DEVELOPING AN INNOVATIVE CLUSTERING TECHNIQUES BASED ON KMEAN-BASED CONVEX HULL TRIANGULATION (KBCHT) GROUPING CALCULATION IN THE ENHANCEMENT INFORMATION MINING AND ARTIFICIAL INTELLIGENCE

Stuti Garg

ABSTRACT

Information Clustering is one of the most significant issues in information mining and AI. Clustering is an errand of finding homogenous gatherings of the examined objects. As of late, numerous analysts have a huge enthusiasm for creating grouping calculations. The most issue in a grouping is that we don't have earlier data information about the given dataset. In addition, the decision of information parameters, for example, the number of clusters, number of closest neighbours and different factors in these calculations make the grouping increasingly challengeable theme. In this way, any of the base decisions of these parameters yields awful clustering outcomes. Besides, these calculations experience the ill effects of unsuitable precision when the dataset contains clusters with various complex shapes, densities, sizes, commotion, and exceptions. Right now, propose another methodology for unaided grouping tasks. Our methodology comprises of three periods of tasks. In the main stage, we utilize the most generally utilized clustering system which is Kmean calculation for its effortlessness and speed by and by. We advantage just from one run of Kmean, despites its exactness, to find and break down the given dataset by getting fundamental clustering to guarantee intently gathering sets. The subsequent stage takes these underlying gatherings for preparing them in an equal manner utilizing contracting dependent on the curved body of the underlying gatherings. From the second stage, we acquire a lot of sub-groups of the given dataset. Henceforth, the third stage considers these sub-clusters for the consolidating process dependent on the Delaunay triangulation. This new calculation is named as Kmean-Based Convex Hull Triangulation grouping calculation (KBCHT). We present analyses that give the quality of our new calculation in finding groups with various non-curved shapes, sizes, densities, commotion and exceptions despite the fact that the awful starting conditions utilized in its first stage. These investigations show the predominance of our proposed calculation when contrasting and most contending calculations.

I. INTRODUCTION

A great deal of information can be assembled from various fields yet this information is pointless without appropriate investigation to acquire helpful data. Right now, center around one of the significant procedures in information mining: Clustering.

Information Clustering: Data clustering is a technique for gathering comparable articles. In this manner, the comparable items are bunched in a similar gathering and divergent articles are grouped in various ones. Information bunching is considered as an unaided learning procedure in which articles are assembled in obscure predefined groups. In actuality, the order is an administered learning wherein objects are allocated to predefined classes (bunches).

Fundamental Concepts of Clustering: The issue of information bunching can be detailed as follows: given a dataset D that contains n objects x1,x2,… ,xn (information focuses, records, cases, designs, perceptions, things) and every datum point are in a d-dimensional space, for example, every datum point has d measurements (qualities, highlights, factors, components). Data grouping depends on the likeness or uniqueness (separation) gauges between information focuses. Thus, these measures make the group investigation significant [1]. The high calibre of grouping is to get high intra-bunch similitude and low between bunch closeness. Moreover, when we utilize the uniqueness (separation) idea, the last sentence turns into: the high calibre of bunching is to acquire low intra-group difference and high between bunch disparity.

Importance of Clustering: Data clustering is one of the main tasks of data mining [2] and pattern recognition [3]. Moreover, it can be used in many applications such as

1.Data compression [4].

2.Image analysis [5].

3.Bioinformatics [6].

4.Academics [7].

5.Search engines.

6.Wireless sensor networks.

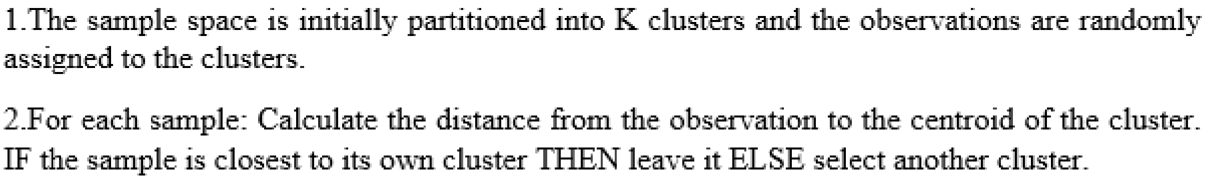
7.Intrusion detection.

8.Business planning.

II.ALGORITHMS USED

A. K-means

K-means is a technique for grouping perceptions into a particular number of disjoint branches. The "K" alludes to the number of groups determined [8]. Different separation estimates exist to figure out which perception is to be annexed to which group. The calculation targets limiting the measure between the centroid of the bunch and the given perception by iteratively annexing a perception to any group and end when the most reduced separation measure is accomplished.



1.The sample space is initially partitioned into K clusters and the observations are randomly assigned to the clusters.

2.For each sample: Calculate the distance from the observation to the centroid of the cluster. IF the sample is closest to its own cluster THEN leave it ELSE select another cluster.

B. Denclue

Lots of clustering algorithms could be applied in large multimedia databases. however,the efficiency and accuracy of the existing algorithm are limited. as the multimedia database has a high amount of noise and requires more clustering with high dimensions' feature vectors. In this paper, we are going to introduce a new algorithm that is DENCLUE. The essential thought of our new methodology is to demonstrate the general point thickness systematically as the total of in upgrade elements of the information focuses. Groups would then be able to be recognized by deciding thickness attractors and bunches of self-assertive shape can be effortlessly depicted by a straightforward condition dependent on the general thickness work.

C. FastDBSCAN

This algorithm can be divided and organized into two steps [10].

1.Partitioning the dataset by k-means and then use random method or Min-Max method to sample data.

2.Thereafter clustering the obtained data by DBSCAN

III. PROPOSED ALGORITHM

In this section, we will propose an enhanced technique of clustering algorithm i.e. enhanced Denclue, which helps to reduces noise in given dataset, for that we have enabled k-means with denclue. As we already know that k means algorithm is most famous algorithm in clustering technique.

A. Approach and methodology

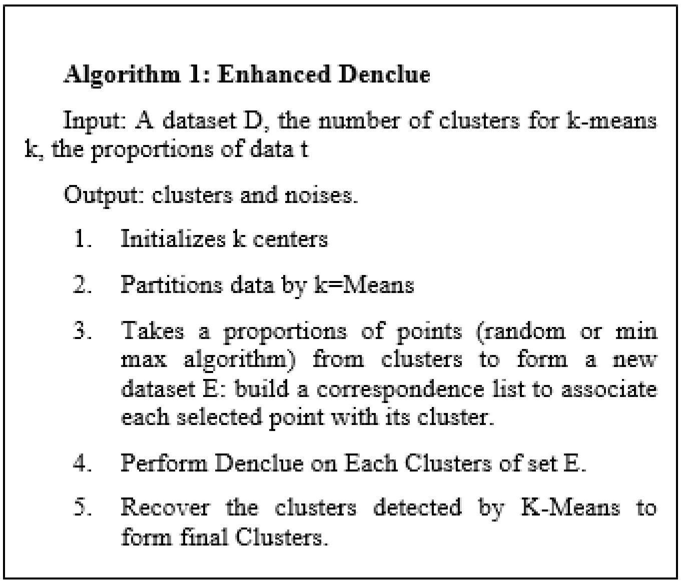
In our proposed algorithm we have mixed k-means clustering with denclue to reduce noise, for this we have taken chameleon dataset with two points.

The Key idea of our approach is divide and conquer method including 2 steps. This idea is also used in some research's such as the fast approximate spectral clustering fast minimum spanning tree algorithm.

To speed up the Performance of DENCLUE, we propose ENHANCE DENCLUE to remove noise

B. Enhanced Denclue Algorithm

In this algorithm, after applying k-means, we use random selection or min-max approach to select ‘t’ points for further cluster formation.

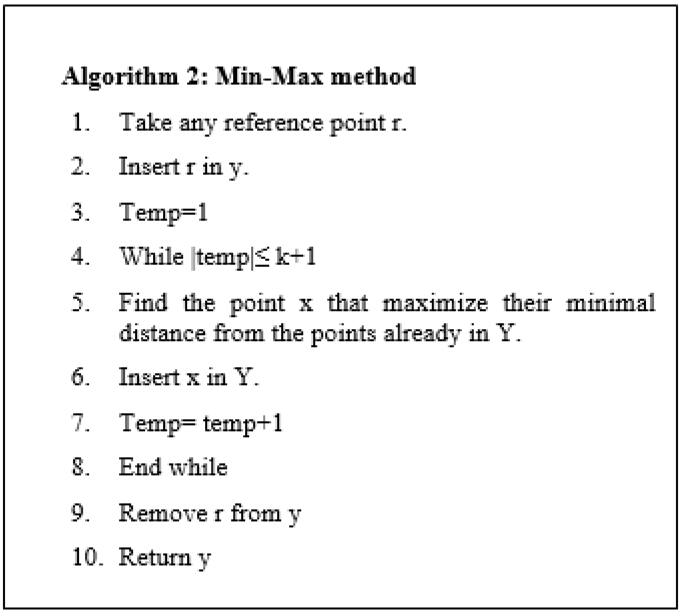


Algorithm 1: Enhanced Denclue

Input: A dataset D, the number of clusters for k-means k, the proportions of data t

Output: clusters and noises.

1. Initializes k centers
2. Partitions data by k=Means
3. Takes a proportions of points (random or min max algorithm) from clusters to form a new dataset E: build a correspondence list to associate each selected point with its cluster.
4. Perform Denclue on Each Clusters of set E.
5. Recover the clusters detected by K-Means to form final Clusters.



Algorithm 2: Min-Max method

1. Take any reference point r.
2. Insert r in y.
3. Temp=1
4. While |temp|≤ k+1
5. Find the point x that maximize their minimal distance from the points already in Y.
6. Insert x in Y.
7. Temp= temp+1
8. End while
9. Remove r from y
10. Return y

IV. EXPERIMENTATION AND RESULTS

A. Datasets

The dataset used is t4.8k which have already been used in evaluating DBSCAN and CHAMALEON algorithms.

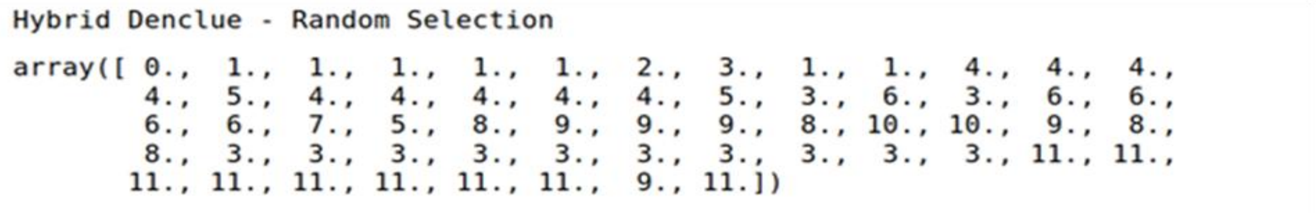
B. Evaluating System Hardware

- Processor:

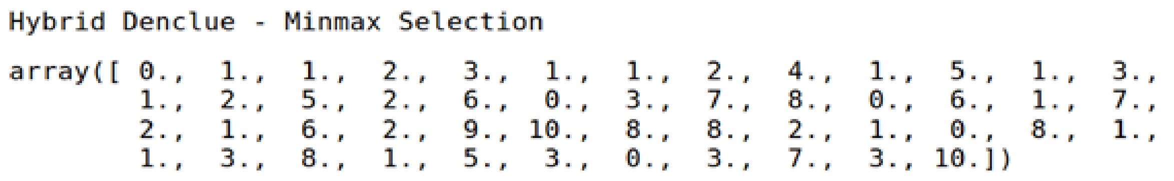
- RAM: 8.00GB
- Operating system: Windows 10 x64
- Program: Python

C. Result Comparison

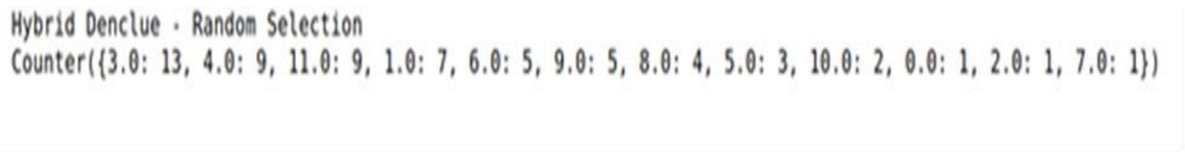
Following are the screenshots after performing Fast DBSCAN and Denclue on same dataset.



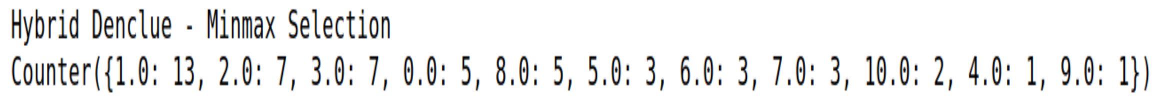
```
Hybrid Denclue - Random Selection

array([ 0.,  1.,  1.,  1.,  1.,  1.,  2.,  3.,  1.,  1.,  4.,  4.,  4.,
        4.,  5.,  4.,  4.,  4.,  4.,  4.,  5.,  3.,  6.,  3.,  6.,  6.,
        6.,  6.,  7.,  5.,  8.,  9.,  9.,  9.,  8., 10., 10.,  9.,  8.,
        8.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3.,  3., 11., 11.,
       11., 11., 11., 11., 11., 11.,  9., 11.])
```



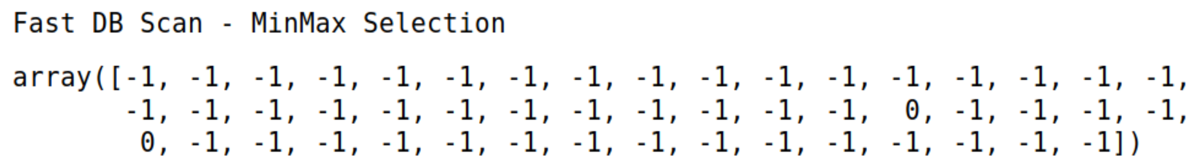
```
Hybrid Denclue - Minmax Selection

array([ 0.,  1.,  1.,  2.,  3.,  1.,  1.,  2.,  4.,  1.,  5.,  1.,  3.,
        1.,  2.,  5.,  2.,  6.,  0.,  3.,  7.,  8.,  0.,  6.,  1.,  7.,
        2.,  1.,  6.,  2.,  9., 10.,  8.,  8.,  2.,  1.,  0.,  8.,  1.,
        1.,  3.,  8.,  1.,  5.,  3.,  0.,  3.,  7.,  3., 10.])
```



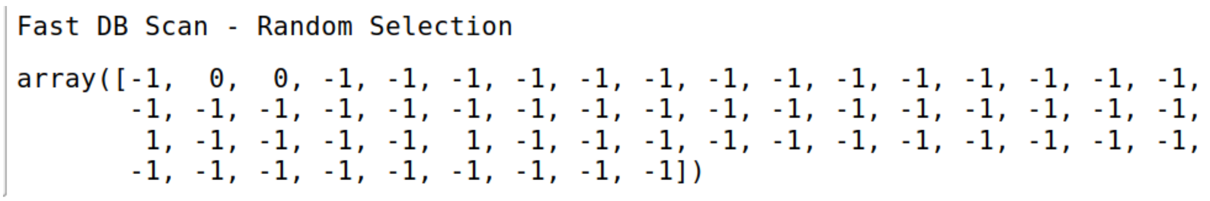
```
Hybrid Denclue - Random Selection
Counter({3.0: 13, 4.0: 9, 11.0: 9, 1.0: 7, 6.0: 5, 9.0: 5, 8.0: 4, 5.0: 3, 10.0: 2, 0.0: 1, 2.0: 1, 7.0: 1})
```



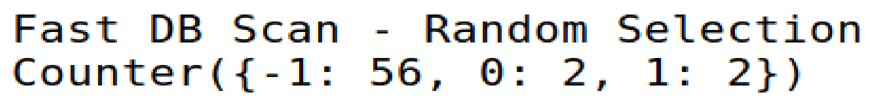
```
Hybrid Denclue - Minmax Selection
Counter({1.0: 13, 2.0: 7, 3.0: 7, 0.0: 5, 8.0: 5, 5.0: 3, 6.0: 3, 7.0: 3, 10.0: 2, 4.0: 1, 9.0: 1})
```



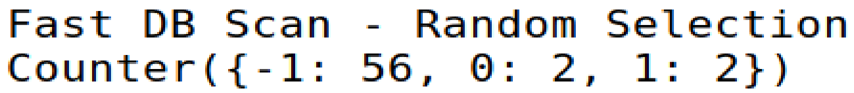
```
Fast DB Scan - MinMax Selection

array([-1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,
       -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,  0, -1, -1, -1, -1,
        0, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1])
```



```
Fast DB Scan - Random Selection

array([-1,  0,  0, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,
       -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,
        1, -1, -1, -1, -1,  1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,
       -1, -1, -1, -1, -1, -1, -1, -1, -1])
```



```
Fast DB Scan - Random Selection
Counter({-1: 56, 0: 2, 1: 2})
```



```
Fast DB Scan - Random Selection
Counter({-1: 56, 0: 2, 1: 2})
```

As the result shows, FastDbScan considers most points as '-1' i.e noise, whereas, our algorithm successfully classifies those points in clusters. Thus, hybrid denclue outperforms the FastDbScan which classifies most of the data as noise.

V. CONCLUSION AND FUTURE SCOPE

We have proposed an enhancement algorithm based on Denclue to cope up with the problems of an already existing clustering algorithm. Our proposed algorithm gives far better estimates of the number of clusters than existing FastDbScan. Exploratory outcomes show that our calculation is successful and productive and beat FastDbScan in distinguishing groups of various densities and in dispensing with commotions. The investigations show the effectiveness of the new calculations, and get the best outcomes with least blunders.

Future work will focus on improving the results for high dimensional dataset.