ISSN:0975-9646

A Survey on Clustering Methods and Algorithms

Swati Harkanth, Prof. B. D. Phulpagar
Computer Science Department, PES Modern College of Engineering
Pune University, Pune, India

Abstract — **Clustering is a technique of data mining. It aims at finding natural partitioning of data. Objects are grouped depending on the similar nature they share. Similarity is measured depending on various parameters like number of parameters which are in common or lowest allowed difference between any parameters of an object. Data mining techniques include classification, clustering, mining frequent pattern, and correlation. Out of all these, in few decades clustering has gain wide attention of researchers. Clustering involves grouping of data objects which are similar in nature. This helps in the abstraction process of huge amount of data. Once the abstraction process is complete group of data can be represented in more compact manner. This is nothing but data compression. To describe the objects more precisely it needs to be defined by all the possible and meaningful dimensions. As the number of attributes increases finding similarity between objects become difficult. Thus discovering the distribution pattern of data becomes difficult. This has necessitate to look for meaningful grouping in subspaces i.e. subset of attributes.**

This paper presents an overview of Clustering-a tool in data mining; its application, its various methods of implementation and challenges to overcome.

Keywords— **Data mining, High Dimensional Data, Subspace clustering**

I. Introduction

Clustering is a technique in data mining which deals with huge amount of data. Clustering is intended to help a user in discovering and understanding the natural structure in a data set and abstract the meaning of large dataset. It is the task of partitioning objects of a data set into distinct groups such that two objects from one cluster are similar to each other, whereas two objects from distinct clusters are dissimilar [1]. Clustering is unsupervised learning in which we are not provided with classes, where we can place the data objects. Clustering is beneficial over classification because cost for labelling is reduced.

Clustering has applications in molecular biology, astronomy, geography, customer relation management, text mining, web mining, etc. All applications use clustering to derive useful patterns from the data which helps them in decision making This is helpful to draw certain conclusions and proceed further in that direction for enhancement of application.

Cluster Analysis is an important tool for exploratory data analysis which aims at summarizing main characteristics of data. Lot of work has been done in past in this field. Clustering algorithms such as K-mean algorithm has a history of fifty years [2]. Clustering methods have also been developed for categorical data. Clustering methods are applied in pattern recognition [3], image processing and information retrieval. Clustering has a rich history in other disciplines [4] such as biology, psychiatry, psychology, archaeology, geology, geography, and marketing [5]. Cluster finding methodologies differ from need to need of an application. Clustering methods and algorithms are dependent on number of instances to be considered, size of a single instance, accuracy of the result required. All these factors give rise to various methods and algorithms. This paper presents an overview of clustering techniques, their comparison, advantages and disadvantages of them, and challenges that needs attention. Section II describes clustering methods, section III of paper discusses how clustering differs when it is high dimensional data. Section IV focuses on subspace clustering. In section V we compare various methods and finally discuss applications and conclusion in the last two remaining sections.

II. Methods of Clustering

Two popular methods of clustering are hierarchical clustering and partitioning clustering [4].

A. Hierarchical Clustering

In this method data set is partitioned into distinct partitions. Again procedure of partitioning can be done in two ways-agglomerative and divisive. In agglomerative approach initially each object is treated as single cluster, later objects are merged depending on the criteria followed. In divisive approach initially entire data set is consider as a cluster. Entire cluster is then separated into smaller sets depending on their similarity. This method of hierarchical clustering generates a data structure called as dendogram. A typical dendogram is as shown in Fig. 1 [4]. Dendogram describes the order in which data objects are merged. Dendogram gives the hierarchy of clusters and hence the name. When we cut the dendogram at a point then we get the cluster at that level.

As shown in the diagram initially x_1 and x_2 these objects are grouped together. This grouping is performed only if x_1 and x_2 exhibit minimum level of similarity required by the application. At the same level object x_3 and x_4 are also grouped together. Again this grouping is done based on similarity the two objects share. Similarity measure can be any distance calculating measure such as Euclidian distance or Manhattan distance. Hierarchical clustering is again divided into two types as agglomerative and divisive.

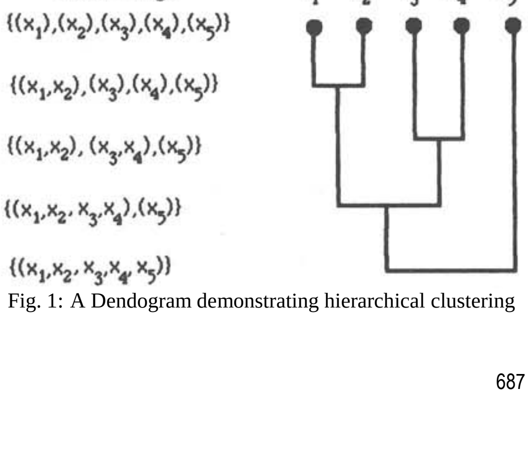
Clusterings
$\{(x_1),(x_2),(x_3),(x_4),(x_5)\}$
$\{(x_1,x_2),(x_3),(x_4),(x_5)\}$
$\{(x_1,x_2),(x_3,x_4),(x_5)\}$
$\{(x_1,x_2,x_3,x_4),(x_5)\}$
$\{(x_1,x_2,x_3,x_4,x_5)\}$
x_1
x_2
x_3
x_4
x_5

Fig. 1: A Dendogram demonstrating hierarchical clustering

1) *Agglomerative Approach:* This is bottom up approach of clustering the objects. Initially every object is a cluster in its own. It first contains *n*-clusters, where *n* is the number of objects in the data set. Algorithms in this category iterate to merge objects which are similar [6] and terminate when there is only one cluster left containing all *n*-data objects. In every stage, method groups the data objects which are most similar [2, 7].

   We need criteria to determine boundary of a cluster. An appropriate approach for this is to decide what should be the distance between two clusters [8]. Depending on the distance measure used again we have single linkage and complete linkage clustering. Both of the methods first calculates the pair wise distance between two objects, one is from first cluster and another object belongs to the second cluster. In single linkage method distance between two clusters is same as the minimum of all the pair wise distance calculated. In complete linkage method distance between two clusters is the maximum of all the pair wise distance calculated [5]. Thus single linkage method uses minimum distance criteria or nearest neighbour approach. Let *X* and *Y* are two sets of elements considered as clusters. Let *D*(*X*, *Y*) denotes distance between clusters *X* and *Y;* and *d(x, y)* is the distance between two elements *x* and *y*. For single linkage method distance is given as follows.

   $D(X, Y) = \min d(x, y)\ where\ x \in X\ and\ y\ \in Y$.

   For complete linkage method distance is given as,

   $D(X, Y) = \max d(x, y)\ where\ x \in X\ and\ y\ \in Y$.

2) *Divisive Approach:* Divisive approach works exactly in the reverse way of agglomerative approach. It begins with the single cluster containing entire objects as the member of that single cluster. It then splits on till there are *n*-clusters, each containing single object. Divisive approach splits the bigger set of objects into smaller one. Attributes of an object are used as criteria for splitting. Either a single attribute or group of attributes or all attributes simultaneously can be considered for splitting, giving rise to further two approaches monoethic and polyethic methods respectively [7]. A cluster with N objects gives $2^{N-1} - 1$ possible two subset divisions [1], which is very expensive in computation. Hence divisive approach is not common in practice.

B. *Partitioning Method*

In this method each object is placed in exactly one set. Membership of an object in two different sets is not allowed. Here user has to give an input the required number of clusters. This method gives another class of clustering algorithms called as non-overlapping clustering algorithms. Output of partitioning method depends on input given. Same algorithm with same data set can give different output with large difference between each of them depending on required number of clusters.

III. Clustering for High Dimensional Data

High dimensional data has attributes ranging from five to ten or more. Such type of attribute range is used in description of many objects .In customer behaviour analysis these attributes are the range of products the customer purchase. Here object is an individual customer and it is described by the product it purchases. Similar example is glass description data where various attributes are the component of glass like magnesium, aluminium, silicon, potassium, calcium etc. and the value for an attribute describes percentage of the component in one mg. Similarly data of a diabetic patient has attributes as number of times pregnant, plasma glucose concentration, diastolic blood pressure, triceps skin fold thickness, two-hour serum insulin, body mass index, diabetes pedigree function, age, etc [9]. Like this there are many real world scenarios where we need many attributes to describe the data. As the number of attributes increases it becomes difficult to analyse the data or to understand the pattern of data. Finding similarity within two objects become very difficult unless and until those two object are exactly same. For similarity basis we have to consider all the attributes. It may be the case that two objects are not similar in all attributes but they may exhibit similarity in some features. This subset of features is called subspace. Modern Clustering approaches aims to find clusters in subspaces.

A. *Problem with high dimensional data*

High dimensional data suffers from the problem called ‘curse of dimensionality’ [10]. As number of dimensions increase, distance between any two objects tends to be similar. Again number of objects in a unit bin decreases with increase in dimensions due to which dense regions which could be identified as clusters are lost in space. Consider the example; we have different types of glass as objects and attributes of those objects are components of glass. We consider here two dimensional data so we take only two components say magnesium and silicon. In tabular form this data is as shown in table I.

TABLE I
Sample Data Set

X	5	6	7	8	9	10	15	17	20
Y	1	2	3	15	16	17	18	19	20

We plot the graph of this data. On *x* axis we take magnesium component and on *y* axis we take silicon component.

$x = \{5, 6, 7, 8, 9, 10, 15, 17, 20\}$

$y = \{1, 2, 3, 15, 16, 17, 18, 19, 20\}$

A two dimensional plot for the sample data set is as shown in Fig. 2.

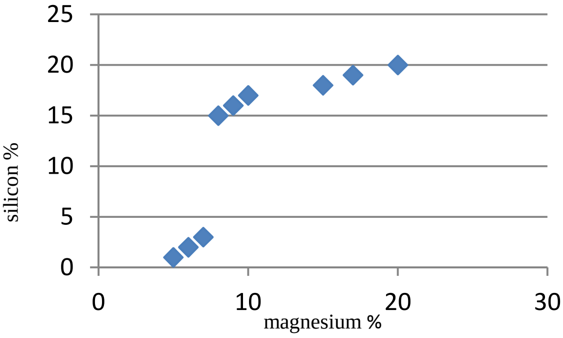

Fig. 2: A two dimensional plot of objects in space.

We consider the range of 0-15 as one unit bin. So in two dimensional plot of points, in one unit bin we get initial four points considering both axes x and y i.e. attributes magnesium and silicon. While considering only one dimension axis x, if we plot the point we get the distribution as shown in Fig 3 there are seven objects in one unit bin. If we consider both the dimensions, number of objects in one unit bin reduces to four. So as the number of dimension increases number of objects per unit bin decreases that means density decreases. When the attributes are more this distribution becomes more spares.

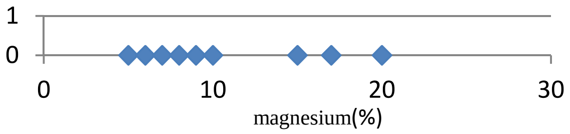

Fig. 3: Plot of objects considering single attribute.

Goal of Clustering technique is to find high quality clusters in reasonable amount of time. Again the result of clustering algorithm should be in a manner which is easy to interpret, analyse and draw further conclusions. Quality measure differs from the need of application to application.

There is no universal definition for the best cluster; it depends on the end user requirement. Initially to test the algorithm manual testing of result is necessary. Once the results are tested for the quality then we are assured for any data set algorithm will give some satisfactory result. Desired properties of cluster are:

- Homogeneity: Homogeneity means similar in nature. To measure similarity, distance measures are available. A set of objects, O can be said homogeneous if their pair wise distances are relatively small in the Euclidean space of the subspace; or if they are density-connected or if they exhibit common trends within the subspace.
- Significant size: Depending on the type of application user decides the size of cluster he wants. For some applications, where obtaining clusters is very easy, user wishes to have large size of cluster. This is advantageous because with this clusters obtained are few so it becomes easy to examine. However if the user wants to group data on very detailed, fine measure of similarity then the threshold for cluster size can be set large and the epsilon value should be set small so that the objects which are very similar only they are grouped.
- Maximal clusters: Pasquier et al. have proposed the concept of maximality for frequent item sets. Cluster in subspace C, D is maximal if there does not exist another subspace cluster. $C' = (O', A')$ such that$O' \subset O \wedge A' \subset A$. A cluster that is a subset of a maximal cluster conveys the same information as the maximal cluster. If we mine only maximal cluster, then there will be no redundant output.

Traditional Clustering algorithm tends to break with increase in number of parameters. Methods have been proposed to resolve this. Two popular methods are feature reduction or dimensionality reduction and subspace clustering.

B. Dimensionality Reduction:

This includes two techniques namely feature selection and feature transformation [4]. Feature transformation includes transforming original set of attributes into new one. Transformation techniques can be linear transformation, quadratic transformation, wavelet transformation, etc. Feature selection includes selecting the most relevant set of attributes and dropping the rest. For example instead of having three attributes for date of birth namely date, month, and year we can simply focus on year attribute and neglect the other two. However when there is no scope for omitting the attributes as all attributes contribute for providing useful information then using this approach is not helpful. In such cases researchers had provided the solution of looking clusters into subspaces.

IV. Subspace clustering

Clustering suffers from the problem of curse of dimensionality [10]. With increase in dimension data is lost in space. When it is not possible to omit or alter any of the attribute then dimensionality reduction approach is no more useful. To solve this problem subspace clustering has been proposed. Subspace clustering aims to find clusters which are hidden in subspaces and which cannot be identified if we consider all the attributes together. The very first algorithm to cope with large number of dimensions using subspace approach was CLIQUE (CLustering In QUEst) [11, 12]. On the similar lines was the ENCLUS (ENtropy based CLUStering) [11, 13] and MAFIA (Merging of Adaptive Finite IntervAls) [11, 14] algorithms. Subspace clustering algorithms can be classified into broad categories as grid based algorithms and density based algorithms.

A. Grid Based Algorithms:

In grid based approach dataset is partitioned into small grids say of width w. Objects which are under a particular grid structure are part of a cluster in that region. For each grid density is found out and from this, cells are sorted according to their densities and centre of clusters is found out. For this reason computational complexity of grid based algorithms is less. Such algorithms are easier to design as well as faster in computing results. Grid based approach on the other hand places a limit on the shape of cluster. Algorithms in this category are STING (STatistical INformation Grid-based method), OptiGrid, GRIDCLUS, GDILC [15] (Grid based Density Isoline Clustering algorithm). Computational complexity of most of these algorithms is linearly proportional to its data set[3].

1) *CLIQUE:* CLustering In QUEst [12] was an algorithm by R. Agrawal et al. proposed in 1998. It generates cluster descriptions in the form of DNF expressions. Algorithm is very much similar to apriori [16] algorithm for mining frequent item sets. It is grid based algorithm which partitions the data space into grid. For choosing the subspaces it uses coverage. Coverage is defined as fraction of dataset covered by dense units. Subspace having maximum coverage are kept rest are pruned. Results of CLIQUE are insensitive of the order of input data given. Algorithm is fast and scales well with size of data. However it does not scale well with number of dimensions in the output cluster.

B. Density Based Algorithm

Density based algorithm overcome the drawback of grid based algorithm. Algorithms in this category are DBSCAN [17] (Density Based algorithm for Discovering Clusters in

Large Spatial Databases with Noise), DENCLUE [13] (DENsity based CLUstEring). It can find cluster of all shapes and size. SUBCLU is an algorithm in this category.

1) *SUBCLU:* Karin Kailing, Hans-Peter Kriegel and Peer Kroger proposed SUBCLU (SUBspace CLUstering) [18] a density based algorithm. According to this clusters are defined as density connected sets. Algorithm runs in a bottom up manner to identify clusters. It initially runs DBSCAN [17] on each dimension to obtain one dimensional clusters. Then again DBSCAN [17] is run on all the relevant subspaces. SUBCLU identifies cluster of all shapes as it is density based. It identifies clusters as dense region separated from the sparse one. Concept of core object is applied in this algorithm. Inputs to this algorithm are minimum threshold and epsilon radius. Epsilon radius is the region in which we look for similar objects. Minimum threshold value specifies the minimum number of objects which when grouped together forms cluster. A core object is an object whose ε neighbourhood has minimum threshold number of objects, such core objects along with neighbouring objects forms the cluster.

V. Comparison of Clustering Algorithms

There is variety of algorithms available and we need to choose which suits best for our need. For hierarchical algorithms the time complexity grows increasingly as the number of instance increases. Partitioning algorithm need that user should have some knowledge about distribution of data because number of cluster has to be given as input. For grid based algorithms, they do not identify clusters of all shapes and sizes. Density based algorithm deals well with noise, can find clusters of all shapes. Table II shows the comparison of these algorithms.

VI. Applications

A. Bioinformatics

In this field gene and protein network and gene expression data are most common for clustering. Once the pre-processing of data is done gene expression data is expressed in the form of matrix where entry at i^{th} row and j^{th} column is the measured value of gene i under condition j. Gene clustering helps in knowing which characteristics are shared by different genes which are responsible for common function. A particular gene say *A,* may be grouped with gene *B* and not with gene *C* in one subset whereas *A* can be grouped with *C* and not with *B* in another subset [19]. Clustering algorithm takes into account this factor also

B. Customer Recommendation System

In this system objects are customer and attributes are products purchased by customers. This helps in understanding buying behaviour of customers. So it helps in knowing which product are purchased in which region, age group etc.

C. Compression of Data

Cluster analysis helps in data compression. Information which is present in the data set is abstracted in clusters. Now instead of mining entire data set we can mine clusters which are representative for a group of similar objects [8]. This is very effective where amount of data is huge.

D. Spatial Data Analysis

A huge amount of spatial data is obtained from satellite images, medical equipment, Geographical Information Systems (GIS), image database exploration etc. It is expensive and difficult for the users to examine spatial data in detail. Clustering helps to automate the process of analysing and understanding spatial data. It is used to identify and extract interesting characteristics and patterns that may exist in large spatial databases.

E. Web Mining

In this case, clustering is used to discover significant groups of documents on the web which are similar in nature. Classification of web documents assists in information discovery.

VII. CONCLUSION

The principal challenge for clustering high dimensional data is to overcome the “curse of dimensionality”. There are several recent approaches to clustering high dimensional data. These approaches have been successfully applied in many areas. We need to compare these different techniques and better understand their strengths and limitations. A particular method can be suitable for a particular distribution of data. We cannot expect that one type of clustering approach will be suitable for all types of data or even for all high dimensional data.

Many issues like scalability to large data sets, independence of order of input, validating clustering result are resolved to much extent. We need to focus on methods which can give us result in a manner which is easy to interpret. Result obtained should be in a manner which can also give us some conclusion and information about data distribution. It should further suggest us on how the clusters obtained can be helpful for various applications.

TABLE II Comparison of Clustering Algorithms

Type	Main characteristic	Time Complexity	Input Parameters	Algorithm
Hierarchical	Creates a hierarchical decomposition of the database represented as dendrogram.	Typically O(n^2)	Radius of cluster, branching factor	BIRCH[1] [20][21], CURE[2] [19]
Partitioning	Partitions the data set into n partitions where n needs to be defined initially by user	Typically O(n) [8]	Number of clusters required	K-mean, K-mode, PAM
Grid based	Define grid for data space and perform all operations w. r. t. grid	Typically O (n)	Size of grid, number of objects in a cell	STING
Density based	Find clusters as dense region separated by sparse regions	Typically O(n log n)	Threshold, radius	DBSCAN, DENCLU

[1] Balanced Iterative Reducing and Clustering using Hierarchies.

[2] Clustering Using REpresentatives.

ACKNOWLEDGMENT

I would like to thank my guide Prof. B. D. Phulpagar for his timely support. Without his guidance it would have not been possible to finish this work. I would also like to thank my co-coordinator, Prof. Deipali Gore and head of the computer department Prof S. A. Itkar whose motivation enabled me to present this work.

REFERENCES

[1] Rui Xu and W. Donald, "Survey of Clustering Algorithms," *IEEE Transaction on Neural Network,* vol. 16, 2005.

[2] J. MacQueen, "Some methods for classification and analysis of multivariate observations," *Proc. of the Fifth Berkeley Symp. On Math. Stat. and Prob.,* vol. 1, pp. 281-296, 1967.

[3] Gan Guojan, Ma Chaoqun, and W. Jianhong, *Data Clustering: Theory, Algorithm and Applications*. Philadelphia.

[4] A. Jain and R. Dubes, *Algorithms for Clustering Data*. New Jersey, 1948.

[5] A. K. Jain, M. N. Murtyand, and P. J. Flynn, "Data Clustering: A Review," *ACM Computing Surveys* vol. 31, pp. 264-324, 1999.

[6] P. Cimiano, A. Hotho, and S. Steffen, "Comparing Conceptual, Divisive and Agglomerative Clustering for Learning Taxonomies from Text," in *European Conference On Artificial Intelligence*, 2004.

[7] B. Everitt, S., S. Landau, M. Leese, and D. Stahl, *Cluster Analysis*. West Sussex, 2011.

[8] M. Halkidi, Y. Batistakis, and V. Michalis, "On Clustering Validation Techniques," *Journal of Intelligent Information Systems,* vol. 17, pp. 107-146, 2001.

[9] K. Bache and M. Lichman. (2013). *UCI Machine Learning Repository*. Available: http://archive.ics.uci.edu/ml/machine-learning-databases/

[10] M. Steinbach, L. Ertoz, and V. Kumar, " The Challenges of Clustering High Dimensional Data.," in *New Directions in Statistical Physics: Econophysics, Bioinformatics, and Pattern Recognition*, ed New Vistas: Springer, 2004.

[11] L. Parson, E. Haque, and H. Liu, "Subspace Clustering for High Dimensional Data: A Review " *Sigkdd Explorations,* vol. 14, pp. 90-106, 2004.

[12] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," in *Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, 1998, pp. 94-105.

[13] C. H. Cheng, A. W. Fu, and Y. Zhang, "Entropy-based subspace clustering for mining numerical data," in *International conference on Knowledge discovery and data mining*, 1999, pp. 84-93.

[14] S. Goil, H. Nagesh, and A. Choudhary, "Mafia: Efficient and scalable subspace clustering for very large data sets," Technical Report, Northwestern University, 1999.

[15] Y. Zhao and J. Song, "GDILC: a grid-based density-isoline clustering algorithm," in *International Conferences on Info-tech and Info-net Proceedings*, Beijing, 2001.

[16] R. Agrawal and S. Ramakrishnan, "Fast Algorithms for Mining Association Rules," in *International Conference on Very Large Data Bases*, 1994.

[17] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," in *International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 169-194.

[18] K. Kailing, H. Kriegel, and P. Kroger, "Density Connected Subspace Clustering for High - Dimensional Data " in *International Conference on Data Mining*, Lake Buena Vista, FL, 2004.

[19] G. Sudipto, R. Rajeev, and S. Kyuswok, "CURE: An Efficient Clustering Algorithm for Large Databases," in *International Conference on Management of data*, 1998, pp. 73-84.

[20] Z. Tian, R. Raghu, and L. Miron, "BIRCH: A New Data Clustering Algorithm and Its Applications," *Data Mining and Knowledge Discovery,* vol. 1, pp. 141-182, 1997.