objects found in actual subspace cluster maps the true subspace cluster. A confusion matrix is formed from actual subspace clusters versus true subspace clusters [9]. The matching sub-objects with each subspace clusters are filled in the matrix. Let “U” is the total number of sub-objects participating in true or actual subspace clusters. Same sub-objects participating found in both true and actual cluster (with the same dimension) is counted once. Let “I” is the intersecting subjects of true and actual subspace clusters, or it is the sum of all elements of confusion matrix. Then RNIA is calculated as follows:

$$\mathrm{RNIA} = \frac{\mathrm{U} - \mathrm{I}}{\mathrm{U}} \tag{6}$$

Clustering Error (CE): RNIA gives the same result in a case when several actual clusters match one true cluster or one actual cluster matches true cluster. Clustering error maps one actual cluster to almost one true cluster and vice versa. CE is also calculated from confusion matrix formed by above method. Let D is the sum of all principal diagonal elements, then CE is given as:

$$\mathrm{CE} = \frac{\mathrm{U} - \mathrm{D}}{\mathrm{U}} \tag{7}$$

Rand Index: This index is measured on the basis of counting the pair of subobjects that do or do not participate in clustering. All sub-objects that do not overlap are considered as single clusters. A confusion matrix is created considering all subspace clusters including singleton clusters. There are four labels important, i.e., N11, pair of sub-objects that are same in both actual and true subspace clusters; N10, pair of sub-objects that are same in a true cluster but not in actual cluster; N01, pair of sub-objects that are same in actual cluster but not in true cluster; and N00, pair of sub-objects that are different in both actual and true subspace clusters.

$$1 - \mathrm{Rand}_{\mathrm{index}} = \frac{\mathrm{N10} + \mathrm{N01}}{\mathrm{N}} \tag{8}$$

where $\mathrm{N} = \frac{\mathrm{U} * (\mathrm{U} - 1)}{2}$, U is the union of all sub-objects participating in clustering.

Scalability: Scalability is the measure which could be used to visualize and analyze the behavior of any algorithm. In subspace clustering algorithms, scalability is measured in terms of dimensionality. It depicts the performance of an algorithm with the increase in a number of dimensions of dataset. The graphs are plotted which could be shown for any evaluation measure with respect to dimensions. The X axis shows the dimensions and Y axis depicts evaluation metric like F1_Measure or accuracy etc. Scalability of subspace clustering algorithms is shown in the latter part of the chapter.

Time Complexity: Run time complexity measures the time taken by an algorithm to cluster a given dataset. It is represented in the form of graphs to compare subspace clustering algorithm with respect to dimensions. Few examples of representation are shown in [8].

Ranking: Subspace clustering algorithms can be ranked on the basis of average ranking (AR) and success rate ratio ranking (SRR) [11]. The average rank of an algorithm is computed by taking the mean of ranks on all dataset on basis of any evaluation measure. Let r_j^i be the j^{th} algorithm rank for i^{th} dataset. The average rank of each algorithm on total "*n*" datasets is computed using following Eq. 9:

$$r_j = \frac{\sum_{i=1}^{n} r_j^i}{n} \tag{9}$$

Success rate ratio rank (SRR) computes the ratio of success rates in a pair of algorithms. This method is useful in determining the significant differences in algorithms. In SRR ranking, an algorithm and dataset are taken and accuracy (any evaluation measure) ratio is calculated with respect to other algorithms. This ratio is computed by the following Equation:

$$\mathrm{SRR}_{j,k,j\neq k}^{i} = \frac{\mathrm{acc}_j^i}{\mathrm{acc}_k^i} \tag{10}$$

where "*i*" is the dataset, "*j*" is the algorithm for which SR is calculated, and "*k*" is the compared algorithm different from *j*. Hence SRR is calculated for algorithm "*j*" with reference to algorithm "*k*" on an *i*th dataset. Likewise, with the same pair of algorithms, SRR is calculated on all datasets. Subsequently, the mean of SRR is computed for all *n* dataset using Eq. 11.

$$\mathrm{SRR}_{j,k,j\neq k} = \frac{\sum_{i=1}^{n} \mathrm{SRR}_{j,k,j\neq k}^{i}}{n} \tag{11}$$

In the same way, the algorithm "*j*" is paired with all other algorithms on all datasets and overall SRR for algorithm "*j*" is given by:

$$\mathrm{SRR}_j = \frac{\sum_k \mathrm{SRR}_{j,k,j\neq k}}{m-1} \tag{12}$$

where "*m*" is a number of subspace algorithms used for comparison. Similarly, SRR for all algorithms against each and every algorithm is calculated and ranked.

The further sections discuss the literature survey outline with analysis of subspace clustering algorithms on a few evaluation metrics.

5 Literature

There is a number of good surveys made by researchers on high dimensional clustering but very few are on subspace clustering. Assent [2] made a brief survey of high dimensional clustering on different types of datasets. Authors have also shown different methods adopted for clustering high dimensional datasets. Clustering in high dimensions is proposed through various models by [6]. A survey on models of high dimensional clustering datasets is given by [12]. Kriegel et al. [5] presented a survey on clustering high dimension data through subspace, correlation, and pattern-based clustering methods. Steinbach et al. illustrated the challenges of high dimensional clustering in detail and proposed a concept-based model for handling dataset with large attributes [13]. Fahad et al. made a survey describing various classes of clustering algorithms along with different evaluation metrics [14]. A comparative analysis of different swarm intelligence-based clustering algorithm is depicted in [15]. A new clustering algorithm with modified flower pollination algorithm is shown in [16]. However, the various studies discussed above mainly focused on clustering high dimensional datasets; limited studies have been performed on subspace clustering approaches. Parson [3] made a comprehensive survey on subspace clustering approaches portraying different subspace search methods. Müller [8] depicted the different evaluation measures as well as different categories of subspace clustering algorithms. The work [17] also provided a WEKA platform for evaluating various subspace clustering algorithms. Subspace clustering through evolutionary technique is proposed by [18]. Table 1 presents the review on various subspace clustering algorithms along with evaluation metrics used, type of dataset applied, and maximum dimensions evaluated and drawbacks.

It can be observed from Table 1 that maximum dimension evaluated in high dimensional dataset is 5920, but 6144 is also there. Many algorithms have been proposed on subspace clustering to handle high dimensional datasets. A Monte Carlo-based subspace clustering algorithm [44] is proposed by Olson et al. and evaluated against subspace and projected clustering algorithms on real and synthetic datasets. The significance of subspace clustering approaches is increasing in this big data era. Figure 7 depicts the number of papers published in a span of 5 years (shown on the x-axis).

Subspace clustering approaches also gained importance in various repositories of publications. Figure 8 shows the percentage of subspace clustering papers deposited in different repositories.

Figure 9 gives the percentage of papers published in conference or journals. 53% of subspace clustering techniques was published in conferences and 47% of papers were published in journals.

The next section illustrates the empirical comparison of subspace clustering algorithms on real and artificial datasets.

6 Empirical Assessment

Subspace clustering algorithms are mainly classified as cell, density, and clustering-oriented-based algorithms [8]. The algorithms belonging to these categories cover almost all subspace clustering algorithms. Present section shows the empirical assessment of subspace algorithms on the basis of two evaluation measure i.e. F1_ Measure and accuracy. The algorithms are compared on the basis of ranking and scalability.

(i) Ranking of Subspace Clustering Algorithms: Rank of algorithms [11] on real and synthetic datasets are made independently on accuracy and F1_Measure.

Table 1 Survey of various subspace clustering algorithms

S. no.	Year	Authors and references	Approach used/ proposed	Evaluation measure	Type of dataset	Maximum dimensions evaluated	Gaps
1.	2019	Pan Ji et al. [19]	Adaptive low-rank kernel subspace clustering	Clustering error	Motion and face dataset	2016	Need to manually select kernel function
2.	2019	Jufeng Yang et al. [20]	Subspace clustering via good neighbors	Accuracy and NMI	Image real dataset	2016	Difficulty in finding good neighbors
3.	2018	Paul and Nayagam [21]	PROCLUS	Accuracy	Student performance studies	–	Authors could have used some latest algorithms to analyze the performance of students
4.	2018	Manolis C. Tsakiris Rene Vidal [22]	Sparse subspace clustering	–	–	–	The paper gives theoretical proves without implementation
5.	2018	Canyi Lu et al. [23]	k-block diagonal regularizer	Clustering error	Real dataset	4000	The proposed method is for nonconvex when makes it challenging for optimization
6.	2017	P.A. Traganitis and G. B. Giannakis [24]	Sketched subspace clustering	Accuracy, time	Real datasets (HOPKIN 155)	2016	Few more evaluation measures should be taken. Statistical comparison is missing
7.	2017	Chen et al. [25]	Projection subspace clustering	Clustering accuracy	Real datasets	5920	Selecting efficient values of the parameter in the algorithm needs to be focused
8.	2016	D. Kumar et al. [26]	clusiVAT algorithm	Accuracy, time	Synthetic and real datasets	500	Comparison of algorithms is made on accuracy and run time only Lacks statistical test on algorithms
9.	2015	A. Kaur and A. Dutta [27]	Subscale algorithm	F1_Measure	Synthetic and real	6144	The only F1_measure is considered for performance evaluation Statistical comparison not made Comparison is made with few subspace clustering algorithms

10.	2014	Goebl et al. [28]	Optimal subspace clustering	Normalized mutual information (NMI)	Synthetic dataset	16	Only NMI used for evaluation of the proposed algorithm Maximum dimensions in high dimensional dataset are quite low, i.e., 16
11.	2014	A. Kaur and A. Dutta [29]	Subscale algorithm	F1_Measure	Synthetic and real	500	Other evaluation measures not considered Only 1 real dataset of 500 dimensions is considered
12.	2014	L. Lin et al. [30]	GA-PSO	Error rate	Real dataset	13	Low dimension dataset is considered for evaluation The proposed algorithm is not compared with standard subspace clustering algorithms
13.	2014	D. Kothari et al. [31]	FCM extended version using random sampling	Run time	Real dataset	54	Method for handling unloadable data is missing Not compared with standard subspace algorithm Low dimension data considered for evaluation
14.	2014	Fahad et al. [14]	Survey	Runtime, stability, internal validity	Real datasets	149	Few subspace clustering algorithms are considered. highest dimension evaluated in 149 only
15.	2014	Vidal & Favaro [32]	Low-rank subspace algorithm	Clustering error	Real dataset (motion segmentation)	2016	Evaluation of algorithms is made on clustering error only

(continued)

Table 1 (continued)

S. no.	Year	Authors and references	Approach used/ proposed	Evaluation measure	Type of dataset	Maximum dimensions evaluated	Gaps
16.	2013	X. Chen et al. [33]	FG-k-means, a soft subspace clustering algorithm	F1_Measure, accuracy, precision, and recall	Synthetic and real datasets	500	Some real application is required for a testing algorithm Few more subspace algorithms should be considered for comparison
17.	2013	Cao et al. [34]	Weighting k-modes algorithm	Scalability, accuracy, and adjusted Rand index	Synthetic and real datasets	50	The proposed algorithm is evaluated on a maximum 50-dimensional data only Comparison is made with few algorithms only
18.	2013	Timmerman et al. [35]	Subspace K-means	Adjusted Rand index, cluster variance	Synthetic and real datasets	9	Lacks statistical comparison in algorithms Low dimension data is evaluated in the name of high dimensions
19.	2013	S. Jahirabadkar and P. Kulkarni [10]	Survey of subspace algorithms	None	None	None	Evaluation measure not defined Did not cover the complete list of subspace algorithms
20.	2013	Vijendra & Laxman [36]	Multi-objective subspace	Clustering error	Artificial and real datasets	250	Very few evaluation measures considered Performance assessment is made against two subspace algorithms only, i.e., Proclus and MOSCL

21.	2012	Gajawada & Toshniwa [37]	Projected particle swarm optimization clustering algorithm	Number of mismatched dimensions in clusters	Synthetic dataset	126	Paper lacks in presenting the results in terms of standard evaluation measures Not compared with any of the existing subspace clustering algorithms
22.	2012	Nourashrafeddin et al. [38]	EsubClus (evolutionary subspace clustering)	F1 score, running time	Synthetic dataset	500	Compared with the few existing subspace algorithms Statistical comparison of algorithms is missing
23.	2011	Lu et al. [39]	Weighted PSO	F SCORE, average run time	Synthetic and real datasets	2000	Comparison of algorithms is made on F SCORE and run time only Lacks statistical test on algorithms
24.	2011	Y. Zhao et al. [40]	Enhanced grid density-based approach for	Scalability, performance, and accuracy	Real datasets	100	Proposed algorithm could find subspaces for 100 dimension data only The statistical test is missing
25.	2010	Chu et al. [7]	Density conscious subspace clustering	Execution time, precision, and recall	Synthetic and real datasets	16	Highest dimension considered for evaluation is 16 only. Statistical comparison is missing
26.	2009	Sun & Xiong [41]	Genetic algorithm -based high dimensional clustering	Error rate	Real life datasets	10	Very few dimensions are considered for performance evaluation Evaluation metrics is error rate only
27.	2007	C. Bouveyron [6]	High dimensional data clustering model	BIC	Real and artificial datasets	100	Evaluated dimensions are up to 100 only

(continued)

Table 1 (continued)

S. no.	Year	Authors and references	Approach used/ proposed	Evaluation measure	Type of dataset	Maximum dimensions evaluated	Gaps
28.	2004	L. Parsons et al. [3]	Survey of subspace algorithms	Running vs. no. of instances/ dimensions	Synthetic dataset	100	Results of only two algorithms MAFIA and FINDIT were shown Evaluation measures were not discussed
29.	2004	L. Woo et al. [42]	FINDIT	F1_Measure, soundness	Synthetic datasets	50	The proposed algorithm was evaluated on different datasets but not compared with any of the clustering algorithms
30.	1998	P. S. Bradley, Fayyad, & C. Reina [43]	Scale K-means(SKM)	Log-likelihood, standard deviation	Synthetic and real datasets	100	Application of the proposed algorithm to the real-world problem is missing. Statistical analysis of the algorithm is lacking

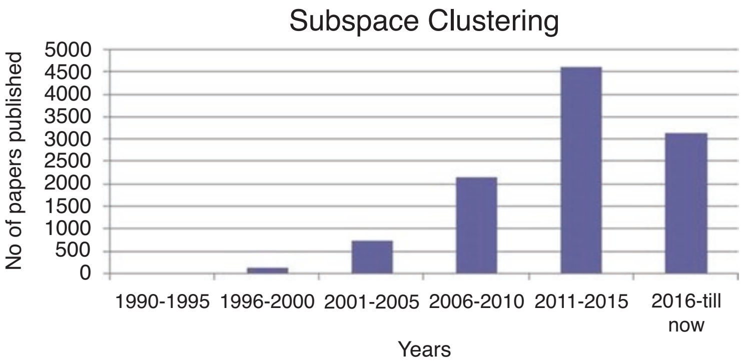

Fig. 7 Year-wise number of papers published on subspace clustering

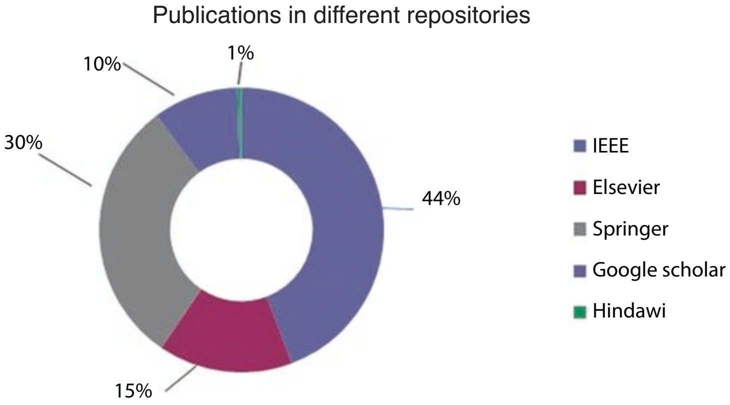

Fig. 8 Subspace clustering in different repositories

Real and synthetic datasets are obtained from [17]. Actual values of accuracy and F1_Measure of various subspace clustering algorithms on real datasets are extracted from [8]. While subscale algorithm is implemented in MATLAB R2013a platform adopting same parameter values given in [27]. Tables 2 and 3 shows the average rank and SRR rank of subspace clustering algorithms on real and synthetic datasets respectively. CLIQUE emerged to be on the first rank on accuracy, while MINECLUS best performs on F1_Measure.

Tables 4 and 5 present average and SRR ranks of subspace clustering algorithms on synthetic datasets in terms of F1_Measure and accuracy, respectively. Actual values of subspace algorithms on accuracy and F1_Measure on synthetic datasets were not available. Hence each algorithm is implemented on WEKA toolbox of subspace clustering provided by [17] with best parameter settings. It is observed that the subscale algorithm depicts better accuracy while DOC is best performer in terms of F1_Measure on synthetic datasets.

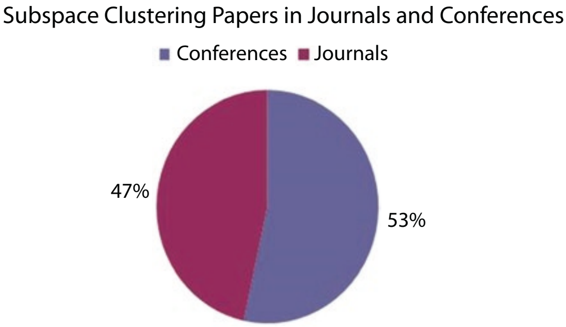

Fig. 9 Percentage of papers published in conferences and journals on subspace clustering

Table 2 Accuracy on real datasets

Accuracy		
	AR	SRR
SUBSCALE	9	11
CLIQUE	1	1
DOC	2	5
MINECLUS	4	7
SCHISM	3	2
SUBCLU	7	8
FIRES	11	10
INCY	4	3
PROCLUS	8	6
P3C	10	9
STATPC	6	4

Table 3 F1_Measure on real datasets

F1_Measure		
	AR	SRR
SUBSCALE	5	7
CLIQUE	9	8
DOC	2	3
MINECLUS	1	2
SCHISM	8	6
SUBCLU	7	9
FIRES	11	11
INCY	3	1
PROCLUS	6	4
P3C	10	10
STATPC	4	5

Table 4 Accuracy on synthetic datasets

Accuracy		
	AR	SRR
SUBSCALE	1	2
DOC	5	3
MINECLUS	6	4
SCHISM	3	7
FIRES	1	1
INCY	4	8
PROCLUS	8	6
STATPC	9	9
P3C	7	5

Table 5 F1_Measure on synthetic datasets

F1_Measure		
	AR	SRR
SUBSCALE	6	5
DOC	1	1
MINECLUS	4	2
SCHISM	7	7
FIRES	2	3
INCY	5	6
PROCLUS	3	4
STATPC	9	9
P3C	8	8

(ii) Scalability: Scalability of cell-based, density-based, and clustering-oriented-based subspace algorithms is shown in Figs. 9, 10, 11, 12, 13, and 14. Graphs are represented in terms of data dimensionality. Scalability depicts the performance of the algorithm with increasing number of dimensions. It is shown on synthetic datasets as number of records/objects are constant and dimensions vary from 10 to 75. Synthetic datasets are given in [17]. The X axis of graphs represents data dimensionality, and Y axis shows accuracy or F1_Measure.

Figures 10, 11, and 12 represent the performance of cell-based, density-based, and clustering-oriented-based algorithms, respectively, in terms of accuracy. It is observed from cell-based and density-based algorithms that SCHISM and INCY could not cope up after 25 dimensions and hence give results till D25 dataset only. MINECLUS accuracy varies randomly from dimensions to dimensions. DOC gives highest accuracy at 15 attribute dataset and then shows downfall. However, its accuracy shows slight improvement from D20 to D25 but again its performance declines. For density-based algorithms, FIRES shows improvement till D25, and then with

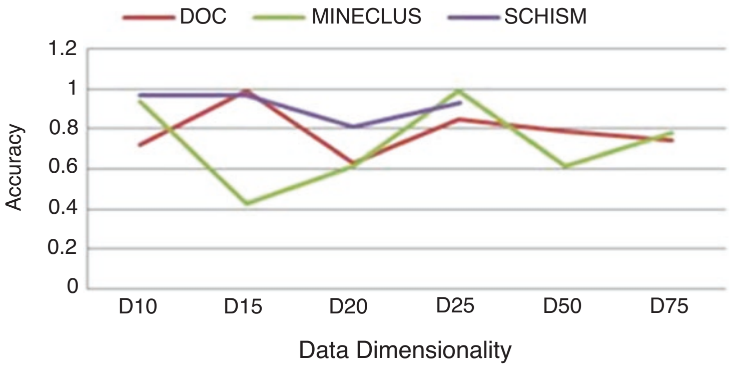

Fig. 10 Scalability of cell-based algorithms (accuracy)

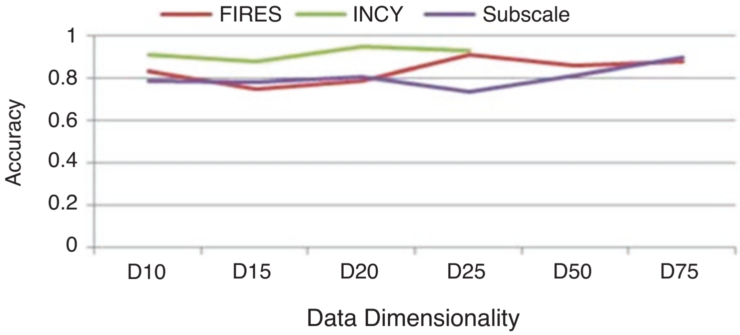

Fig. 11 Scalability of density-based algorithms (accuracy)

slight downfall, its performance becomes stagnant. SUBSCALE shows improvement in accuracy after D25 dimensional dataset. SUBSCALE and FIRES give highest and approximately same accuracy at 75-dimensional dataset. For clustering-oriented-based algorithms shown in Fig. 12, STATPC shows highest accuracy at D75 dataset. PROCLUS performance falls after 25-dimensional dataset. P3C shows low performance for overall datasets.

Figures 13, 14, and 15 depict the scalability of algorithms in terms of F1_Measure. It has been noticed that subscale gives highest F1_Measure on the 75-dimensional dataset. However, the efficacy of DOC, FIRES, and P3C shows downfall in performance with an increase in dimensions.

It can be concluded from given experiments that on synthetic datasets, FIRES and DOC depict the best performance on the basis of accuracy and F1_Measure, while CLIQUE and MINECLUS are best performers on accuracy and F1_Measure on real datasets. That means cell-based and density-based algorithm is more appropriate to opt for subspace clustering.

The next section illustrates the various application areas along with future prospects.

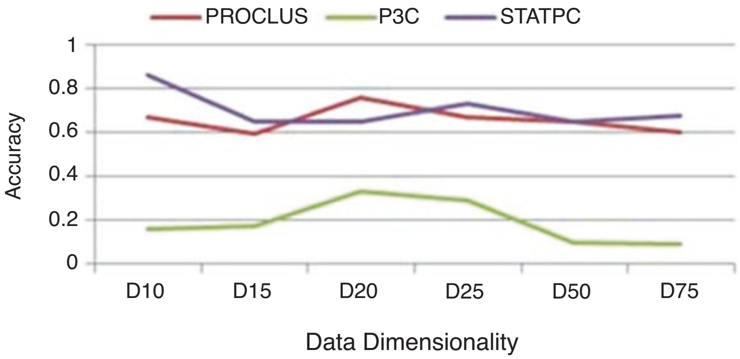

Fig. 12 Scalability of clustering-oriented-based algorithms (accuracy)

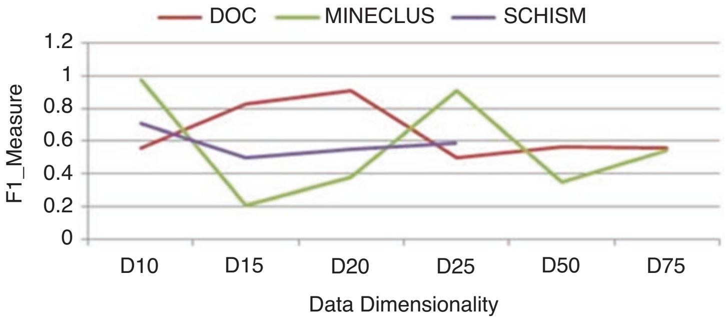

Fig. 13 Scalability of cell-based algorithms (F1_Measure)

7 Applications and Future Prospects

In previous sections, it has been shown that the trend of using subspace clustering algorithms for high dimensional problems is rising. This section discusses the application areas where subspace clustering algorithms are suitable. Additionally developing amalgamated subspace clustering algorithms are suggested. Following are the application areas of high dimensional clustering:

(i) Collaborative Filtering: The other name of collaborative filtering is a recommendation system. It is a social filtering technique where information is defined on basis of recommendations given by people [5]. People who like certain item in past are more likely to purchase in the future. People may like the items recommended by friends, neighbor, family, colleagues on social media, etc. Recommendations can be user-based or item-based. High dimensional clustering algorithms play an important role in such systems. The dataset is matrix of users and products. Clustering in such dataset retrieves the group of users liking the same product or group of items with relevant users. Hence subspace

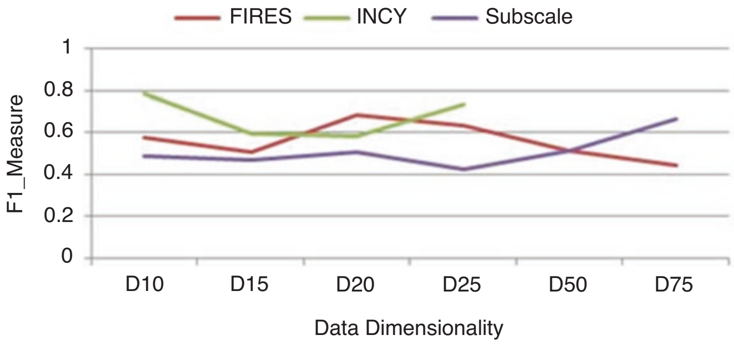

Fig. 14 Scalability of density-based algorithms (F1_Measure)

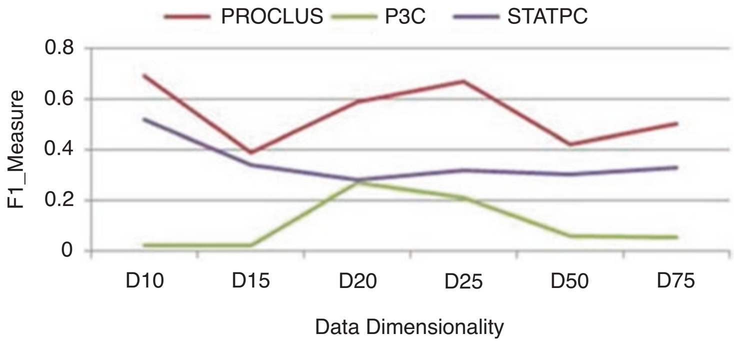

Fig. 15 Scalability of clustering-oriented-based algorithms (F1_Measure)

clustering could be applied [45, 46]. Some examples are Customer Recommendation System, Movie Recommendation System, etc.

(ii) Computer Vision: This field is based on mining useful information from single image or video to attain automatic visual understanding. The attributes extracted from image or video are large in number. An example of such dataset is image segmentation data [47]. The dataset contains 2310 data objects and 19 attributes. Clustering techniques are applied to cluster in either shape group or RGB group [33]. Similarly there a number of datasets with large attributes where clusters may exist in different subspaces [48]. Field of computer vision where subspace clustering can be applied is Facial Recognition, Gesture Recognition, etc.

(iii) Biological Dataset: Gene expression dataset is the most widely used dataset where subspace clustering can be applied [5]. Microarray DNA is a technology which measures a large amount of genes expressions under different circumstances. In order to understand the various types of diseases caused by genetic disorder at different levels, subspace clustering is required [49].

(iv) Text Documents: Clustering in text documents is an important task for web mining. Web pages are clustered on the basis of frequency of terms occurring in the page. Text documents are represented as high dimensional feature vector, where each feature is a frequency of term in document. Each document is represented by a data record/object. Hence the dataset formed from text document is high dimensional data [50]. The cluster of related document may exist on basis of similar word count, themes, etc. Hence subspace clustering techniques are applicable in such datasets.

(v) Distributed Databases: Massive amount of data is being dissipated by a number of sources like social media, microarray DNA, etc. Nowadays such a large amount of data is stored in different physical locations named distributed databases. The massive data is fragmented so that it can be distributed over multiple servers and parallel queries can be executed. Fragmentation can be row-wise (horizontal) or column-wise (vertical). Subspace clustering techniques can be applied to aid the fragmentation process. However, no work has been done yet on this application using subspace clustering approach.

(vi) Social Network: The data produced from social media is large in volume. Clustering can be performed on the social network to find influential groups in the network. This task is achieved by analyzing the linkages (edges) and nodes of the network. Clustering can be done on bases of various attributes associated with each node depending upon the objective in hand. Determining influential groups on the basis of various topics can be helpful in promotional activities, market segmentation problem, community detection problem [51], etc. Hence subspace clustering can be applied to find the groups in subsets of dimensions. However, a little amount of work is done on this application using subspace clustering approach.

In recent years, subspace clustering is being used with metaheuristic approaches. Nature-inspired algorithms [52, 53] are the prominent metaheuristic techniques that are used for determining the near-optimal solution to complex and hard problems. The first hybrid approach of subspace clustering with an evolutionary algorithm is proposed in [54]. With the advent of new metaheuristic algorithms, subspace clustering results could be improved [30, 36]. Some other algorithms can be developed amalgamating artificial bee colony algorithm, grey wolf algorithm, etc., with various subspace clustering algorithms. The hybrid algorithms developed were not applied to the applications described above. This will give a new direction of future work to researchers in the field of subspace clustering of high dimensional data.

8 Conclusion

Subspace clustering finds the clusters existing in various subsets of dimensions. The chapter presents a comprehensive survey on subspace clustering approaches, evaluation metrics, and application areas. The chapter also reveals the significance of subspace clustering in literature by presenting the statistical data. Comparison of conventional subspace clustering algorithms is also depicted through average ranking and success rate ratio ranking. Performance assessment of algorithms is made through scalability on basis data dimensionality. The chapter answers the following research questions:

(i) What are the major challenges faced by traditional clustering algorithms to cluster high dimensional data?
ANSWER: The problem of the curse of dimensionality is described in the first paragraph of Sect. 1.

(ii) What search techniques are being used in subspace clustering to determine subspaces?
ANSWER: Top-down and bottom-up search techniques used to find subspaces and are described in Sect. 3.

(iii) What are evaluation measures for comparing subspace clustering algorithms?
ANSWER: Different evaluation measures are described in Sect. 4.

(iv) What is the current scenario of subspace clustering?
ANSWER: Literature survey on subspace clustering with statistical data is illustrated in Sect. 5.

(v) What are the research gaps in the literature and the future prospects of subspace clustering?
ANSWER: Research gaps are given in Table 1, and future prospects are presented in Sect. 7.

Thus, the chapter is useful to the researchers planning to work in the field of subspace clustering. Additionally, it suggests the algorithms to develop in the future along with application areas.

References

1. J. Han, J. Pei, M. Kamber, *Data Mining: Concepts and Techniques* (Elsevier Inc, 2011)
2. I. Assent, Clustering high dimensional data. Wiley Interdiscip. Rev. Data Min. Knowl. Discov. **2**, 340–350 (2012). https://doi.org/10.1002/widm.1062
3. L. Parsons, E. Haque, H. Liu, Subspace clustering for high dimensional data: A review. ACM SIGKDD Explor. Newsl. **6**, 90–105 (2004). https://doi.org/10.1145/1007730.1007731
4. H.-P. Kriegel, P. Kroger, M. Renz, S. Wurst, A generic framework for efficient subspace clustering of high-dimensional data. ICDM, 250–257 (2005)
5. H.-P. Kriegel, P. Kröger, A. Zimek, Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. ACM Trans. Knowl. Discov. Data **3**, 1–58 (2009). https://doi.org/10.1145/1497577.1497578
6. C. Bouveyron, S. Girard, C. Schmid, High-dimensional data clustering. Comput. Stat. Data Anal. **52**, 502–519 (2007). https://doi.org/10.1016/j.csda.2007.02.009
7. Y.H. Chu, J.W. Huang, K.T. Chuang, D.N. Yang, M.S. Chen, Density conscious subspace clustering for high-dimensional data. IEEE Trans. Knowl. Data Eng. **22**, 16–30 (2010). https://doi.org/10.1109/TKDE.2008.224

8. E. Müller, S. Günnemann, I. Assent, T. Seidl, Evaluating clustering in subspace projections of high dimensional data, in *Proceedings of the VLDB Endowment*, (2009), pp. 1270–1281. https://doi.org/10.14778/1687627.1687770
9. A. Patrikainen, M. Meila, Comparing subspace clusterings. IEEE Trans. Knowl. Data Eng. **18**, 902–916 (2006). https://doi.org/10.1109/TKDE.2006.106
10. S. Jahirabadkar, P. Kulkarni, Scaf – An effective approach to classify subspace clustering algorithms. Int. J. Data Min. Knowl. Manag. Process. **3**, 69–86 (2013)
11. P. Brazdil, C. Soares, A comparison of ranking methods for classification algorithm selection. Mach. Learn. ECML 2000 **1810**, 63–75 (2000). https://doi.org/10.1007/3-540-45164-1_8
12. C. Bouveyron, C. Brunet, Model-based clustering of high-dimensional data: A review. Comput. Stat. Data Anal. Elsevier, 52–78 (2013)
13. M. Steinbach, E. Levent, V. Kumar, The challenges of clustering high dimensional data. New Vistas Stat. Phys. Appl. Econophys. Bioinf. Pattern Recognit., 273–309 (2004). https://doi.org/10.1007/978-3-662-08968-2_16
14. A. Fahad, N. Alshatri, Z. Tari, A. Alamri, I. Khalil, A.Y. Zomaya, S. Foufou, A. Bouras, A survey of clustering algorithms for big data: Taxonomy and empirical analysis. IEEE Trans. Emerg. Top. Comput. **2**, 267–279 (2014). https://doi.org/10.1109/TETC.2014.2330519
15. P. Agarwal, S. Mehta, Comparative analysis of nature inspired algorithms on data clustering. IEEE Int. Conf. Res. Comput. Intell. Commun. Netw., 119–124 (2015)
16. P. Agarwal, S. Mehta, Enhanced flower pollination algorithm on data clustering. Int. J. Comput. Appl. Taylor Fr. **7074**, 144–155 (2016). https://doi.org/10.1080/1206212X.2016.1224401
17. E. Müller, S. Günnemann, I. Assent, T. Seidl, I. Färber, Evaluating clustering in subspace projections of high dimensional data, http://dme.rwth-aachen.de/en/OpenSubspace/evaluation
18. I.A. Sarafis, P.W. Trinder, A.M.S. Zalzala, Towards effective subspace clustering with an evolutionary algorithm. 2003 Congr. Evol. Comput. CEC 2003 – Proc. **2**, 797–806 (2003). https://doi.org/10.1109/CEC.2003.1299749
19. P. Ji, I. Reid, R. Garg, H. Li, M. Salzmann, Adaptive low-rank kernel subspace clustering. arXiv:1707.04974 [cs.CV] (2017)
20. J. Yang, J. Liang, K. Wang, P. Rosin, M.-H. Yang, Subspace clustering via good neighbors. IEEE Trans. Pattern Anal. Mach. Intell., 1 (2019). https://doi.org/10.1109/tpami.2019.2913863
21. D.V. Paul, C.S. Nayagam, Use of subspace clustering algorithm for students' competency and subject knowledge assessment. Int. J. Knowl. Syst. Sci. **9**, 70–83 (2018). https://doi.org/10.4018/IJKSS.2018040104
22. M.C. Tsakiris, R. Vidal, Theoretical analysis of sparse subspace clustering with missing entries. 35th Int. Conf. Mach. Learn. ICML 2018 **11**, 7940–7949 (2018)
23. C. Lu, J. Feng, Z. Lin, T. Mei, S. Yan, Subspace clustering by block diagonal representation. IEEE Trans. Pattern Anal. Mach. Intell. **41**, 487–501 (2019). https://doi.org/10.1109/TPAMI.2018.2794348
24. P.A. Traganitis, G.B. Giannakis, Sketched subspace clustering. IEEE Trans. Signal Process. **2018**, 1–18 (2017). https://doi.org/10.1109/TSP.2017.2781649
25. X. Chen, M. Liao, X. Ye, Projection subspace clustering. J. Algorithms Comput. Technol. **11**, 224–233 (2017). https://doi.org/10.1177/1748301817707321
26. D. Kumar, J.C. Bezdek, M. Palaniswami, S. Rajasegarar, C. Leckie, T.C. Havens, A hybrid approach to clustering in big data. IEEE Trans. Cybern. **46**, 2372–2385 (2016). https://doi.org/10.1109/TCYB.2015.2477416
27. A. Kaur, A. Datta, A novel algorithm for fast and scalable subspace clustering of high-dimensional data. J. Big Data. **2**, 17 (2015). https://doi.org/10.1186/s40537-015-0027-y
28. S. Goebl, H. Xiao, C. Plant, C. Bohm, Finding the optimal subspace for clustering, in *IEEE International Conference on Data Mining (ICDM)*, (2014), pp. 130–139. https://doi.org/10.1109/ICDM.2014.34
29. A. Kaur, A. Datta, SUBSCALE: Fast and scalable subspace clustering for high dimensional data, in *IEEE International Conference on Data Mining Workshops (ICDM)*, (2014), pp. 621–628. https://doi.org/10.1109/ICDMW.2014.100

30. L. Lin, M. Gen, Y. Liang, A hybrid EA for high-dimensional subspace clustering problem. Proc. 2014 IEEE Congr. Evol. Comput. CEC 2014, 2855–2860 (2014). https://doi.org/10.1109/CEC.2014.6900313
31. D. Kothari, S.T. Narayanan, K.K. Devi, Extended fuzzy C-means with random sampling techniques for clustering large data. Int. J. Innov. Res. Adv. Eng. **1**, 1–4 (2014)
32. R. Vidal, P. Favaro, Low rank subspace clustering (LRSC). Pattern Recogn. Lett. **43**, 47–61 (2014). https://doi.org/10.1016/j.patrec.2013.08.006
33. X. Chen, Y. Ye, X. Xu, J.Z. Huang, A feature group weighting method for subspace clustering of high-dimensional data. Pattern Recogn. **45**, 434–446 (2012). https://doi.org/10.1016/j.patcog.2011.06.004
34. F. Cao, J. Liang, D. Li, X. Zhao, A weighting k-modes algorithm for subspace clustering of categorical data. Neurocomputing **108**, 23–30 (2013). https://doi.org/10.1016/j.neucom.2012.11.009
35. M.E. Timmerman, E. Ceulemans, K. De Roover, K. Van Leeuwen, Subspace K-means clustering. Behav. Res. Methods **45**, 1011–1023 (2013). https://doi.org/10.3758/s13428-013-0329-y
36. S. Vijendra, S. Laxman, Subspace clustering of high-dimensional data: An evolutionary approach. Appl. Comput. Intell. Soft Comput. **2013**, 1–13 (2013)
37. S. Gajawada, D. Toshniwal, Projected clustering using particle swarm optimization. Procedia Technol. **4**, 360–364 (2012). https://doi.org/10.1016/j.protcy.2012.05.055
38. S. Nourashrafeddin, D. Arnold, E. Milios, An evolutionary subspace clustering algorithm for high-dimensional data. Proc. Fourteenth Int. Conf. Genet. Evol. Comput. Conf. Companion, 1497–1498 (2012). https://doi.org/10.1145/2330784.2331011
39. Y. Lu, S. Wang, S. Li, C. Zhou, Particle swarm optimizer for variable weighting in clustering high-dimensional data. Mach. Learn. **82**, 43–70 (2011). https://doi.org/10.1007/s10994-009-5154-2
40. Y. Zhao, J. Cao, C. Zhang, S. Zhang, Enhancing grid-density based clustering for high dimensional data. J. Syst. Softw. **84**, 1524–1539 (2011). https://doi.org/10.1016/j.jss.2011.02.047
41. H. Sun, L. Xiong, Genetic algorithm-based high-dimensional data clustering technique, in *International Conference on Fuzzy Systems and Knowledge Discovery*, (2009), pp. 485–489. https://doi.org/10.1109/FSKD.2009.215
42. K.G. Woo, J.H. Lee, M.H. Kim, Y.J. Lee, FINDIT: A fast and intelligent subspace clustering algorithm using dimension voting. Inf. Softw. Technol. **46**, 255–271 (2004). https://doi.org/10.1016/j.infsof.2003.07.003
43. P.S. Bradley, U. Fayyad, C. Reina, Scaling clustering algorithms to large databases. KDD-98, 1–7 (1998)
44. C.F. Olson, D.C. Hunn, H.J. Lyons, Efficient Monte Carlo clustering in subspaces. Knowl. Inf. Syst. **52**, 751–772 (2017). https://doi.org/10.1007/s10115-017-1031-7
45. L. Boratto, S. Carta, Using collaborative filtering to overcome the curse of dimensionality when clustering users in a group recommender system, in *Proceedings of the 16th International Conference on Enterprise Information Systems*, (2014), pp. 564–572. https://doi.org/10.5220/0004865005640572
46. U. Kuzelewska, Clustering algorithms in hybrid recommender system on MovieLens data. Stud. Logic. Gramm. Rhetor. **37**, 125–139 (2014). https://doi.org/10.2478/slgr-2014-0021
47. K. Bache, M. Lichman, *UCI Machine Learning Repository* (University of California, School of Information and Computer Science, Irvine, 2006)
48. R. Vidal, Subspace Clustering. IEEE Signal Process. Mag. **28**, 52–68 (2011). https://doi.org/10.1109/MSP.2010.939739
49. D. Jiang, C. Tang, A. Zhang, Cluster analysis for gene expression data: A survey. IEEE Trans. Knowl. Data Eng. **16**, 1370–1386 (2004)
50. L. Jing, M.K. Ng, J. Xu, J.Z. Huang, Subspace clustering of text documents with feature weighting K -means algorithm. Lect. Notes Comput. Sci, 802–812 (2005). https://doi.org/10.1007/11430919_94

51. Z. Zhao, S. Feng, Q. Wang, J.Z. Huang, G.J. Williams, J. Fan, Topic oriented community detection through social objects and link analysis in social networks. Knowledge-Based Syst. **26**, 164–173 (2012). https://doi.org/10.1016/j.knosys.2011.07.017
52. P. Agarwal, S. Mehta, Nature-inspired algorithms: State-of-art, problems and prospects. Int. J. Comput. Appl. **100**, 14–21 (2014). https://doi.org/10.5120/17593-8331
53. P. Agarwal, S. Mehta, Empirical analysis of five nature-inspired algorithms on real parameter optimization problems. Artif. Intell. Rev., 1–57 (2017). https://doi.org/10.1007/s10462-017-9547-5
54. H. Road, S. Jose, Automatic subspace clustering mining of high dimensional applications for data. Proc. 1998 ACM SIGMOD Int. Conf. Manag. Data **27**, 94–105 (1998). https://doi.org/10.1145/276305.276314