A Method for Selecting Initial Centroids in Clustering Algorithms

Abstract. In the recent years, the Internet users have contributed huge amounts of data to businesses and industrial companies. These data can be used by business analysts extracting new and useful information. This information can help them in increasing the revenue, analyzing, and better understanding the customer behavior. Data mining in a nutshell is analyzing data from different aspects and bringing out useful and human-understandable information. In this paper, a clustering algorithm, which is based on a way of how to select the initial centroids of clusters, is introduced. The algorithm is tested on some of data sets from UCI machine learning repository. The results show that this algorithm is promising based on the quality of the clustering using Dunn index, Davies-Bouldin index and standard deviation.

Keywords: Data Mining, Unsupervised Clustering, Partitioning, K-Means, Centroid.

1 Introduction

Nowadays, businesses make a huge amount of data even on a daily basis. These data come from the customer's behavior, consumption habits, company's business flow, and so on. There is a need to extract useful information from this amount of data. One of the solutions to deal with huge amount of data is knowledge discovery and data mining [1]. In the recent years, knowledge discovery and data mining have been researched by lots of academic and industrial scholars.

Knowledge discovery is the extraction of potentially hidden information from structured data like xml and relational database files, or from semi-structured or unstructured data like web pages, images or videos [2]. Data mining is the analysis process or phase of knowledge discovery [3], which discovers the useful patterns or correlations from datasets. It is an interdisciplinary field in computer science [4-6] and involves algorithms and methods from artificial intelligence, machine learning, statistics, and database systems [4].

A branch of data mining is clustering, which is the task of grouping data points in a way that data points of the same group or cluster are similar, and data points of different groups are different or less similar to each other [7, 8]. For example, clustering the transaction records of customers of a company may show that customers, who are

married and have child(ren), buy milk and school snacks, or customers, who are students and male, buy peanut butter and coke. There are two types of clustering approaches in general [8]: partitioning and hierarchical. Partitioning algorithms group n data points to k clusters. Thepartitioning algorithms start with k initial random data points (called centroid of the cluster) and assign other data points to these k clusters based on the data points distance from the centroids. Hierarchical algorithms create a hierarchical decomposition of data points, which is represented by a tree. This tree is built in two ways. One way is top down (called divisive approach), in which all data points are at first in one cluster, and they split iteratively until each subset consists of only one data point. Another way is bottom-up, in which each subset consists of only one data point, and they merge iteratively until all data points become one cluster.

There are lots of free or commercial tools, which are used in data mining and clustering. Some of them are Weka [9], Matlab [10], R [11], ELKI [12], etc. There are very common tools, which can be used in data mining. A survey of available tools is presented in [13]. In this paper, Matlab and Weka applications are used to implement and test the algorithm. The rest of paper is as follows. Section 2 describes the proposed clustering algorithm. In section 3, the metrics to check the quality of any clustering algorithm are introduced concisely. The experimental results of the algorithm are shown in section 4. The proposed algorithm is executed on datasets from UCI Machine Learning Repository [14]. Also, the proposed algorithm is compared with other three popular and standard clustering algorithms in section 4. Section 5 concludes the paper.

2 The Proposed Clustering Algorithm

The proposed algorithm is a partitioning clustering algorithm. It consists of two steps. At the first step, the algorithm finds the centroids of the clusters, and at the second step it assigns the data points to each cluster based on their distance from the found centroids. The algorithm works as follows. Consider there is a dataset D with n data points and asked to group them into k clusters. The first step is to find k centroids. As a start, the mean of all the n data points is calculated. Then k data points are randomly chosen from the available data. The closest data point (from these k data points) to the mean is selected as the first centroid. Then again, k other data points are randomly chosen, and this time the farthest data point from the previously chosen centroid is selected as the second centroid. This task is repeated for the remaining k-2 centroids: k data points are randomly chosen and the farthest one to all other previously selected centroids is selected as the next centroid. For the distance function, Euclidean distance has been used, but any other distance measure can be used as well. The pseudo code of finding centroids step of the algorithm is as follows. The proposed Algorithm is implemented in MATLAB [10].

```
function centroids=find_centroids(dataset, num-
ber_of_clusters)
k = number_of_clusters;
```

```
m = mean(dataset);
candidate set = choose k data points randomly;
centroid 1 = min{distance(m, candidate set)};
add centroid 1 to the centroids set;
repeat k-1 times:
candidate set = choose k data points randomly;
next_centroid = max{distance(candidate set, centroid
set)};
add next_centroid to the centroid set
end
end of function
```

The above method is the first step of the proposed algorithm. The second step of the algorithm is as follows: for the remaining n-k data points (n minus k centroids), the Euclidean distance of each data point from all the centroids is calculated, and then the data point is assigned to a cluster, in which the data point has the minimum distance from its centroid. This step in repeated until all data points are assigned to the clusters. This algorithm will be compared to the popular K-Means [15], EM [16], and Farthest First [17] algorithms in section 4.

3 Cluster Validity Metrics

Clustering is an unsupervised process in data mining, and most of the clustering algorithms are very sensitive to the input parameters. Thus, it is important to evaluate the result of the clustering algorithms [18]. Compactness and separation are two measurement criteria proposed for evaluating the goodness or quality of clustering algorithms. Compactness says that members of a cluster should be as close to each other as possible. A common measure of compactness is variance [18]. Standard deviation is used in this work to measure the quality of compactness of the proposed algorithm and compare it with the other three popular algorithms. Standard deviation is a measure of how spread out the members are. The following shows the standard deviation formula:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}\,, \qquad where\ \mu = \frac{1}{N}\sum_{i=1}^{N} x_i\,, \quad (1)$$

where N is the number of members in a cluster and xi is ith member of the cluster.

Separation criterion implies that the clusters themselves should be widely separated. To measure and compare the separation quality of the proposed algorithm with other three algorithms Dunn Index [19] and David-Bouldin index [20] are used. Dunn index is defined as the ratio between the minimal inter-cluster distance to maximal intra-cluster distance. The following formula shows the Dunn index:

$$D = \min_{1 \le i \le n} \left\{ \min_{1 \le i \le n, i \ne j} \left\{ \frac{d(i,j)}{\max_{1 \le k \le n} d'(k)} \right\} \right\} \quad (2)$$

Where d(i,i) is the distance between clusters i and j, and d'(k) shows the intra-cluster distance of cluster k. inter-cluster distance d(i,i) can be any number of distance measure for example distance between centroids or means of the clusters. Similarly, intra-cluster distance can be measured in different ways. For a given set of clusters, a higher Dunn index indicates a better clustering.

The following formula shows the Davies-Bouldin index. For a given set of clusters, a lower DB index indicates a better clustering:

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{i \ne j} \left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)} \right), \quad (3)$$

where k is the number of clusters, cx is the centroid of the cluster x, σx is the average distance of all members of cluster x to its centroid cx, and d(ci,cj) is the distance between centroids ci and cj. In the following, the proposed algorithm is applied to some of the UCI machine learning data sets [14], and its results are compared with popular algorithms K-Means, EM and Farthest First using the three above mentioned cluster validity metrics.

4 Experimental Results

The proposed algorithm, which is described in section 2, is applied to some of the clustering datasets from UCI Machine Learning repository [14]. These data sets have been selected somewhat randomly from the clustering section of UCI machine learning repository. We have selected the data sets that their attribute types are numerical rather than text, which suits better the present work. The proposed algorithm is compared with Simple K-Means, EM and Farthest First algorithms, which have been executed on Weka software [9]. All the four algorithms have been tested on different number of clusters k=3,4,…,10, and their clustering quality has been measured by standard deviation, Dunn index and Davies-Bouldin index.

Table 1 shows the parameters, which have been set in Weka when applying the EM, Farthest First and K-Means algorithms on the data sets.

Table 1. Clustering parameters in Weka

Parameter	Clusters	MaxIterations	Seed	MinStdDev	ClusterMode	DisFunction
EM	3-10	100	100	1.0e-6	training set	-
FF	3-10	100	100	-	training set	-
KM	3-10	100	100	-	training set	Euclidean

In Table 1, FF, KM, MinStdDev and DisFunction represent Farthest First, K-Means, minimum standard deviation, and distance function respectively. The first

data set (from UCI repository), which is used, is Dow Jones Index Data Set [21]. It contains weekly data for the Dow Jones Industrial Index. Each record (row) of the data set is data for a week. There are 750 data records. 360 are from the first quarter of the year (Jan to Mar), and 390 are from the second quarter of the year (Apr to Jun) in 2011. For the current work, this data set has been cleaned, and non-numerical attributes (stock names and date), as well as records with missing values has been removed from the data set. The Fig. 1 shows the Davies-Bouldin index value of the algorithms on data set 1 (Dow Jones Index data set). In all the following figures, X-axis shows the number of clusters k, which is k=3,4,5,…,10, and Y-axis shows Davies-Bouldin index, and Dunn index respectively.

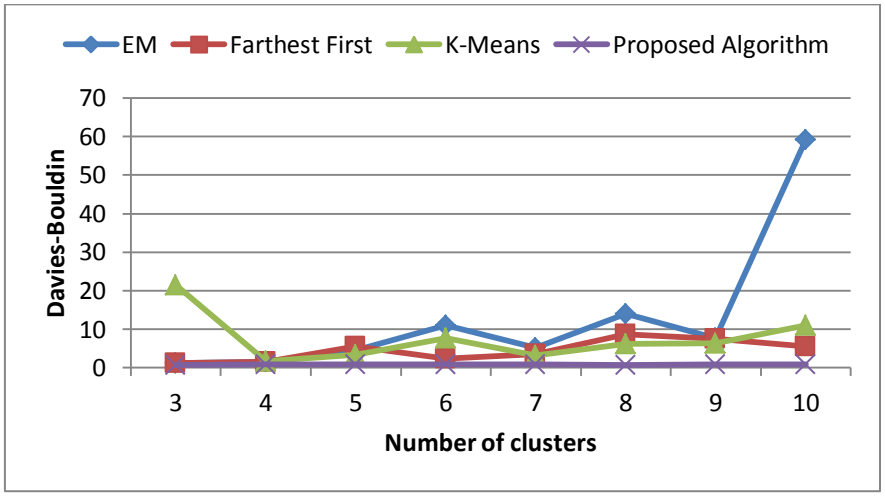

Fig. 1. Davies-Bouldin values of the algorithms on data set 1 (Dow Jones index data set).

For a given set of clusters, a clustering algorithm with a lower Davies–Bouldin (DB) index value has a better clustering. As the Fig.1 shows the proposed algorithm has a very low DB value, and better than the other algorithms. Fig. 2 shows the Dunn's index value of the algorithms on data set 1.

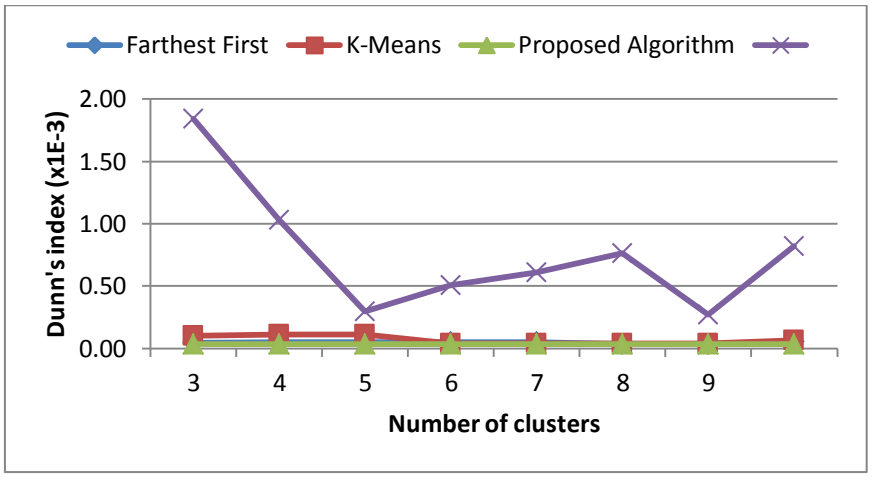

Fig. 2. Dunn's index values of the algorithms on data set 1 (Dow Jones index data set).

The Dunn index is used to identify dense, as well as well-separated clusters. Algorithms, which generate clusters with higher Dunn index, are more desired (have better clustering). As the Fig.2 shows, the proposed algorithm has a higher Dunn value and works better than the other algorithms.

The second data set, with which the algorithms have been tested on, is the Heart Disease Data Set [22]. This data set contains 4 different databases, in particular, the Cleveland database is the only one that has been used by the providers of this data. We also use this data set for the current work. The Cleveland data set contains 303 instances and 14 attributes. The attribute 'num' is used as the class attribute for classification purposes. Fig. 3 and Fig. 4 show the Davies-Bouldin index and Dunn's index values of the algorithms on this data set respectively.

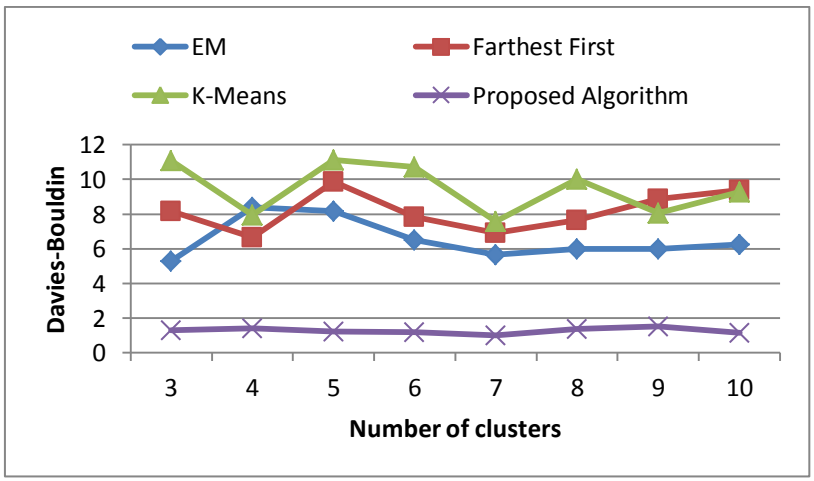

Fig. 3. Davies-Bouldin value of the algorithms on data set 2.

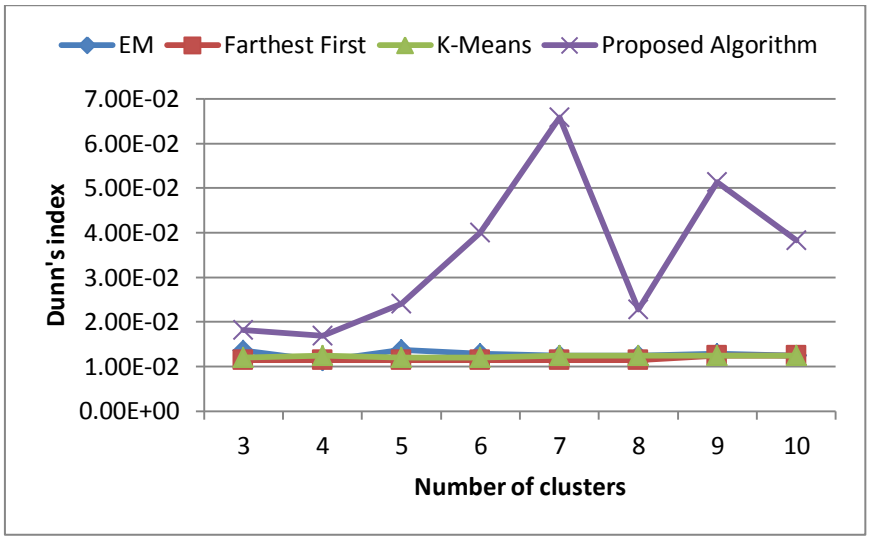

Fig. 4. Dunn's index values of the algorithms on data set 2.

As Fig. 3 and Fig 4 show, the proposed algorithm has better results on data set 2 as well. The third data set, with which the algorithms have been tested on, is the Wholesale customers Data Set [23]. The data set refers to clients of a wholesale distributor. It includes the annual spending in monetary units on diverse product categories [23]. The data set has 440 instances, and each instance has 8 attributes. Fig. 5 and Fig. 6 show the Davies-Bouldin index and Dunn's index values of the algorithms on this data set respectively.

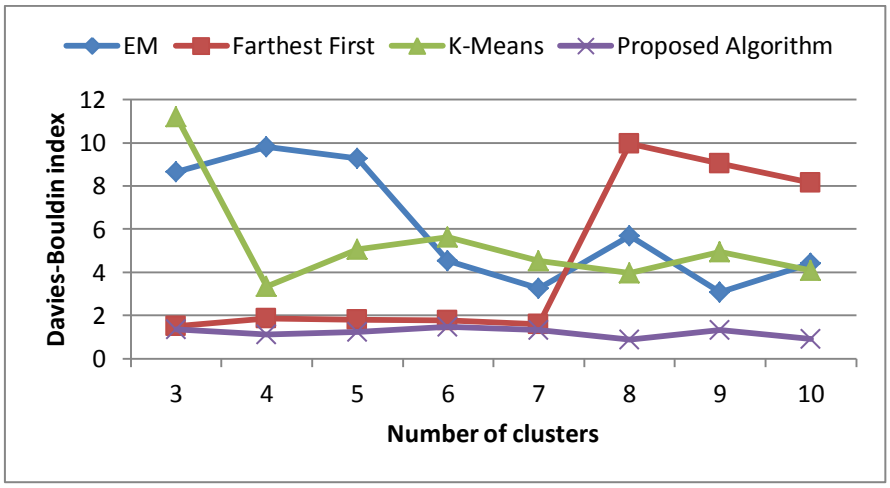

Fig. 5. Davies-Bouldin value of the algorithms on data set 3.

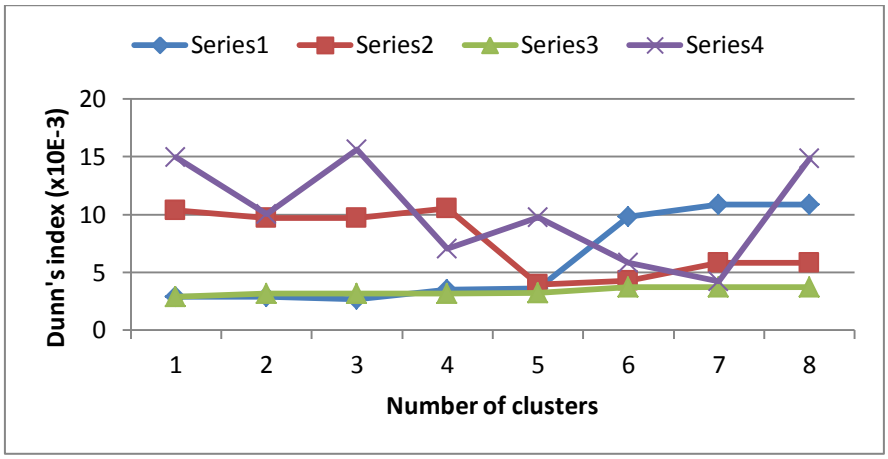

Fig. 6. Dunn's index values of the algorithms on data set 6.

As it can be seen from Fig. 5 and Fig 6, the proposed algorithm has better results in most cases. It should be mentioned that the proposed algorithm has been run only once in each case in compare to other algorithms with max iterations is 100. It means that for the more iterations of the proposed algorithm, better results may obtain.

The following tables 2, 3 and 4 show the standard deviation of the algorithms for data sets 1,2 and 3 respectively. Because of space saving, standard deviation only for the case, where the number of clusters is 10 (k=10), is shown.

Table 2. (split in two) Standard deviation of algorithms on data set 1 (when k=10)

Attributes	atr. 1	atr. 2	atr. 3	atr. 4	atr. 5	atr. 6	atr. 7
EM	0.401	3.896	3.862	3.725	3.745	62809441	2.547
Farthest First	0	11.164	11.278	10.855	10.99	1.33E+08	2.721
KM	0.050	7.664	7.711	7.437	7.603	78831953	2.466
Proposed Algorithm	0.448	13.669	13.950	13.439	13.748	51464276	2.559

atr. 8	atr. 9	atr. 10	atr. 11	atr. 12	atr. 13	atr. 14
40.164	69174911	3.741	3.741	2.652	37.864	0.143
33.965	95929497	10.978	10.819	2.564	28.349	0.137
41.401	82414822	7.606	7.581	2.684	26.646	0.139
35.791	55878068	13.727	13.855	2.411	29.696	0.170

Fig. 7 shows the standard deviation of the algorithms from Table 2 in a visual way. As it can be seen from this figure, the standard deviations of all of the algorithms are very close and similar to each other.

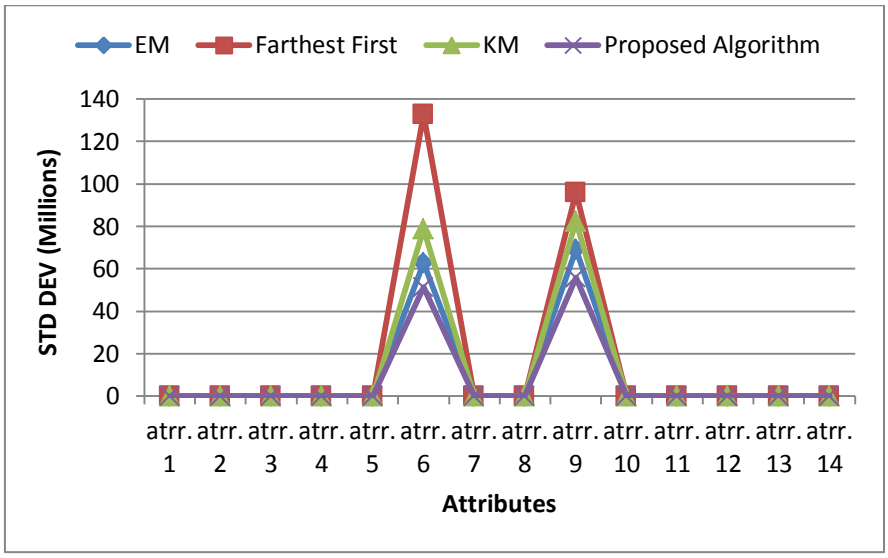

Fig. 7. Chart of the Standard Deviation from Table 2.

Table 3. (split in two) Standard deviation of algorithms on data set 2 (when k=10)

Attributes	atr. 1	atr. 2	atr. 3	atr. 4	atr. 5	atr. 6	atr. 7
EM	7.559	0.188	0.727	16.351	48.434	0.239	0.522
Farthest First	7.529	0.270	0.662	18.322	50.777	0.192	0.648
KM	7.818	0.275	0.817	16.102	47.87	0.139	0.263
Proposed Algorithm	6.502	0.376	0.743	12.670	19.929	0.281	0.769

Attributes	atr. 8	atr. 9	atr. 10	atr. 11	atr. 12	atr. 13
EM	18.286	0.270	1.004	0.557	0.840	0.496
Farthest First	20.329	0.155	1.128	0.545	0.720	1.188
KM	21.601	0.144	1.003	0.563	0.760	0.719
Proposed Algorithm	11.754	0.392	0.929	0.475	0.725	1.500

Table 4. (split in two) Standard deviation of algorithms on data set 3 (when k=10)

Attributes	atr. 1	atr. 2	atr. 3	atr. 4	atr. 5
EM	0.1520	0.400	12528.55	7253.486	6249.078
Farthest First	0	0.191	9787.113	7237.876	7946.795
KM	0	0.093	12274.07	6589.473	6514.499
Proposed Algorithm	0.112	0.574	6062.858	6173.384	5876.561

Attributes	attr. 6	atr. 7	atr. 8
EM	4764.377	2918.193	2943.887
Farthest First	2069.197	3568.528	1392.667
KM	5089.584	2812.141	3710.746
Proposed Algorithm	3971.824	2968.855	4347.3

In all the above tables, the atribute x on columns represent the attributes of data points. As tables 2, 3 and 4 show, the values of standard deviation for all the algorithms are close to each other. For space saving purpose, the standard deviation for other number of clusters (k=3...9) are not shown in the present paper. However, the results are similar and close to each other as for the case k=10. The introduced algorithm in this work has two steps in clustering data. In comparison to other clustering algorithms, the proposed has promising results in clustering data. As it was seen in experimental results, in most of the cases the proposed algorithm performs better than the other three algorithms.

5 Conclusion

Nowadays, businesses are creating huge amount of data from their transactions every day. They can bring out patterns and useful information from these data. One of common ways to find useful information is data mining. Clustering is a subfield of data mining which looks for patterns in data. In the present paper a partitioning clustering algorithm has been proposed and tested on different popular data sets. The proposed algorithm is a fast algorithm, and uses two steps to cluster data. The algorithm is compared to standard clustering algorithms such as K-Means, EM and Farthest First, and shows better clustering results. The quality of clustering is tested against Davies-Bouldin index, Dunn's index and standard deviation. The results show that the proposed algorithm in this study is a powerful method to cluster given data points.

References

1. Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., & Uthurusamy, R. 1996. Advances in knowledge discovery and data mining.
2. Piatetsky, Gregory, and William Frawley. 1991. Knowledge discovery in databases. MIT press.
3. Fayyad, Usama, Piatetsky-Shapiro, Gregory, Smyth, Padhraic. 1996. From Data Mining to Knowledge Discovery in Databases.
4. Chakrabarti, Soumen, Martin Ester, Usama Fayyad, Johannes Gehrke, Jiawei Han, Shinichi Morishita, Gregory Piatetsky-Shapiro, Wei Wang, and Intensive Working Group of ACM SIGKDD Curriculum Committee. 2006. Data Mining Curriculum: A Proposal (Version 1.0). ACM SIGKDD, April 2006.
5. Clifton, Christopher. 2010. Encyclopedia Britannica: Definition of Data Mining.
6. Hastie, Trevor, Tibshirani, Robert, Friedman, Jerome. 2009. The Elements of Statistical Learning: Data Mining, Inference, and Prediction.
7. Jain, A.K. and Dubes, R.C. 1988. Algorithms for Clustering Data. Prentice Hall.
8. Kaufman, L. and Rousseeuw, P. 1990. Finding Groups in Data: An Introduction to Cluster Analysis. John Wiley and Sons.
9. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. 2009. The WEKA data mining software: an update. ACM SIGKDD explorations newsletter, 11(1), 10-18.
10. MATLAB version 7.10.0. Natick, Massachusetts: The MathWorks Inc., 2010.
11. R Development Core Team. 2008. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, http://www.R-project.org.
12. Elke Achtert, Hans-Peter Kriegel, Erich Schubert, Arthur Zimek. 2013. Interactive Data Mining with 3D-Parallel-Coordinate-Trees. Proceedings of the ACM International Conference on Management of Data (SIGMOD), New York City, NY, 2013.
13. Mikut, Ralf, and Markus Reischl. 2011. Data mining tools. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 1.5 (2011): 431-443.
14. Asuncion, Arthur, and David Newman. 2007. UCI machine learning repository.
15. Hartigan, John A., and Manchek A. Wong. "Algorithm AS 136: A k-means clustering algorithm."Applied statistics(1979): 100-108.
16. Moon, Todd K. "The expectation-maximization algorithm." Signal processing magazine, IEEE 13.6 (1996): 47-60.
17. Hochbaum D.S., Shmoys D.B.: A best possible heuristic for the k center problem. Mathematics of Operations Research 1985, 10(2):180-184.
18. Kovács, Ferenc, Csaba Legány, and Attila Babos. "Cluster validity measurement techniques." 6th international symposium of Hungarian researchers on computational intelligence. 2005.
19. J.C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, J. Cybern.,vol. 3,pp. 32-57,1973.
20. D.L. Davies and D.W. Bouldin, A cluster separation measure, IEEE Trans. Pattern Anal. Machine Intell. Vol. 1, pp. 224-227, 1979.
21. Brown, M. S., Pelosi, M. & Dirska, H. (2013). Dynamic-radius Species-conserving Genetic Algorithm for the Financial Forecasting of Dow Jones Index Stocks. Machine Learning and Data Mining in Pattern Recognition, 7988, 27-41.
22. C. L. Blake and C. J. Merz. Repository of machine learning databases, University of California, Irvine, http://www.ics.uci.edu/~mlearn/mlrepository.html, 1998.

23. Abreu, N. (2011). Analise do perfil do cliente Recheio e desenvolvimento de um sistema promocional. Mestrado em Marketing, ISCTE-IUL, Lisbon.