JCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-1, Jan 2019 E-ISSN: 2347-2693

Analysis of K*(STAR) and Fuzzy C-Means Algorithm for Education Completion Performance

S.N.Ali Ansari[1*], Srinivasa Rao V[2]

[1]Rayalaseema university, Kurnool, India
[2]Dept of CSE, V.R.Siddhartha Engineering College, Vijayawada, India

Corresponding Author: ansarisn@gmail.com, Tel.: +91-7382147868

Available online at: www.ijcseonline.org

Accepted: 21/Jan/2019, Published: 31/Jan/2019

Abstract— in education domain, We can mine the hidden knowledge in the available databases for generating various analytical reports for proper decision making [10]. Grade Point Average (GPA) is commonly used as an indicator of academic performance [11]. An academic performance evaluation is a basic way to evaluate the progression of student performance, when evaluating student's academic performance, there are occasion where the student data is grouped especially when the amounts of data is large. Thus, the pattern of data relationship within and among groups can be revealed. Grouping data can be done by using clustering methods such as K-Means, K*(STAR) and the Fuzzy C-Means algorithms.

Classifying students using conventional techniques cannot give the desired level of accuracy, while doing it with the use of computing techniques may prove to be beneficial. Clustering or grouping a set of data sets is a key procedure for data processing .It is an unsupervised technique that is used to arrange pattern data into clusters. This research work deals with two of the most representative clustering algorithms namely centroid and crisp values based Fuzzy C-Means, K*(STAR) and represent object based on calculation of membership function. Fuzzy C-Means are described and analyzed for a datasets. Based on experimental results the algorithms are compared regarding their clustering quality and their performance, which depends on the time complexity between the various numbers of clusters chosen by the end user. The total elapsed time to cluster all the datasets and Clustering time for each cluster are also calculated and the results compared with one another [7].

Keywords—K*(STAR) Algorithm, Fuzzy C-Means Algorithm, cluster Analysis, fuzzy logic

I.INTRODUCTION

K*(STAR) clustering algorithm is a clustering algorithm that partitions a given dataset into C (or K) clusters dynamically. It needs a parameter C or K representing the number of clusters which should be known or determined as a fixed a priory value before going to cluster analysis or the number of clusters which can be calculated dynamically based on the data sets to cluster analysis. K*(STAR) is reported fast, robust and simple to implement. It gives comparatively good results if clusters in datasets are distinct or well separated. It was also examined that K*(STAR) is relatively efficient in computational time complexity with its cost of O (tcnp) in Lloyd algorithm (where t: number of iterations, c: number of clusters, n: number of objects, p: number of dimensions or number of features). a Certain dataset representations with Cartesian coordinates and polar coordinates may give different clustering results.

Bezdek (1981) introduced Fuzzy C-Means (FCM) [1] which is based on Dunn's study (Dunn 1973) as an extension of K-means algorithm [3]. As reviewed by M.-S. YANG (1993) and a dozen of the algorithms [2] have been developed in order to improve the efficiency and accuracy of FCM [4]. The basic FCM algorithm has frequently been used in a wide area of applications from engineering to economics. FCM is a soft algorithm clustering fuzzy data in which an object is not only a member of a cluster but member of many clusters in varying degree of membership as well. In this way, data sets located on boundaries of clusters are not forced to fully belong to a certain cluster, but rather they can be member of many clusters with a partial membership degree between 0 and 1. In spite of its higher cost with O (tc^2np) as when compared to K*(STAR), FCM has also been used in many clustering applications [5] because of its above mentioned advantages in education and economics area.

According to the research findings, it does not have a constant superiority in all cases of data structures. This paper studies generally have focused on comparison of K*(STAR) and FCM by using some education datasets. Thus, it would be helpful to examine these hard-and-soft C-Means partitioning algorithms for the data structures following different patterns and shapes of clusters. For that reason, in this paper we compared the efficiency of K*(STAR) and FCM algorithms on synthetically generated datasets consisting of different shaped clusters scattering with regular and non-regular patterns in two dimensional space.

II. K*(STAR) AND FUZZY C-MEANS ALGORITHMS

K*(STAR) algorithm iteratively computes cluster centroids for each distance measure in order to minimize the sum with respect to the specified measure in a systematic manner. Clusters are described by their data sets and by their centres. Usually centroids are used as the centres of clusters. The centroid of each cluster is the point to which the sum of distances from all objects in that cluster is minimized. By using a partitioning clustering algorithm, the data set ***X*** is partitioned into ***c*** clusters with a goal of obtaining low within-cluster and high between-cluster heterogeneity. That is, a cluster consists of objects which are as close to each other as possible, and as far from objects in other clusters as possible. Depending on research domains, dataset ***X*** is formed with large data points [12] that are the representations of objects which can be individuals, observations, cases, requirements, pixels etc.

While hard clustering algorithms like K*(STAR) assign each data item to exactly one cluster, soft partitioning or fuzzy clustering algorithms like FCM assign each data item to different clusters with varying degrees of membership as mentioned above. In other words, while the membership to a cluster is exactly either 0 or 1 in K*(STAR), it varies between 0 and 1 in FCM [9]. Therefore, in the cases that we cannot easily decide that objects belongs to only one cluster, especially with the datasets having noises or outliers, FCM may be better than K*(STAR). For that reason, it is expected that K*(STAR) algorithm may be a good option for exclusive clustering but FCM may give good results for overlapping clusters. In the following subsections K*(STAR) and FCM is explained with their algorithmic steps.

2.1. K*(STAR) algorithm

K*(STAR) algorithm iteratively computes cluster centroids for each distance measure in order to minimize the sum with respect to the specified measure in a systematic manner. The K*(STAR) clustering algorithm is as follows:

Input:

Input: I = n_1, n2, n3, ---n // Set of n number of data points
Output: A set of R (Systematic Regions) Clusters. // Number of desired Clusters.

Method:

Step 1: Calculate the required number of clusters based on dataset and a set of predicates dynamically and Apply standard K-means algorithm, or to calculate the required number of clusters by creating clusters dynamically, Apply the below steps.
Step 2: Calculate R = sqrt (n/2).
Step 3: Allocate Clno = 1 where Clno indicates cluster number.
Step 4: Find the closest pair of data point from I. Move those points to new set N_{Clno}.
Step 5: Find the closest point of N_{Clno} and move it to N_{Clno} from input I.
Step 6: Repeat step 5 until the number of elements of NClno reach (n/R)*Clno.
Step 7: When, N_{Clno} are full and the number of elements of Input I! = 0. Increment the value of Clno. New Clno = Clno + 1. Repeat step 4.
Step 8: The center of gravity of each N_{Clno}. Those are the initial centroids $C_{\mathbf{j}}$ and elements of N_{Clno} are the elements of R_m. Step 9.k= k+1 go to Step 1.
Step 9: Find the closest centroid for each data Points in a systematic way of identifying boundary curve and allocate each data points Cluster. Calculate new centroid for each R_m. Every time, the Asynchronous regions and corresponding data points are aligned into respective clusters.
Step 10: Calculate the largest distance D_L (largest distanced data point from each centroid of each cluster).
Step 11: Get the data points in the interval of (D_L * 4/9) to D_L .
Step 12: Find the closest centroid for those points and allocate them to closest centroid cluster.
Step 13: Find new centroid for each R_m.If, any change in any R_m.Repeat step 10. Otherwise go to end of the algorithm Step 14.
Step 14: Stop

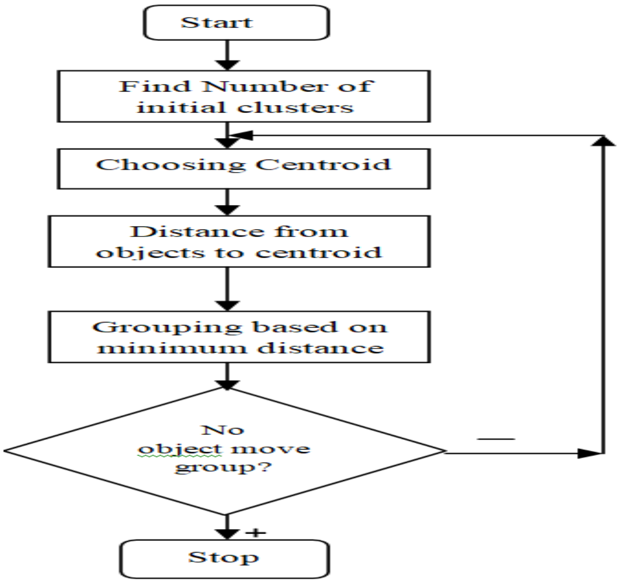

Start
Find Number of initial clusters
Choosing Centroid
Distance from objects to centroid
Grouping based on minimum distance
No object move group?
—
+
Stop

Figure-1: K*(STAR) Clustering Algorithm

2.2. Fuzzy C-means algorithm

This algorithm works by assigning membership to each data point corresponding to each cluster centre on the basis of distance between the cluster centre and the data point. More the data is near to the cluster centre more is its membership towards the particular cluster centre. Clearly, summation of membership of each data point should be equal to one. After each iteration, membership and cluster centres are updated according to the formula:

$$\mu_{ij} = 1 / \sum_{k=1}^{c} (d_{ij} / d_{ik})^{(2/m-1)}$$

$$v_j = (\sum_{i=1}^{n} (\mu_{ij})^m x_i) / (\sum_{i=1}^{n} (\mu_{ij})^m), \forall j = 1, 2, ...c$$

where, *'n'* is the number of data points 'V*j'* represents the *j*th cluster centre *'m'* is the fuzziness index m € [1, ∞]. *'c'* represents the number of cluster centre. *'µij'* represents the membership of *i*th data to *j*th cluster centre. *'dij'* represents the Euclidean distance between *i*th data and *j*th cluster centre.

Main objective of fuzzy c-means algorithm is to minimize:

$$J(U,V) = \sum_{i=1}^{n} \sum_{j=1}^{c} (\mu_{ij})^m \left\| x_i - v_j \right\|^2$$

Where, $'||x_i - v_j||'$ is the Euclidean distance between *i*th data and *j*th cluster centre.

Algorithmic steps for Fuzzy C-Means clustering:

Let X = $\{x_1, x_2, x_3 ..., x_n\}$ be the set of data points and V = $\{v_1, v_2, v_3 ..., v_c\}$ be the set of centres.

1) Randomly select *'c'* cluster centres.

2) calculate the fuzzy membership $'\mu_{ij}'$ using:

$$\mu_{ij} = 1 / \sum_{k=1}^{c} (d_{ij} / d_{ik})^{(2/m-1)}$$

3) Compute the fuzzy centers $'v_j'$ using:

$$v_j = (\sum_{i=1}^{n} (\mu_{ij})^m x_i) / (\sum_{i=1}^{n} (\mu_{ij})^m), \forall j = 1, 2, ...c$$

4) Repeat step 2) and 3) until the minimum *'J'* value is achieved or $||U^{(k+1)} - U^{(k)}|| < \beta$.

where, *'k'* is the iteration step. *'β'* is the termination criterion between [0, 1]. *'U =* $(\mu_{ij})_{n*c}$*'* is the fuzzy membership matrix. *'J'* is the objective function.

The FCM Algorithm should specify the following parameters the number of clusters, c, the 'fuzziness' exponent, m, the termination tolerance, ϵ, and the norm-inducing matrix, A. The fuzzy partition matrix U must be initialized. The determination of the no. of clusters c is more important as it has more influence on the partitioning as compared with the other parameters. When clustering real data without any a priori information about the structures in the data,

Fuzzy C-Means allows data points to be assigned into more than one cluster each data point has a degree of membership (or probability) of belonging to each cluster. Fuzzy C-Means has been a very important tool for image processing in clustering objects in an image [12].

The conventional clustering algorithms are the partitioning algorithms where each data object belongs to only single cluster. So, the clusters in k-means are said to be disjointed. Fuzzy clustering (Hoppner, 2005) extends this notion and suggests a soft clustering schema [5]. Here, the pattern is represented by the membership function given to each cluster. The assignment of the pattern to the cluster larger membership values gives better performance. In a fuzzy clustering when the threshold of this membership values are obtained a hard clustering can be retrieved.

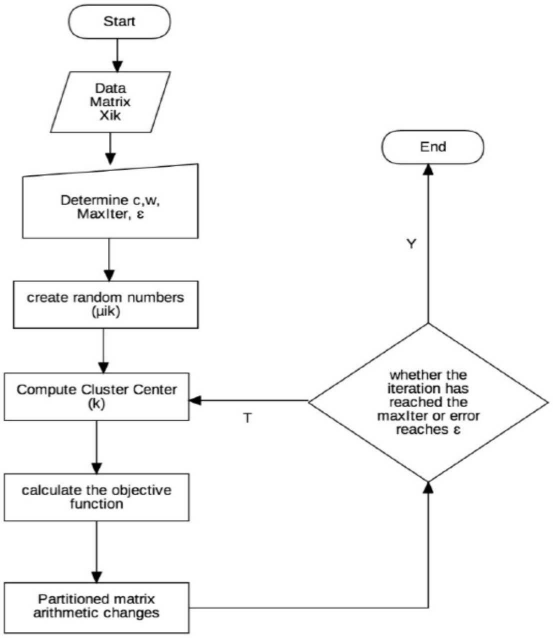

Start
Data Matrix Xik
Determine c,w, MaxIter, ε
create random numbers (μik)
Compute Cluster Center (k)
calculate the objective function
Partitioned matrix arithmetic changes
whether the iteration has reached the maxIter or error reaches ε
T
Y
End

Figure 2. Diagram of Fuzzy *C*-Means Algorithm

III. IMPLEMENTATION METHODOLOGY

For the purpose of testing the efficiency of K*(STAR) and FCM in Python, Students Results data set is used. Students Results Dataset: Total number of attributes is six i.e., SSCGPA, EAMCET, GENDER, I-I-GPA, I-II-GPA and Location. Among these attributes, I-I-GPA, I-II-GPA attributes are used for testing the efficiency of K*(STAR) and FCM in Python.

TABLE 1: Student Related Attributes and Data Types

S.NO	Variable	Description	Data Type
1	SSCGPA	Secondary School Certificate GPA	Numeric
2	EAMCET	Engineering Entrance Test	Numeric
3	GENDER	Student's Gender M or F	Nominal
4	I-IGPA	I-I Semester GPA	Numeric
5	I-II GPA	I-II Semester GPA	Numeric
6	Location	Area may be Rural or Urban	Nominal

Implementation of K*(STAR) Clustering:

In step 1, Calculate the required number of clusters based on dataset and a set of predicates dynamically and Apply standard K-means algorithm, or to calculate the required number of clusters by creating clusters dynamically, Apply the below steps.

In step 2, calculate the equation and get a concept about the number of clusters.

From step 3, assign a value in Clno for maintaining cluster number. By step 4, 5 and 6, algorithm assigns a new cluster and assigns data points to this cluster.

From step 7, algorithm finalizes the initial cluster's member data points, and gets decision to start a new cluster. Step 8; find the initial centroids for clusters.

By help of step 8, step 9 finalized a stable cluster and centroid. Step 9 finds the closest centroid for each data points in a systematic way of identifying boundary curve and allocates each data points Cluster [8]. Calculate new centroid for each Rm. Every time, the Asynchronous regions and corresponding data points are aligned into respective clusters

In step 10, 11, 12 and 13, there have tricks to find feasible data points those have chance to change current cluster. Calculate interval's points.

Normally other points don't move clusters. For this step, algorithm saves a lot of time. It minimizes a lot of calculation. To get decision, step 13 is used. In this step; algorithm take decision about the algorithm continue or all clustering is finished. At initial stage, data points of clusters can be scattered, During processing of this algorithm, the data points of asynchronous regions can be transformed in a systematic manner. By comparing boundaries of each region, the data points can be aligned to corresponding clusters.

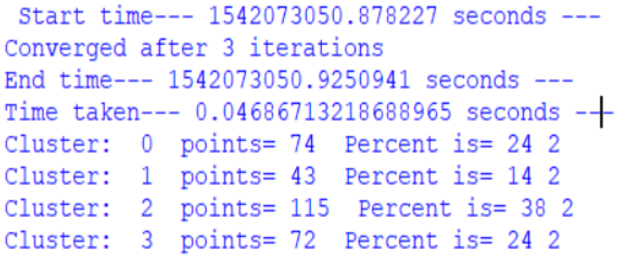
IV. EXPERIMENTAL RESULTS

This experiment reveals the fact that K*(STAR) clustering algorithm consumes less elapsed time i.e. 0.0468672 seconds than FCM clustering algorithm which takes 0.0624000 seconds depends on performance of the system and its resources. On the basis of the result drawn by this experiment it may be safely stated that K*(STAR) clustering algorithm less time consuming than FCM algorithm and hence superior.

Comparison of Time Complexity of K*(STAR) and FCM

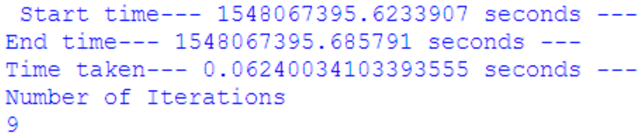
The time complexity of K*(STAR) [6] is O(ncdi) and time complexity of FCM [6] is O(ndc^{2}i). Keeping the number of data points constant we may assume that n = 304, d = 3, i = 3 and varying number of clusters where n = number of data points, c = number of cluster, d = number of dimension and i = number of iterations. The following table represents the comparison details.

K*(STAR) Results:



```
 Start time--- 1542073050.878227 seconds ---
Converged after 3 iterations
End time--- 1542073050.9250941 seconds ---
Time taken--- 0.04686713218688965 seconds --
Cluster:  0  points= 74  Percent is= 24 2
Cluster:  1  points= 43  Percent is= 14 2
Cluster:  2  points= 115  Percent is= 38 2
Cluster:  3  points= 72  Percent is= 24 2
```

FCM Results:



```
 Start time--- 1548067395.6233907 seconds ---
End time--- 1548067395.685791 seconds ---
Time taken--- 0.06240034103393555 seconds ---
Number of Iterations
9
```

TABLE II: Analysis of K*(STAR) and FCM time complexity based on performance of the computer

Algorithm	Time Complexity	Elapsed Time (Seconds)
K*(STAR)	O(ncdi) (6)	0.0468672
FCM	O(ndc^{2}i) (6)	0.0624000

TABLE III: Time Complexity Of K*(STAR) And FCM based on number of clusters and based on performance of the computer.

S.No.	Dynamic Number of Clusters	K*(STAR) Time Complexity	FCM Time Complexity
1	1	2736	2736
2	2	5472	10944
3	3	8208	24624

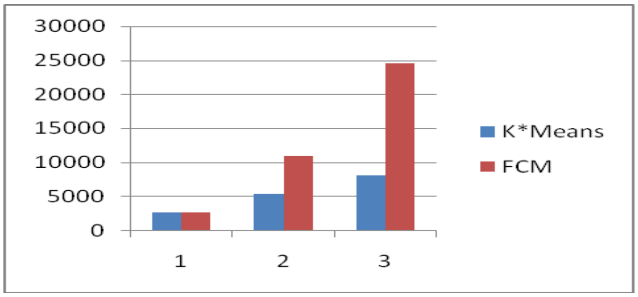


Figure3: Time Complexity of K*(STAR) And FCM based on number of clusters.

V. CONCLUSION

The study examined the available enrolment data of students in the university's database. Based on result from K*(STAR) clustering, Fuzzy C-Means algorithms, The Elapsed Time (Seconds) of K*(STAR) is 0.0468672 and The Elapsed Time (Seconds) of Fuzzy C-Means is 0.0671112.

K*(STAR) clustering algorithm can be processed to define the number of final clusters (k) dynamically based on data sets. The time complexity of the K*(STAR) algorithm is O (ncdi) and the time complexity of FCM algorithm is O(ndc^{2}i). From the obtained results we may conclude that K*(STAR) algorithm is better than FCM algorithm.

FCM produces close results to K*(STAR) clustering but it still requires more computation time than K*(STAR) because of the fuzzy measures calculations involvement in the algorithm. So, the final conclusion is that K*(STAR) algorithm is more efficient algorithm than Fuzzy C-Means algorithm.

REFERENCES

[1] J. C. Bezdek, "Pattern Recognition with Fuzzy Objective Function Algorithms", New York: Plenum Press, 1981

[2] M.-S. YANG, "A Survey of Fuzzy Clustering", Mathl. Computer Modelling Vol. 18, No. 11, pp. 1-16, 1993 Printed in Great Britain

[3] Dunn, J. C.(1973) 'A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well Separated Clusters', Cybernetics and Systems, 3: 3, 32 — 57

[4] Neha D, B.M. Vidyavathi, PhD," A Survey on Applications of Data Mining using Clustering Techniques", International Journal of Computer Applications (0975 – 8887) Volume 126 – No.2, September 2015

[5] Bora, DJ & Gupta, AK 2014 'A Comparative study Between Fuzzy Clustering Algorithm and Hard Clustering Algorithm'. Int. J. of Computer Trends and Technology, vol. 10, no. 2, pp. 108-113.

[6] A. Rui and J. M. C. Sousa, "Comparison of fuzzy clustering algorithms for Classification", International Symposium on Evolving Fuzzy Systems, 2006, pp. 112-117.

[7] Sheshasayee, A & Sharmila, P 2014 'Comparative Study of Fuzzy C-means and K-means Algorithm for Requirements Clustering'. Indian J. of Science and Technology, vol. 7, no 6, pp. 853–857.

[8] Velmurugan, T 2012 'Performance Comparison between K-Means and Fuzzy C-Means Algorithms Using Arbitrary Data Points'. Wulfenia Journal, vol. 19, no. 8, pp. 234-241.

[9] X. Rui, D. Wunsch II, "Survey of Clustering Algorithms", IEEE Transactions on Neural Networks, vol.16, no.3, 2005.

[10] Veerappa V, Letier E. Clustering stakeholders for requirements decision making. Requirements Engineering Foundation for Software Quality; 2011. pp. 202–08.

[11] Michael Delucchi, "Academic performance in college town", Education Vol.114 No,1 p96-100.

[12] Shailendra Singh Raghuwanshi, PremNarayan Arya,"Comparison of K-means and Modified K-mean algorithms for Large Data-set", International Journal of Computing, Communications and Networking, Volume 1, No.3, November –December 2012

Authors Profile

S N Ali Ansari Master of Computer Applications from Andhra University Visakhapatnam in 1996 and Master of Technology from Nagarjuna University in year 2009. He is currently pursuing Ph.D. and currently working as Associate Professor in Department of Computer Science, VSM College of Engineering,Ramachandrapuram India since 1996. His main research work focuses on Data Maining , Artificial Intelligence,. He has 23 years of teaching experience

V.Srinivasa Rao M.Tech (CSE), Ph.D. He is 27 Years Teaching Expirence and currently working as Professor & Head in Department of Computer Science & Engineering,V R Siddhartha Engineering College, Vijayawada , India. He is a member of ISTE, CSI. He has published more than 38 research papers in reputed international & national and conferences . His main research work focuses on Data Mining, Video Analytics and Bioinformatics.He visit Stanford University, University of Illinois at Chicago(UIC),, UC Berkeley, UC,Davis Loyola University Chicago (LUC) and Indiana University, USA