Research Journal of Applied Sciences, Engineering and Technology 14(8): 281-290, 2017
DOI:10.19026/rjaset.14.4952
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: December 14, 2016 Accepted: May 25, 2017 Published: August 15, 2017

Research Article

A Refined Rough K-Means Clustering Algorithm based on Minimizing the Effect of Local Outlier Objects to Improve Overlapping Detection

Khaled Ali Othman, Md. Nasir Sulaiman, Norwati Mustapha and Nurfadhlina Mohd Sharef
Department of Computer Science, Faculty of Computer Science and IT Putra University of Malaysia, Selangor, Malaysia

Abstract: In order to improve the quality of overlapping detection, Rough K-Means (RKM) was proposed as the first kind of rough clustering algorithm. It was found that this recent RKM algorithm known as π RKM is the most powerful and effective version in which there is an increase in the number of objects that are correctly clustered and a decrease in the number objects that are incorrectly clustered compared to the issues which the previous RKM had. However, there are challenges associated with the clustering process which uses RKM as a result of the difficulty in establishing a standard measure for reducing the effect of local outlier objects on a means function. Therefore, the RKM algorithm is refined in this study to address the problem. Through this study we contribute two components. Firstly, we intend to employ the use of Local Outlier Factor (LOF) technique for the discrimination of a number of objects as outliers and secondly, we propose to reduce the effect of local outliers on means function by using a weight. The result of the experiments which were performed through the use of synthetic and real life datasets prove that there is an improvement in the quality of overlapping detection when compared to recent versions.

Keywords: Clustering, data analysis, local outlier factor, Rough K-Means

INTRODUCTION

K-Means which is also regarded as Hard K-Means is a clustering algorithm that is simple and unsupervised (Hartigan and Wong, 1979). The purpose is to group similar objects into a given cluster as well as different objects into an appropriate cluster by partitioning the natural structure of data objects. K-Means is regarded in the literature as of the frequently used clustering algorithms which for over 50 years has been in use (Jain, 2010; Xiao and Yu, 2012) several domains of application (Peters *et al*., 2013). However, it was found that this popular algorithm is weak because of its inability to differentiate objects that are vague or ambiguous. So, in order to address the shortcomings of this algorithm soft clustering algorithms like Fuzzy C-Means (Bezdek and Harris, 1978) and its derivatives such as Possibilistic C-Means (PCM) (Krishnapuram and Keller, 1993).

One of the major aims of clustering algorithms is to detect objects that are overlapping. Rough clustering is considered as a unique approach that adopts the interpretation of rough set properties in partitioning algorithms. The first algorithm to adopt this approach is the Rough K-Means (RKM) (Lingras and West, 2004). The aim of this algorithm is to distinguish objects that overlap between positive clusters based on the process of Hard K-Means. As a solution for each cluster, the lower and upper approximation is initiated (a brief description of each approximate space is provides in related work).

Some of the improved versions are introduced to achieve satisfactory RKM clustering results such as that in Peters (2006, 2012) which minimize the effect of the objects in the upper regions against the objects in the lower region. Recently, Peters (2014) further refined the RKM algorithm which was introduced as the Laplace's Principle Indifference as a method of improving the overlapping detection quality. A rough classifier (Peters, 2015a) was introduced as a new validity index and used to evaluate the experiments results of RKM algorithm. The experiments results found that, the number of correctly clustered objects has been increased and the number of incorrectly clustered objects has been decreased in comparison to previous RKM and classical K-Means (Peters, 2015b). However, the currently available algorithm has a weakness in minimizing the effect of local outlier objects on the means function. In this study, we contribute in refining the RKM clustering algorithm by handling the problem mentioned above. A weight (w) is proposed to minimize the effect of outlier objects on the

Corresponding Author: Khaled Ali Othman, Department of Computer Science, Faculty of Computer and IT, Putra University of Malaysia, 43400, Serdang, Selangor, Malaysia

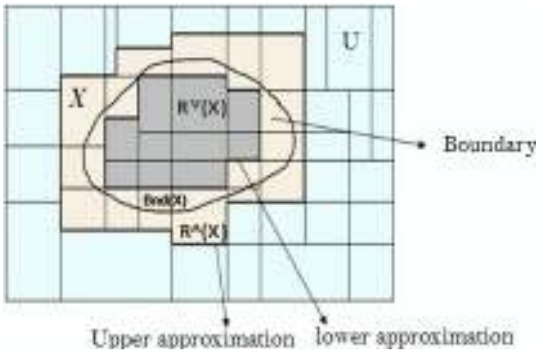


Fig. 1: Definitions of approximation space

means function. A method called Local Outlier Factor (LOF) (Breunig *et al*., 2000) is used as a measure to distinguish the number of outlier objects. The effectiveness of the proposed weight is demonstrated on synthetic and real datasets from Iris Plant and Vowel dataset. Results indicated that the number of the correctly clusters increased. In contrast, the number of incorrectly clusters is decreased.

LITERATURE REVIEW

Rough clustering which was first introduced in Lingras and West (2004) is derived from the interval interpretation of rough sets (Pawlak, 1982) in contrast to clustering algorithm. For instance, the K-Means algorithm is modified by incorporating the concepts of rough approximation space. Generally, approximation is a fundamental construct that distinguishes the rough set from other approaches. The key concept of approximation (rough) is the isolation of the indiscernible form objects into lower and upper approximation. The lower contains the objects that only belong to one cluster; and the upper contains objects that belong to more than one cluster. Figure 1 depicts a definition of approximation in rough concept.

Assuming U (called Universe) is a certain nonempty set of objects $X = \{x_1, x_2, \ldots, x_n\} \in R$, where R is an equivalence relation of the U and the pair (U, R) called the approximation space. Hence, U divides the space into the three regions as following:

- The lower approximation region is $R_V(X)$, (also called the positive region $Pos(X) = R_V(X)$).
- The upper approximation region $\hat{R}(X)$, (also called the negative region $Neg(X) = \hat{R}(X)$).
- The boundary region $Bnd(X) = \hat{R}(X) - R_V(X)$. The boundary region is generally not spatial, where it is just for gathering ambiguous objects not related to any positive region, ($Bnd(X) = Neg(X) - Pos(X)$.

In rough clustering approach, all the objects in the positive region belong to one cluster, while all objects in the negative regions; possibly belong to two or more clusters (Peters, 2006). The basic properties can be outlined as follows:

- $X \in R_V(X)$, for $R_V(X) \subset \hat{R}(X)$ & $X \notin Bnd(X)$,
- If $X \in R_V(X)$, then also $X \in \hat{R}(X)$,
- If $X \notin R_V(X)$, then $X \in Bnd(X)$,

According to some perspectives, these basic properties are not necessarily independent or complete (Mitra *et al*., 2006). However, enumerating them will be helpful in understanding how the rough set is adapted into Hard K-Means algorithm (Lingras and Peters, 2011). An example of three rough clusters (e.g., RKM) is shown in Fig. 2.

Therefore, the essential effort of RKM algorithm includes the calculation of the means of "Centroids" and the assigning of the object to the cluster

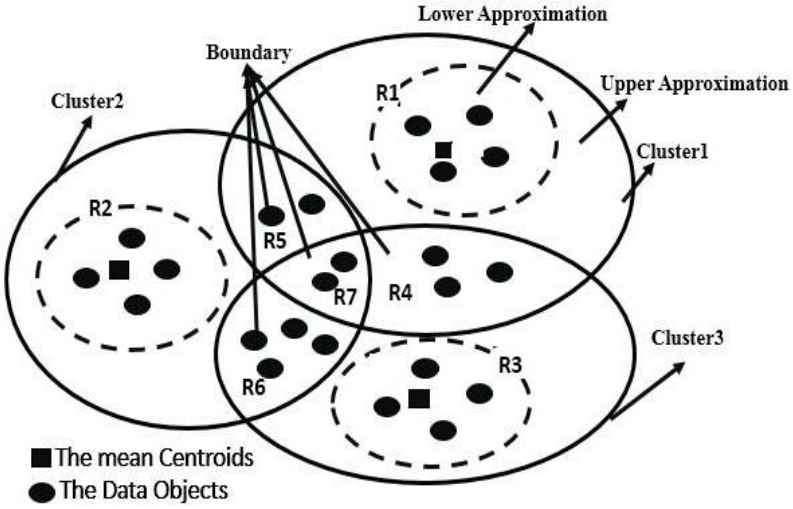


Fig. 2: Three rough clusters (e.g., RKM)

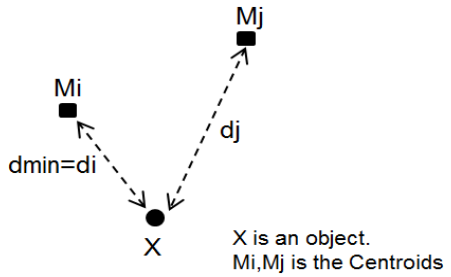


Fig. 3: Assigning part of the algorithm

region/regions based on three input factors such as follows: Firstly, estimation of the number of clusters k (finding the value of k is based on a trial or error process). Secondly, two weights are set as parameters ($wR_V(X)$, $wR^{\wedge}(X)$) (which represent the linear combination of lower and upper parameters. Thirdly, determination of the size of the boundaries by using a threshold (T). The objects are then assigned either to the lower or the boundary regions based on distance from the positive cluster centroids. A Laplace's (Laplace, 1998) distance is used as a measure for assigning the objects. At this point, the numbers of objects in the boundary region would be increased by increasing the value of rough clustering threshold.

Recently, several RKM versions have been proposed (Peters, 2006, 2014, 2015b). In fact, most studies focus on improving the algorithm to be more robust and effective based on the input factors mentioned in the previous paragraph. Some of the significantly improved versions of RKM have been introduced. First, Peters (2006) made some refinements. His studies recommend that the weight of an object in lower region $wR_V(X)$ should be higher than the weight of the object in the upper region (in this case boundary region $Bnd(X)$). The alternative proposed set $wR_V(X)$ = 0.7, where $wR^{\wedge}(X)$ =1- $wR_V(X)$ is used for calculating the means (M_k) of the cluster. Respectively, the improved means function is presented in Eq. (1) as follows:

$$M_k = \left[wR_V(X)\ \sum_{i=1}^{n} \frac{X_i}{R_V(X_n)}\right] + \left[wR^{\wedge}(X) \sum_{i=1}^{n} \frac{X_i}{R^{\wedge}(X_n)}\right]$$
$$[\text{For } R_V(Xn)^{1}\emptyset; \text{ with } wR_V(X) + wR^{\wedge}(X) = 1] \quad (1)$$

He too applied Relative distance for assigning part instead of the Laplace's distance method proposed in the initial version. An example of using the relative distance measure is depicted in the Fig. 3. Assumes Mi and Mj are two means of clusters. Hence, the minimum distance (d_{min}) between the object X and the closest means Mi. Meanwhile, dj is a distance between the object X and other means Mj. In this case:

$$d_{min} = d(X, Mi) \quad (2)$$

The relative distance Eq. (3) is used in determining if the object is overlapped or non-overlapped and is computed as follows:

$$T' = \left(\frac{d_j}{d_i}\right) \geq T$$
$$[where\ (i \neq j), T' \text{ is boolean return}] \quad (3)$$

Lately, an important refinement of RKM algorithm was presented (Peters and Lingras, 2014; Peters, 2014). Moreover, a method called Laplace's Principle of Indifference (Laplace, 1998) is applied to determine the weights in the mean function of RKM algorithm. The existing algorithm version called πRKM. The main concern is replacing the variant weights of RKM by neglecting the number of objects in lower and upper regions.

To understand Laplace's applied method in RKM, Fig. 2 illustrates this. The three clusters (Cluster1, Cluster2 and Cluster3) distribute the data objects into 7 possible regions $R1$, $R2$, $R3$, $R4$, $R5$, $R6$ and $R7$. Hence, $R1$, $R2$ and $R3$ represent the Positive regions, where the objects are not overlapping with other regions. In this case, $R_{1V}(X) = R_{2V}(X) = R_{3V}(X) = 1$, where the effect of these objects on region $R^{\wedge}(X) = 1/1 = 1$. In the other case, the objects in $R4$, $R5$ and $R6$ belong to two clusters, where the effect of these objects on region is represented by $R^{\wedge}(X) = ½ = 0.5$. In contrast, the same applies to $R7$, where each object belongs to three clusters, where the effect of these objects on region is denoted by $R^{\wedge}(X) = 1/3 = 0.3$. As a consequence, the effect of the objects would decrease, when the number of belongs regions increased. Formally, the means function is extended as below:

$$M_k = \left(\sum_{i=1}^{n} \frac{X_i}{R^{\wedge}(X_i)}\right) / \left(\sum_{i=1}^{n} \frac{1}{R^{\wedge}(X_i)}\right)$$
$$\left[where\ R^{\wedge}(X) \neq \emptyset\right] \quad (4)$$

It should be noted that, besides the original RKM improved version, there are other extensions of RKM algorithm that attempt to improve the quality of an algorithm by studying the optimization parameters. This include studies such as evolutionary rough clustering (Lingras, 2009; Mitra, 2004; Peters *et al*., 2008) where the initial parameters are optimized in relation to cluster validity indexes. The hybrid clustering which combines rough with fuzzy or possibilistic approaches have been proposed by researches like Mitra *et al*. (2006), Maji and Pal (2008) and Maji and Paul (2012). In the other approach an interesting issue is related to the detection of outliers through RKM using entropy computation to measure similarity among cluster (Setyohadi *et al*., 2014). For recent surveys on rough clustering and the relationship into further soft clustering approaches refer to Peters

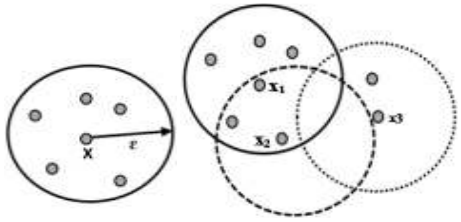
et al. (2013). The focus of this study is on existing (πRKM) (Peters and Lingras, 2014; Peters, 2014), where RKM algorithm is upgraded to become a more robust version. Hence, the existing version of RKM algorithm can be found in a recent publication (Peters, 2015b).

PROPOSED METHOD

The RKM process is much closer to the statistical K-Means (Peters *et al*., 2013) and the C-Means algorithm. The aim is to discriminate a number of overlapping objects in between positive clusters by finding accurate centroids. K-Means algorithm is more sensitive to outliers (Jain *et al*., 2000; Velmurugan and Santhanam, 2010). An outlier is an "observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism" (Hawkins, 1980). In this study, we propose a weight to minimize the effect of outlier objects on means function of RKM algorithm as our contribution. In particular, we separate a number of objects (outliers) using LOF method which was introduced by Breunig *et al*. (2000) and then minimize its effect, where it appears in the positive region of each cluster.

Local Outlier Factor (LOF): LOF is a ratio which estimates reachability density of the area around the object to the local densities of its neighbors. The successful method has widely been used to detect outliers and it doesn't suffer local density problem. Additionally, the method is a single-link which is commonly used with a hierarchical clustering algorithm known as OPTICS (Ordering Points to Identify the Clustering Structure) (Breunig *et al*., 1999). OPTICS is an extension of DBScan (Density-Based Spatial clustering of applications with noise) used in hierarchical clustering (Ankerst *et al*., 1999). The advantage of using OPTICS is it is less sensitive for use in parameter setting and finding the clustering structure. The Local Outlier Factor (LOF) requires observing some definition as proposed by Breunig *et al*. (2000). The definition consists of three steps as follows:

Step 1: Determine the neighborhood: LOF defines the neighborhood border distance d(X, k_{th}) from each X object to its k_{th} nearest neighbor by using similarity distance. A simple distance measure like Euclidean distance can often be used to reflect the difference between two objects. However, other distance metrics such as Manhattan distance or Chebyshev distance can also be used. For instance, suppose there are three objects (x_1, x_2 and x_3). The x_2 is 1 distance unit from $x_1$and x_3 is 2 distance units from x_1. Therefore, x_2 is the nearest neighbor to x_1 and x_3 is the second nearest Neighbor to x_1. The formula for calculating the distance of the k_{th} nearest neighbor to object (x_1) is described as follows:



(a) (b)

Fig. 4: Example of LOF definition

$$K_{th}d(x_1) = \{x_2, x_3 \in D \setminus d(x_1, x_2) \leq d(x_1, x_3)\} \quad (5)$$

Step 2: Determine the local reachability distance: Reachability Distance can be determined based on two parameters:

- A parameter MinPts specifying a minimum number of objects
- A parameter radius (ε) specifies a volume

For example, suppose there are 5 nearest neighbors (*MinPts* = 5) of object X (Fig. 4a) exceed the radius (ε = 0.3) threshold (called core Distance). Moreover, we call x_1 Core-Distance Object if all detected neighbors are too close. In the other case, an object x_2 (Fig. 4b) is called reachability Object, where the 2 neighbors exceed the radius threshold.

In this case, the reachability distance of object X with respect to k_{th} objects (X') is defined as:

$$\text{reach\_dist}\,k_{th}(X, X') = \max\left\{k_{th}\text{-dis}(X), d(X, X')\right\} \quad (6)$$

In summary, the local reachability density of an object X is the inverse of the average reachability distance based on the *MinPts* of X'. The local reachability density of X is defined as follows:

$$\text{lrd}_{MinPts}(X) = 1 / \left\{\frac{\sum_{X' \in_{MinPts}(X)} \text{reach-dist}_{MinPts}(X, X')}{N_{MinPts}(X)}\right\} \quad (7)$$

Note that the local density can be Undefined if all the reachability neighbors are present. Also, the local density can be infinite (∞) if all the reachability distances in the summation are 0. This may occur for an object X if there are at least *MinPts* objects, different from X', but sharing the same spatial coordinates, i.e., if there are at least *MinPts* duplicates of X' in the dataset.

Step 3: Compute LOF: The outlier factor of object X is the average of the ratio of the local reachability density of X and those of X's*MinPts*-nearest neighbors. The Local Outlier Factor of X is defined as follows:

$$LOF_{MinPts}(X) = \frac{\sum_{X' \in_{MinPts}(X)} \frac{\text{lrd}_{MinPts}(X)}{\text{lrd}_{MinPts}(X')}}{\left|N_{MinPts}(X')\right|} \quad (8)$$

In conclusion, the results showed that the higher the LOF value of X becomes when the lower X local reachability density is, then the higher the local reachability densities of X' nearest neighbors become. The LOF computation procedure is presented as given below:

Input: MinPts, ε.
Output: LOF Score.

Step 0: Calculate the k_{th}-distance of each object in the dataset as given in Eq. (5).

Step 1: Determine the local Reachability Distance for each object in the dataset as in Eq. (6) and (7).

Step 2: Calculate the LOF as in Eq. (8).

Minimize the effect of local outlier objects on the means function: As already mentioned, the effect of each object on the means function would decrease based on the number of belongs regions. More specifically, the weight for each object in the boundary region is used to reduce its effect on calculating the means. Hence the weight would be equal or less than 0.5(w< = 0.5), depending on the number of upper regions that object belongs to. On the other hand, the weight of each object in the lower region is 1(w = 1), where no other regions belong to. However, taking the means of partition cluster may also have the effect of local outlying nature on the object. In this study, a proposed weight (w) (0.5<w<1) is concerned with the object/objects in the lower region, where the degree of each object being outlaid is provided. In summary, the effect of each object on the lower region would decrease only if it exceeds the outlying threshold (*LOFT*).

Furthermore, the proposed weight (w) is defined in the means function as follows:

$$M_k = \begin{bmatrix} M_k = \frac{\left[\sum_{(i=1)}^{n} w * \frac{X_i}{R\hat{}(X_i)}\right]}{\left[\sum_{(i=1)}^{n} w * \frac{1}{R\hat{}(X_i)}\right]} \\ \left[for\ (LOF\ (Xi) > LOFT\ \&\ R\hat{}(X_i) = 1), w = (1 < w < 0.5)\right] \\ w = 1\ Otherwise \end{bmatrix} \quad (9)$$

The proposed algorithm is described as below:

Input: K Numbers, w, T, *LOFT*.
Output: rough Clusters.
BB Initialization.

- Determine the initial means (max distance where LOF<= LOFT).
- Assign each object X to the corresponding upper approximation of its nearest centroid.

Step 1: Compute the new means as Eq. (9).

Step 2: Assign into approximations space:

- Determine the nearest Centroid as shown in Eq. (2):
- Determine if further data object is also close to other centroids or not by using relative distance and threshold as defined in Eq. (3)
- If T' ≠ ∅ then at least one other centroid is similarly close to the object.
- If T' =∅ then no other centroids are similarly close to the object.

Step 3: Check convergence of the algorithm.

- If the algorithm has not converged continue to Step 1.
- Else STOP.

Despite the fact that LOF method is a useful one, the computation of the LOF value of each data object requires a lot of MinPts nearest neighbor queries. This makes each calculation of LOF a costly operation. However, in this study, LOF calculation does not affect the calculation process of RKM algorithm. At the same time, it offers more benefits when LOF is applied on constrained RKM algorithm. In addition, applying LOF to RKM addresses the issue of algorithm sensitivity to initial centroids as well as reducing the algorithm run time.

EXPERIMENTAL EVALUATION

Three experiments were conducted in our laboratory lab. The first experiment is based on a synthetic dataset and the rest are applied to Iris and Vowel datasets taken from the UCI Machine Learning Repository. The results of Iris and Vowel datasets were examined by comparing between proposed weight to Hard K-Means and πRKM.

Furthermore, the experimental results are evaluated based on a rough classifier validity index introduced in Peters (2015a). The rough classifier which is a simple and effective validation index can be applied as external criteria when labeled data are given. In addition, sufficient description on the rough classifier index is provided in Peters (2015a, 2015b). Besides, the paper provides the calculation of the returns of the obtained results when correctly clustered objects deliver positive returns (gains) and incorrectly clustered objects negative returns (penalties). A basic notation to assess the classifier quality index and returns penalty are described in Table 1.

Synthetic dataset: A sample data (15 Objects) are presented as follows:

$$Xn = \begin{bmatrix} 0.1 & 0.2 & 0.0 & 0.0 & 0.1 & 0.2 & 0.3 & 0.4 & 0.5 & 0.7 & 0.07 & 0.8 & 0.8 & 0.9 & 1.0 \\ 0.1 & 0.0 & 0.1 & 0.2 & 0.3 & 0.3 & 0.2 & 0.4 & 0.5 & 0.6 & 0.8 & 0.8 & 0.7 & 0.6 & 0.9 \end{bmatrix}$$

Table 1: Definition of important symbols in rough classifier validity index

Notation	Description
✔	The number of correctly classified objects derived from the objects assigned to lower approximations.
✘	Number of incorrectly classified objects.
QI1	Quality index of the objects in lower approximations.
QI2	Represents a conservative assessment strategy since it puts the objects in the lower approximations in relation to all objects.
QI5	Unweighted boundary objects.
QI6	π-weighted boundary objects.
ρ	Consider any deviation from this as slack and indicates how strongly boundaries are populated by objects.
ψ	A penalty factor.

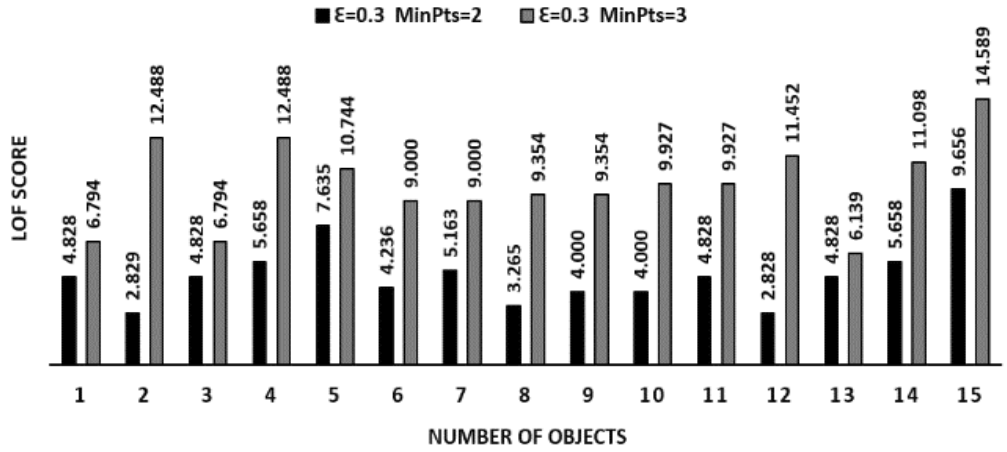


Fig. 5: Score of 15 objects

For ease of understanding, suppose the parameters *MinPts* = 2 and ε = 0.3. The LOF ratio for each object can be up or down (Fig. 5) and it is based on how the object is isolated from its neighbors. Similarly, among the 3-nearest neighbors (*MinPts* = 3) and ε = 0.3, the value of LOF score is also marked in Fig. 5.

In contrast, Fig. 6 shows the Data clusters based on first possible inputs of *LOFT*> = 4 and w = 0.7. The means become more accurate when the effect of local outliers is minimized. In addition, the means are depicted as M_k = (0.1971, 0.2188), M_k = (0.6847, 0.6361).

Iris plant dataset: Iris dataset is a real world dataset (Anderson, 1935). The available dataset has 150 random samples of flowers and three types of classes which are *Setosa*, *Versicolor* and *Virginica*. In fact, the nature of the dataset shows that the first class is very easy to separate from the two other classes. LOF scores as visualized in the Fig. 7 to 9 is based on the inputs

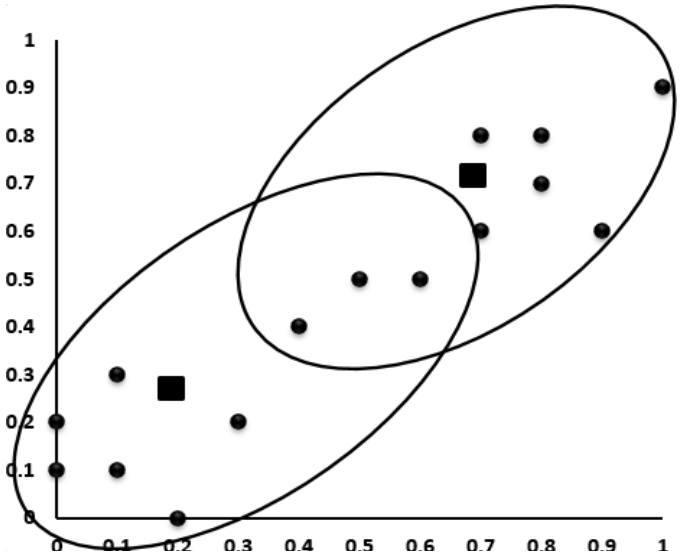


Fig. 6: Two clusters based on proposed weight

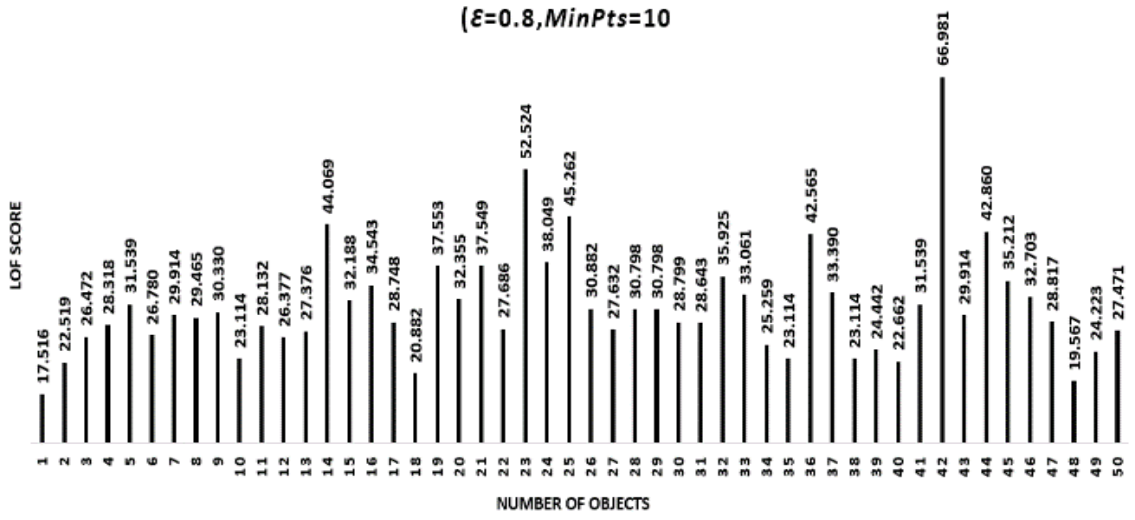


Fig. 7: LOF score for *setosa* type

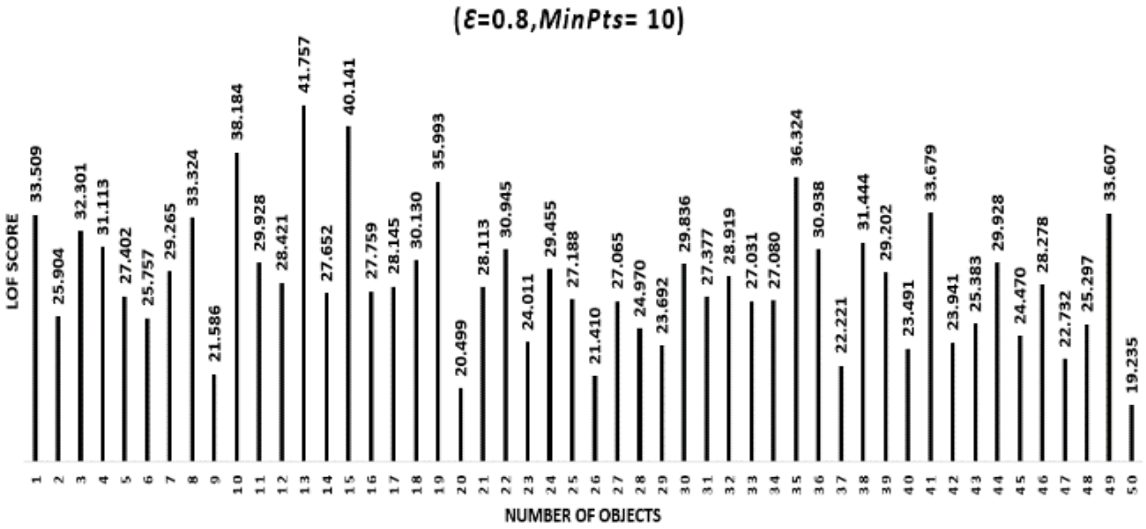


Fig. 8: LOF score (*Versicolor* type)

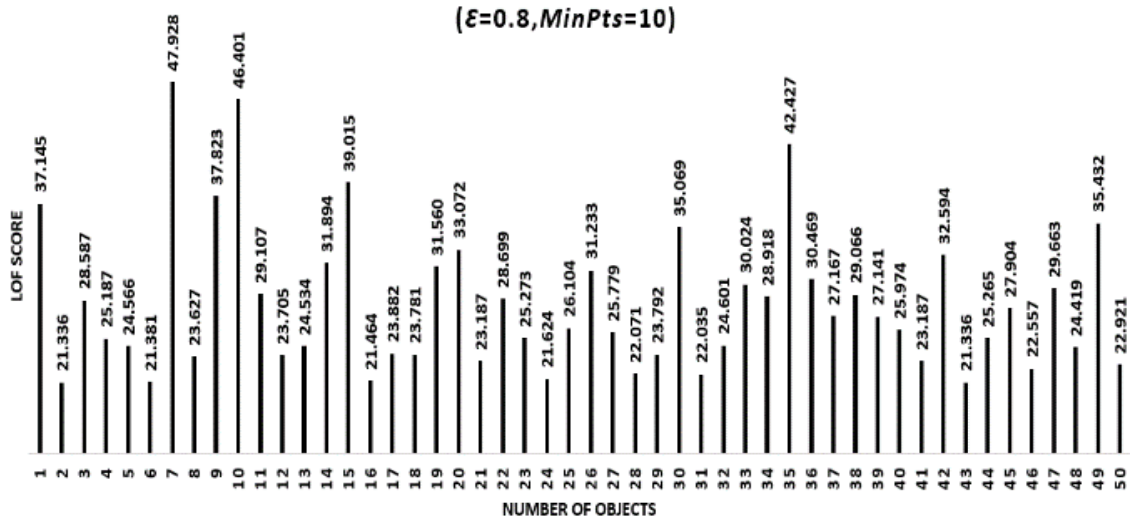


Fig. 9: LOF score (*virginica* type)

Table 2: Summary of quality indices: The iris and the Vowel data sets

Dataset	Algorithm	index	✔	✘	∑(✔+✘)	ρ=1-∑(✔+✘)
Iris						
	Hard K-Means					
		Cluster	132	18	150	-
		QI0	0.88	0.12	1	-
Proposed *w* to Hard K-Means						
	Cluster		136	14	150	-
		QI0	0.91	0.09	1	-
	Propose *w* to π RKM					
		Cluster	132	11	143	
		Lower approximation	7	0	7	-
		Boundary:unweighted	3.5	0	3.5	-
		QI1	0.9231	0.0769	1	0.0000
		QI2	0.8800	0.0733	0.9533	0.0467
		QI5	0.9267	0.7333	1	0.0000
		QI6	0.9099	0.7333	0.9833	0.0167
		QI1				
Vowel						
	Hard K-Means					
		Cluster	458	413	871	-
		QI1	0.53	0.47	1	-
Proposed *w* to Hard K-Means						
		Cluster	473	398	871	-
		QI1	0.5431	0.4569	1	-
		Cluster	407	256	663	-
		Lower approximation	143	65	208	-
		Boundary:unweighted	66.356	29.67	94.35	-
		QI1	0.6139	0.3861	1	0.0000
		QI2	0.4673	0.2939	0.7612	0.2389
		QI5	0.6315	0.3854	1	0.0000
		QI6	0.5383	0.3327	0.8709	0.1291

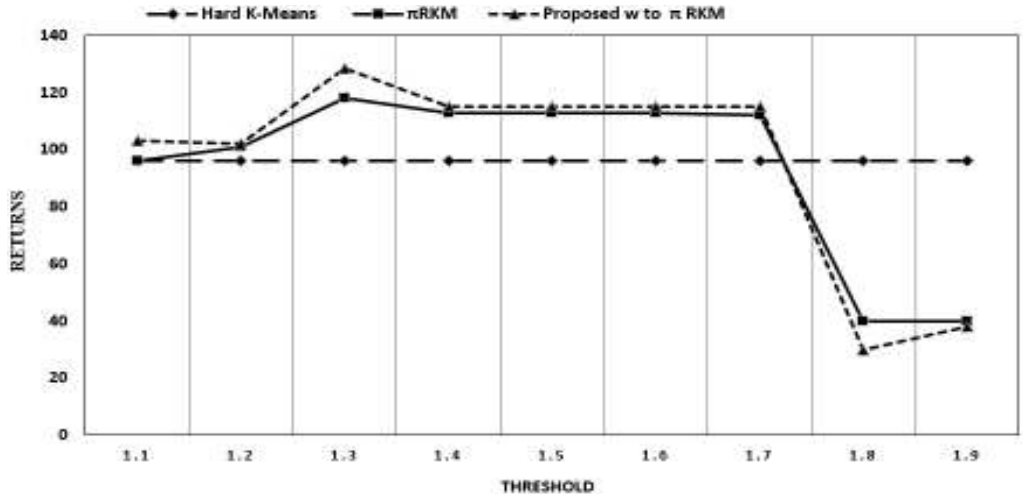


Fig. 10: Returns for the iris data (penalty $\psi = 2.0$)

MinPts = 10, $\varepsilon = 0.8$. The maximum value is 69.6 and minimum ratio is 21.4. Additionally, *setosa* has a ratio that is in between 17.516 to 66.981, *versicolor* has a ratio that is in between 19.235 to 41.757 and that of *virginica* is between 21.336 to 47.928.

Table 2 shows the different results between Hard K-Means, proposed weight to K-Means and proposed Weight to πRKM. Improved results are observed when *LOFT* = 33.4, *T* = 1.3 and the *w* = 0.7.

Vowel data: The Vowel data consists of a set of 871 Indian Telugu vowel sounds (Pal and Majumder, 1977), uttered by three male speakers in the age group of 30-35 in a Consonant-Vowel-Consonant context. The three features correspond to the first, second and third vowel format frequencies obtained through spectrum analysis of the speech data. For LOF method, we applied parameters (*Minpts* = 30, $\varepsilon = 0.25$) and range of LOF score are in between (2.10 to 13.45). The improved

Table 3: Returns (Iris and Vowel data)

ψ	A1. Hard K-Means	A2. Proposed w to Hard K-Means	A3. Proposed w to π RKM	(A2-A1) $\Delta 1$	(A3-A1) $\Delta 2$
Iris data set					
	P = 132, N = 18	P = 136, N = 14	P = 132, N = 7		
0.5	123	129	128.5	6	5.5
2.0	96	108	118	12	22
5.0	42	66	97	24	55
Vowel data set					
	P = 458, N = 413	P = 136, N = 14	P = 132, N = 7		
0.5	251.5	274	279	22.5	27.5
2.0	-368	-323	-105	45	263
5.0	-1607	-1517	-873	90	734

results are seen in Table 2 with parameters *LOFT* = 5, *T* = 1.5, alongside the proposed w = 0.7.

RESULTS AND DISCUSSION

The results in Table 2 shows that the proposed weight improves the number of correct objects in positive clusters, while the number of incorrect object reduced. *QI6* is considered as the most adequate for assessing the quality of rough clustering results. Our proposed weight, improved 2.66% of iris dataset and 0.65% for Vowel dataset in comparison with results indicated in the related paper (Peters, 2015b). Also, the results in Table 3 shows the slacks used in indicating the proportion of objects that are neglected in the numerators in relation to the denominators (Peters, 2015b) for more details). Based on the results, the penalties of proposed weight obviously decreased in comparison to proposed Weight to Hard K-means. Figure 10 shows the returns obtained from the Iris data for a penalty of $\psi = 2.0$, where the range of $1.1 \leq T \leq 1.8$ than the returns obtained by Hard k-means, πRKM and proposed weight to πRKM.

CONCLUSION AND RECOMMENDATIONS

Rough clustering is an effective alternative to hard clustering. RKM algorithm is conducted based on adopting the interpretation of rough set properties through applying traditional K-Means algorithm. This successful idea has received acceptance in many application domains with versions upgrade. Recently, a newly proposed method using Laplace's principle of indifference has been applied to the means function of RKM algorithm. However, the implemented results by the authors in Peters (2015b) indicated that the RKM algorithm still requires more attention. One reason is the number of incorrectly clustered objects. In this study, we attempt to find a solution to obtain a high number of correctly clustered objects. Therefore, we proposed a weight to minimize the effect of a local outlier on mean function. Furthermore, the LOF method was formulated to be used in measuring the objects in the dataset. The results are provided based on synthetic and real datasets. The inclusion of proposed weight to RKM provides convincing results. Moreover, the improved solution increased the number of correctly objects in the clusters and as well decreased the number of incorrectly objects in clusters. In future work, the use of the algorithm in real life application domain will be employed.

REFERENCES

Anderson, E., 1935. The irises of the Gaspé Peninsula. B. Am. Iris Soc., 59(1): 2-5.

Ankerst, M., M.M. Breunig, H.P. Kriegel and J. Sander, 1999. OPTICS: Ordering points to identify the clustering structure. ACM SIGMOD Record, 28(2): 49-60.

Bezdek, J.C. and J.D. Harris, 1978. Fuzzy partitions and relations; an axiomatic basis for clustering. Fuzzy Set. Syst., 1(2): 111-127.

Breunig, M.M., H.P. Kriegel, R.T. Ng and J. Sander, 1999. Optics-of: Identifying local outliers. In: Żytkow, J.M. and J. Rauch (Eds.), Principles of Data Mining and Knowledge Discovery. PKDD, 1999. LNCS, Springer Verlag, Berlin, Heidelberg, 1704: 262-270.

Breunig, M.M., H.P. Kriegel, R.T. Ng and J. Sander, 2000. LOF: Identifying density-based local outliers. Proceeding of the ACM Sigmod International Conference on Management of Data, pp: 93-104.

Hartigan, J.A. and M.A. Wong, 1979. Algorithm AS 136: Algorithm AS 136: A K-means clustering algorithm. J. Roy. Stat. Soc. C, 28(1): 100-108.

Hawkins, D.M., 1980. Identification of Outliers. Chapman and Hall, London, Vol. 11.

Jain, A.K., 2010. Data clustering: 50 years beyond K-means. Pattern Recogn. Lett., 31(8): 651-666.

Jain, A.K., M.N. Murty and P.J. Flynn, 2000. Data clustering: A review. ACM Comput. Surv., 31(3): 264-323.

Krishnapuram, R. and J.M. Keller, 1993. A possibilistic approach to clustering. IEEE T. Fuzzy Syst., 1(2): 98-110.

Laplace, P., 1998. Philosophical Essay on Probabilities. Translated from the Fifth French Edition of 1825, Springer, Berlin.

Lingras, P., 2009. Evolutionary rough K-means clustering. In: Wen, P., Y. Li, L. Polkowski, Y. Yao, S. Tsumoto and G. Wang (Eds.), Rough Sets and Knowledge Technology. RSKT 2009. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 5589: 68-75.

Lingras, P. and C. West, 2004. Interval set clustering of web users with rough K-means. J. Intell. Inf. Syst., 23(1): 5-16.

Lingras, P. and G. Peters, 2011. Rough clustering. Data Min. Knowl. Disc., 1(1): 64-72.

Maji, P. and S.K. Pal, 2008. RFCM: A hybrid clustering algorithm using rough and fuzzy sets. Fund. Inform., 80(4): 475-496.

Maji, P. and S. Paul, 2012. Rough-Fuzzy C-Means for Clustering Microarray Gene Expression Data. In: Kundu, M.K., S. Mitra, D. Mazumdar and S.K. Pal (Eds.), Perception and Machine Intelligence. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 7143: 203-210.

Mitra, S., 2004. An evolutionary rough partitive clustering. Pattern Recogn. Lett., 25(12): 1439-1449.

Mitra, S., H. Banka and W. Pedrycz, 2006. Rough-fuzzy collaborative clustering. IEEE T. Syst. Man Cy. B, 36(4): 795-805.

Pal, S.K. and D.D. Majumder, 1977. Fuzzy sets and decisionmaking approaches in vowel and speaker recognition. IEEE T. Syst. Man Cyb., 7(8): 625-629.

Pawlak, Z., 1982. Rough sets. Int. J. Comput. Inf. Sci., 11(5): 341-356.

Peters, G., 2006. Some refinements of rough k-means clustering. Pattern Recogn., 39(8): 1481-1491.

Peters, G., 2012. Rough Sets: Selected Methods and Applications in Management and Engineering. Springer, London, New York.

Peters, G., 2014. Rough clustering utilizing the principle of indifference. Inform. Sciences, 277: 358-374.

Peters, G., 2015a. Assessing rough classifiers. Fund. Inform., 137(4): 493-515.

Peters, G., 2015b. Is there any need for rough clustering? Pattern Recogn. Lett., 53: 31-37.

Peters, G. and P. Lingras, 2014. Analysis of User-Weighted π Rough k-Means. In: Miao, D., W. Pedrycz, D. Ślęzak, G. Peters, Q. Hu and R. Wang (Eds.), Rough Sets and Knowledge Technology. RSKT, 2014. Lecture Notes in Computer Science, Springer, Cham, 8818: 547-556.

Peters, G., M. Lampart and R. Weber, 2008. Evolutionary Rough K-Medoid Clustering. In: Peters, J.F. and A. Skowron (Eds.), Transactions on Rough Sets VIII. Lecture Notes in Computer Science, Springer-Verlag, Berlin, Heidelberg, 5084: 289-306.

Peters, G., F. Crespo, P. Lingras and R. Weber, 2013. Soft clustering - Fuzzy and rough approaches and their extensions and derivatives. Int. J. Approx. Reason., 54(2): 307-322.

Setyohadi, D.B., A.A. Bakar and Z.A. Othman, 2014. Rough K-means outlier factor based on entropy computation. Res. J. Appl. Sci. Eng. Technol., 8(3): 398-409.

Velmurugan, T. and T. Santhanam, 2010. Computational complexity between K-means and K-medoids clustering algorithms for normal and uniform distributions of data points. J. Comput. Sci., 6(3): 363-368.

Xiao, Y. and J. Yu, 2012. Partitive clustering (K-means family). Data Min. Knowl. Disc., 2(3): 209-225.