VOL. 10, NO. 23, DECEMBER 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences


www.arpnjournals.com

AN IMPROVED DENSITY BASED k-MEANS ALGORITHM

Kabiru Dalhatu[1] and Alex Tze Hiang Sim[2]
[1]Department of Computer Science, Faculty of Computing and Mathematical Science, Kano University of Science and Technology Wudil, Kano, Nigeria
[2]Department of Information Systems, Faculty of Computing, Universiti Teknologi Malaysia, Johor, Malaysia
E-mail: kabir2008@ymail.com

ABSTRACT

Clustering is a fundamental unsupervised data mining technique which is loosely defined as a process of arranging data objects into clusters based on similarity measures, k-Means is one of the most renowned clustering algorithm used across different domains, however k-Means suffers from multiple limitations with its results negatively affected by the presence of outliers. As a result of this limitation, k-Means algorithm has a series of its improvement algorithms among them is Outlier Detection Based on Density Approach k-means algorithm (ODBD-k-Means algorithm). Although this algorithm has better outlier detection accuracy, different results was given with different execution, this usually affect its clustering accuracy. In this paper, an improved algorithm was proposed to overcome the limitation of ODBD-k-Means algorithm. To denote the accuracy of the proposed Improved Density Based k-means algorithm (IDB-k-Means algorithm), an evaluation test was conducted using three different real-world datasets from UCI repository. Our experimental results shows that IDB-k-Means algorithm outperformed ODBD-k-Means algorithm in both the clustering and outlier detection accuracy.

Keywords: clustering, outlier, k-means, outlier detection, clustering accuracy.

INTRODUCTION

Clustering is an unsupervised classification of patterns (data items, feature vectors or observation) into groups (clusters) which had been address in different contexts by many researchers in different domains across the globe. Clustering being unsupervised in nature, is arguably difficult to manage compared to that of supervised class of data manipulation algorithms such as classification algorithms. The clustering algorithms are of two categories that is, partitioning and hierarchical. K-means clustering algorithm is a well-known partitioning type of clustering algorithm used across different domains due to its simplicity (Abubaker and Ashour, 2013), it is mostly used in scientific and industrial applications. This algorithm attempts to reduce the means square error function and works mostly with numerical attributes (Rai and Singh, 2010), k-means algorithm however suffers from several obvious drawbacks as follows:

i. The number of clusters in a given data set has to be defined in advance.
ii. Its result strongly depends on an initial and selected centroid.
iii. It may generate empty clusters.
iv. It only detects globular shape type of cluster.
v. It only works on numeric data.

To handle the above mentioned problems faced by k-means clustering algorithm, a series of k-means enhanced algorithms were proposed such as k-median, k-means++ and k-means--, dbk-means and odbd-k-means.

In addition to the above mentioned limitations, k-means is sensitive to outlier. The latter would swift cluster centroid towards outlier, leading its convergence to a, arbitrarily, false cluster, quickly reduces the algorithm's clustering accuracy(Yektaei, 2013).

Outliers are pattern in data that do not conform to the well-defined notion of the outlying behaviour, an outlier is often the anomalous, noise, discordant, exception, fault, error, contaminants even novelty (exceptional) discovery in an application domain(Chandola *et al.* 2007). Outliers mostly occur as a result of instrumental error, human error, environmental changes and/or malicious activities. Despite the subsequent k-means improvement algorithms which improves on k-means, it suffered from one or more limitations. In this paper, we present an improved version of ODBD-k-means algorithm to minimise some of its drawbacks.

The rest of the paper is organized as follows: We discuss the literature review of clustering algorithms in section 2. We present our algorithm in section 3. In section 4, we conducted an experimental test to prove the efficiency of our proposed IDB-k-Means algorithm using 3 real-world datasets. Section 5 concludes with some directions for future research.

RELATED WORK

There are several literatures written for classical k-Means improvement based on outlier detection and its removal using different outlier detection techniques. Historically, outlier detection was initially formulated using statistical approach which was further categorized into two main classes. In the first, outliers are extracted out of the dataset (by the use of statistical distribution such as poison and probability distribution for multiple discordancy test (V. and T., 1994), for example). These approaches, however, is limited to univariate-distribution and there are numbers of multivariate distribution problems to be considered. Secondly, the depth based approach works by assigning depth to each data point

based on its position on the plane, where points within the shallow depth are most likely to be considered as outliers. Practically this approach is inefficient when dealing with dataset with k>3, this is because this approach depends on k-d convex hulls computation with lower bound complexity of Ω(nk/2) for n objects (Breunig *et al.* 2000). In general, to use this approach of outlier detection an apriori knowledge of data distribution is needed in advance. A distance based outlier detection technique was formulated by Knorr and Ng(Knorr and Ng, 1999). With a view of overcoming the limitation of the subsequent approach by the use of k-nearest neighborhood (KNN) approach. It works by calculating the distance between the most nearest neighbor points and ranks them based on their proximity where points with the highest proximity are consider to be outliers (Knorr and Ng, 1999). One of the major limitation of this approach is finding the optimum normal and outlier values such that outliers would be detected correctly (Hautamaki *et al.* 2004). This approach performed poorly with very large dataset and high dimensional dataset. It's mainly used for detecting globular shape cluster. A local outlier factor(LOF) was proposed by Markus et al by assigning a degree to which an object will be regarded as outlier based on how isolated it is with respect to its neighbors, through the use of density based outlier detection approach (Breunig *et al.* 2000). Even though this approach works well, it suffers from computational complexity limitation as the computation of LOF of each data element requires a number of k-nearest neighbor search, to overcome the preceding limitation, a micro cluster technique was proposed by Jin at el for mining n-top outliers where the LOF of those data objects that are most likely to be outliers will be computed only because majority of the data object are not outliers (Jin *et al.* 2001). A density based algorithm for discovering clusters in spatial databases with noises(DBSCAN) was proposed by (Ester *et al.* 1996) through the use of neighborhood radius (Eps) and minimum number of points (minpts) parameters in order to detect both individual and group of outliers called micro cluster. This algorithm works by calculating the density of each data point in relation to its neighborhood within a particular radius and the group of data points were declared as outliers if they are less than the minpts value. Due to the popularity and efficiency of the two most proceeding techniques that is distance and density based outlier detection, many k-means enhancement literature based on outlier detection were written using them. Among these is ODBD-k-Means algorithm, this is one of the k-means enhancement algorithms proposed by (Yektaei, 2013) with a focus of detecting outliers and reducing their unwanted effect in k-Means processed data by the use of density based outlier detection approach. This algorithm consist of two phases where the first phase separated outliers from the normal data points by the used of dissimilarity measure mechanism while the second phase clustered the normal data points by the use of k-Means algorithm. The outliers points were then assigned to the most nearby clusters, even though this algorithm improved the clustering accuracy of k-means algorithm based on the evaluation test, it generates different results upon different executions due to the random selection of initial centroids.

METHODOLOGY

Given the subsequent definition of outlier and the limitation of ODBD-k-Means algorithm in trying to improve the efficiency of traditional k-means algorithm by handling its sensitivity to outlier limitation, our problem is to formulate a new algorithm that serves as an enhancement version of ODBD-k-Means algorithm. Our algorithm could detect and delete outlier while maintaining consistent result given the same dataset and threshold value. In this section, we are going to provide a brief explanation of the terms, concept and analysis of the proposed algorithm for overcoming ODBD-k-means algorithm limitations.

Related concept

Definition 1: Euclidean distance is one of the simplest formula used for calculating distance between points in the same way as we measure it with ruler. A Euclidean distance formula for similarity measures is shown below.

$$d(c,p) = \sqrt{\sum_{i=1}^{n}(\mathrm{Ci} - \mathrm{Pi})^2} \quad (1)$$

Where, d(C,P) represents the distance between the closest cluster center C and any data point P in dataset D, n represents the total number of dataset points with a *i* is a counter.

Definition 2: The density value based on IDB-k-Means algorithm was calculated using an in degree formula by using the above *d(C,P)* as an input.

$$\rho_i = \frac{1}{d(C,P)+1} \quad (2)$$

Where, ρi and di stand for density and distance respectively.

Definition 3: Threshold (*Eps*) is a term representing the neighborhood radius of a data points say, *k*, denoted by *NEps(k)* which is mathematically defined as

$$\mathrm{NEps(p)} = (g \in k \mid d(c,p) \leq \mathrm{Eps}) \quad (3)$$

Where, Eps represents threshold value, NEps(p) represents neighborhood of p within the threshold value and the distance, d(c,p) stands for the distance between c and p.

Definition 4: The process of finding cluster center *C* is quite different with that of the traditional clustering method where the cluster center was found at random, which usually affects the clustering accuracy of the algorithm. As in the case of IDB-k-Means algorithm, an initial imaginary cluster center was assigned to a point

with maximum density value. After which, the cluster center will be systematically shifted from one point to another within the cluster until convergence. Below is a figure depicting the multi-center scenario of IDB-k-Means algorithm.

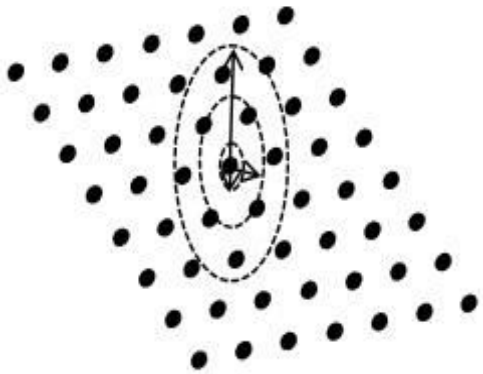

Figure-1. Multi-cluster center scenario.

When referring to Figure-1 above, the cluster center is initially assigned to a point with the highest density value which usually falls inside a cluster based on the definition of density. The proposed algorithm will then use the cluster center together with a threshold value to iteratively assign all points whose distance from the cluster center is within the threshold value into the cluster, having assign all the points to the cluster using the same cluster center. The initial cluster center will be shifted to any other point within the cluster as depicted in the above figure continuously until convergence.

Clustering and proposed methods

Clustering is the process of dividing dataset into groups based on similarity measures and it consist of a lot of algorithm main for such grouping among which k-Means algorithm is among the well-known clustering algorithm.

The hybridization between traditional k-means algorithm and outlier detection techniques is among the common practice used by many k-means enhancement algorithms mainly for overcoming the limitation of k-mean's insensitivity to outlier. k-means algorithm is a traditional clustering algorithm that assigns clusters centers at random, which usually affects its clustering accuracy based on the fact that an outlier can mistakenly be selected as cluster center. Unlike the traditional ways of clustering and outlier detection, the proposed algorithm provides an efficient clustering and outlier detection based on outlying-ness factor approach, and it works by using the point with maximum density value as a cluster center, instead of random selection of cluster center. The cluster center will be used as a reference point for data clustering. The proposed algorithm cluster dataset by the used of threshold value and cluster center, where the distance between the cluster center and the remaining data points will be calculated and compared with the threshold value and those points whose distance are within the threshold bound to the current cluster are assign to the cluster center. The momentarily cluster center will then be shifted to another point within the cluster. The process iterated with repeated calculation on the distance between the new and momentarily centroid and the remaining data points (excluding those who were already in the cluster as depicted in Figure.1), this process continues until all points belonging to the first cluster are clustered and removed out of the dataset. It follows that subsequent search and clustering will continue until all dataset points are appropriately clustered. Having done that, all data points that are not in either of the clusters will be regarded as outliers thus, detected and removed. Below are steps of the proposed algorithm.

Input: A dataset D containing n number of points, number of clusters k.
Output: k clusters with maximum clustering accuracy and number of outliers.
Step 1. Calculate the distance (dist) between all dataset points, say a,b,c depicted in Figure-2 below.

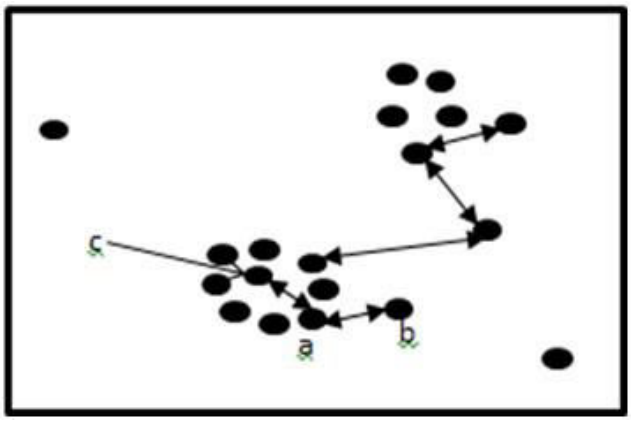

Figure-2. Distance calculation.

Step 1.1. Calculate the density of each data points using Equation. 2
Step 2. Find the threshold value T by using the mean value of the calculated distance and then keep incrementing or decrementing it until an appropriate T is found.
Step 3. Select a point with maximum density value say C (imaginary cluster center) from Figure-2 and assign it to CLUST(temporary memory space)
Step 4. Calculate the distance between C and any data point P in D, if the distance between C and P is less than or equal to T, then assign P into CLUST and delete it from D.
Step 5. If the points inside CLUST are greater than 1, then calculate the distance between any point in CLUST excluding C and P in D which were initially compared with C, and if the condition didn't hold assign P into CLUST whose distance from any point in CLUST is less than or equal to T and delete it from D.

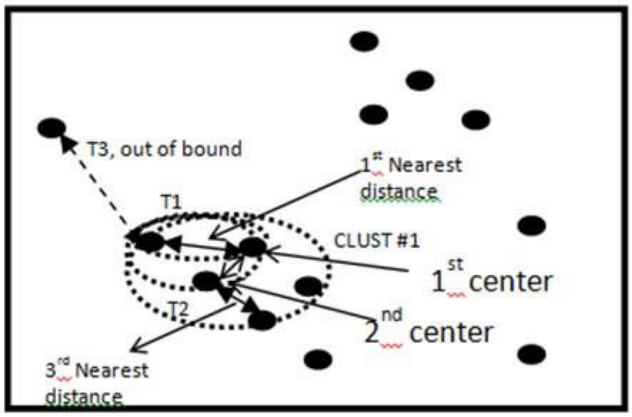

Figure-3. First cluster processes.

Step 6. Repeat steps 4 and 5 until no any point P in D, whose distance from that of any points in CLUST is less than or equal to T.
Step 7. Increment number of CLUST and Repeat step 3.
Step 8. Repeat step 7 until all points are appropriately clustered
Step 9. Delete all remaining points that is outliers in D that didn't belong to either of the CLUST.

Outlier detection in dataset

The dataset point can be concluded by the proposed algorithm to be outlier if after clustering all the dataset points based on the number of cluster given to the proposed algorithm, there are still remaining dataset points that are not in either of the clusters can be detected by cross marking them, as depicted in Figure-4 below by the proposed algorithm and hence deleted. This number of outliers remain fixed irrespective of the number of the algorithm is executed provided the threshold value remain the same.

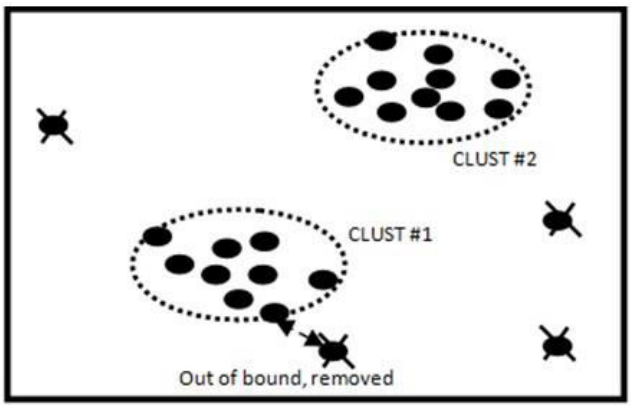

Figure-4. Appropriate clusters with detected outliers.

Experimental design

To further prove how efficient IDB-k-Means is over traditional k-Means and ODBD-k-Means algorithms, this paper uses Iris, Glass and Bupa real datasets from UCI repository for the experiment. Below is a table depicting the names, attributes and instances of these datasets.

Table-1. UCI dataset for the experiment.

Name	Attribute	Instance
Iris	4	150
Glass	9	214
Bupa	6	345

UCI repository is a machine learning datasets repository that group datasets into different specialization for machine learning experiments. Before executing our proposed algorithm in section 3.2 using the above datasets, these datasets are normalized and divide into training and testing parts by the used of k-fold cross validation technique. We then set the value of cluster (K) and threshold (T) as depicted in Table-2 below

Table-2. Parameters used for running of IDB-k-Means algorithm.

Name	Number of cluster (K)	Threshold Value (T)
Iris	*3*	*0.0290*
Glass	*6*	*0.0078*
Bupa	*2*	*0.0100*

It is important to note that the above threshold values are not fixed nor a standard. These were found by calculating the mean value of the distances between points and then keep incrementing or decrementing through testing until an appropriate T is detected. This is because it is quite difficult to formulate a threshold formula that can satisfy both datasets.

RESULTS AND DISCUSSION

After getting the above parameters we turn to run our algorithm, the summary of the experimental results on percentages clustering accuracy is shown in Table-3 below.

Table-3. The results of algorithms on datasets.

DATASET	*k-Means*	*ODBD-k-Means*	*IDB-k-Means (the proposed)*
IRIS	88.98	90.36	95.33
Bupa	52.43	53.72	84.4
GLASS	45.79	47.09	48.15

The above datasets results on k-Means and ODBD-k-Means algorithm was found from(Yektaei, 2013). As observed from Table-3 the percentage of clustering accuracy of IDB-k-Means is significantly improved in Bupa, Iris and slightly improved in Glass dataset due to the overlapping nature of Glass dataset when compared with both k-Means and ODBD-k-Means algorithms. This is because both k-Means and ODBD-k-Means algorithms used the traditional way of random approach for cluster center selection while IDB-k-Means algorithm considered maximum density value as an imaginary cluster center, this cluster center will be shifted from one point to another same growing cluster (based on the same consideration on density) until all points within the same cluster are packed together. This made our algorithm to be a multi centroid algorithm. This is quiet efficient when compared with the traditional single centroid approach where the cluster growth based on threshold length and any other point outside threshold coverage area will be regarded as outlier, thus many normal points may be mistakenly be regarded as outliers and vice versa. The latter commonly happen on irregular shape type of clusters. In addition, based on the properties of IDB-k-Means algorithm, the results found remain fixed after a number of executions although different number of clusters was provided with its threshold value remain constant. Below is a figure depicting the graphical representation of clustering accuracy results presented in Table-3 above.

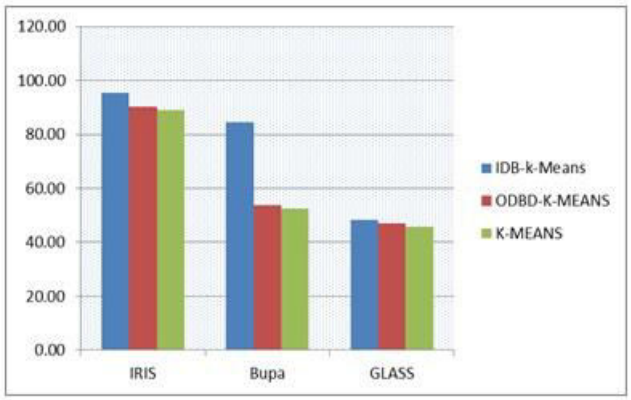

Figure-5. Clustering accuracy comparison between IDB-k-Means, ODBD-k-Means and k-Means algorithms.

Below is a Table-4 depicting the comparison between IDB-k-Means and ODBD-k-Means algorithm based on outlier detection

Table-4. Result of IDB-k-Means and ODBD-k-Means algorithms on outlier detection.

	IRIS	Bupa	GLASS
IDB-k-Means	4.00	10.00	8.00
ODBD-K-MEANS	5.00	12.00	8.00

Looking at the results presented in the above Table-4 it clearly shows that IDB-k-Means algorithm outperformed ODBD-k-Means algorithm on both Iris and Bupa while performed ultimately the same in Glass datasets due its overlapping nature, this is because the clustering accuracy of an algorithm increased with the decrease of outlier. Below is a figure depicting the graphical representation of outlier detection accuracy results presented in Table-4 above.

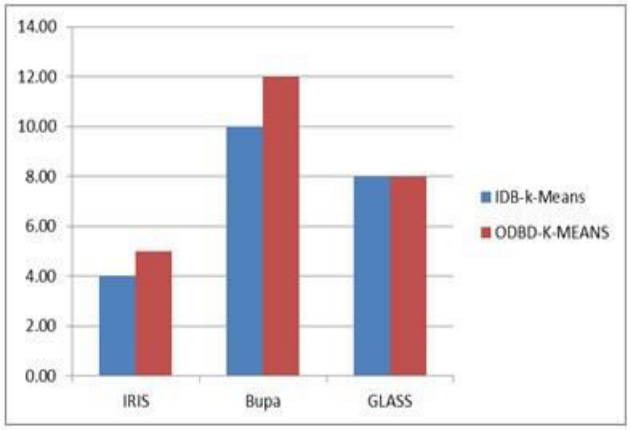

Figure-6. Result of IDB-k-Means and ODBD-k-Means algorithms on outlier detection.

Based on the above clustering and outlier detection results, it shows that IDB-k-Means algorithm outperformed ODBD-k-Means algorithm in both aspect. Based on this finding IDB-k-Means algorithm is concluded as a suitable alternative to ODBD-k-Means algorithm, which in turn, improved on k-Means significantly.

CONCLUSIONS

In this paper, a new algorithm was presented based on density based outlier detection approach, where a point with maximum density value is used as an imaginary cluster center which will be shifted from one point to another within a cluster until all points belong to a particular cluster are well separated, the process continued with the number of cluster incremented until all data points are accurately divided into clusters. The remaining points are the outliers hence removed. This algorithm was proposed with a view of solving the limitation of ODBD-k-Means algorithm, which improved on k-Means significantly. The experimental results found during our algorithm executions remain fixed in different executions provided the threshold value remain constant, this solved one of the ODBD-k-Means limitations and similarly outperformed it in both clustering and outlier detection accuracy respectively. For future studies, we hope to formulate an equation for determine the required threshold irrespective of the dataset.

REFRENCES

[1] Abubaker, M. and Ashour, W. 2013. Efficient Data Clustering Algorithms: Improvements over Kmeans. I.J. Intelligent Systems and Applications, 37-49.

[2] Breunig, M. M., Kriegel, H.-P., NG, R. T., and Sander, J. 2000. LOF: Identifying Density-Based Local Outliers. In Proceedings of 2000 ACM SIGMOD International Conference on Management of Data. ACM Press, 93-104.

[3] Chandola, V., Banerjee, A., and Kumar, V. 2007. Outlier Detection: A Survey. ACM Computing Surveys, 1-83.

[4] Ester, M., Kriegel, H.-P., Sander, J., and XU, X. 1996. A Density-Based Algorithm for Discovering Clusters In Large Spatial Databases With Noise. Proceed of 2nd International Conference on Knowledge Discovery and Data Mining., 226-231.

[5] Hautamaki, V., Karkkainen, I., and Franti, P. 2004. Outlier Detection Using K-Nearest Neighbour Graph. in Proceedings OF 17th International Conference on Pattern Recognition (ICPR'04), 3, 430-433.

[6] Jin, W., Tung, A. K. H., and Han, J. 2001. Mining TOP-N Local Outliers in Large Databases. In Proceedings of The Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 293-298.

[7] Knorr, E. M. and NG, R. T. 1999. Finding International Knowledge of Distance-Based Outliers Proceeding of the 25th VLDB Conference, Edinburgh, Scotland, 211-222.

[8] Rai, P. and Singh, S. 2010. A Survey of Clustering Techniques. International Journal of Computer Applications, 7(12), 1-5.

[9] V., B. and T., L. 1994. Outliers in Statistical Data. John Wiley.

[10] Yektaei, B. A. S. H. M. H. 2013. Detection of Outliers and Reduction of Their Undesirable Effects for Improving the Accuracy of K-Means Clustering Algorithm. International Journal of Computer Applications Technology and Research, 2(5), 552-556.