ISSN: 0975-766X
CODEN: IJPTFI
Available Online through *Research Article*
www.ijptonline.com

A ROUGH SET BASED KERNEL POSSIBILISTIC C-MEANS ALGORITHM FOR MEDICAL DATA

Utkarsha Tiwari*, Prof. Anbarasi M, M. A. Saleemdurai
SCOPE, VIT University, Vellore, Tamil Nadu, India
Email: utkarsha.0913@gmail.com

Received on 06-08-2016 *Accepted on 27-08-2016*

Abstract

The Possibilistic C-Means (PCM) has many advantages over Fuzzy C-Means (FCM) in loud situations, it has a great tendency to produce similar clusters. The fuzzy set handles overlapping partitions, the rough set handles instability, dubiousness, fragmentation, and incoherence in a class. The kernel handles linear separability of the clusters. In this paper, RKPCM algorithm is proposed.

Keywords: PCM, fuzzy set, kernels,rough set, RKPCM.

I. Introduction

Clustering helps to differentiate the dataset of similar data points in a better way by using various techniques. Clustering techniques have effectively been connected to design acknowledgment, machine learning, science, drug, PC vision, interchanges, remote detecting, etc. In c-means clustering the cluster centers are used to form the clusters. The fuzzy c-means uses the concept of fuzziness in order to handle overlapping clusters. Therefore, fuzzy property helps in recognizing information designs which could have a place with one or more groups with a specific level of enrollments. [1]

Rough sets [2], [3] deal with vulnerability, unclearness, deficiency, and ambiguity in the datasets [8]. It has been widely used with fuzzy-rule extraction, reasoning with uncertainty, fuzzy modeling, etc. [3]. Lingras and West built up another bunching strategy called rough c- means (RCM) [4], which explains a cluster by a pair of lower and upper approximations which are weighted contrastingly utilizing couple of parameters which are helpful in registering the new focuses. Amalgamation of fuzzy and rough sets gives an outlook change in managing prevailing upon instability. KFCM by supplanting the internal item with a proper "piece" capacity, naturally performs a nonlinear mapping to a high dimensional element space without expanding the quantity of parameters. This 'kernel method' has been successfully connected into numerous learning frameworks[8]. The integration of the kernel s [5], [6], [7] (with rough

fuzzy c-means) permits the mapping of the first component space into a higher dimensional space along these lines permitting straight partition of clusters. On the other hand, the FCM has a constraint that the enrollments of an information point crosswise over classes are equivalent to solidarity. Despite the fact that this is helpful in making segments, the enrollments don't generally contrast with the instinctual thought of level of belongingness or likeness. Adding on, the FCM is more sensitive to noise. To overcome this problem, Krishnapuram and Keller had proposed the possibilistic c-means (PCM) algorithm [9]. The advantages of PCM are that it conquers the need to indicate the quantity of groups and is very vigorous in an uproarious domain. At the point when the information is excessively loud, the circumstance is entirely diverse in light of the fact that the FCM is severely sensitive to outliers. This paper proposes a rough kernel possibilistic c-means (RKPCM) algorithm to overcome weakness of PCM.

II. Kernel Fuzzy C-Means Clustering

Given a dataset, $A=\{a,.....,a\} \subset R^p$, the original FCM algorithm partitions *A* into *c* fuzzy subsets by minimizing the following objective function

$$J(U,V)=\sum\sum u^m \|a-v\|^2 . \qquad (1)$$

where c is the number of bunches and chose as a predefined esteem, *n is* the number of data points, u_{ik} *is* the membership of a_k in class *i*, satisfying $\sum^c u_{ik}=1$, *m is* the quantity controlling clustering fuzziness, and *V as* the set of cluster centers or prototypes ($v \in R^p$).

Presently consider the proposed kernel fuzzy c-means (KFCM) algorithm. Characterize a nonlinear guide as $\Phi: x \rightarrow \Phi(x) \in F$, where $x \in X$. *X* denotes the data space, and *F* the changed element space with higher even unbounded measurement. KFCM minimizes the accompanying target capacity

$$J(U,V)=\sum\sum u^m \|\Phi(x)-\Phi(v)\|^2 (2)$$

Where

$$\|\Phi(x)-\Phi(v)\|^2=K(x,x)+K(v,v)-2K(x,v). \qquad (3)$$

In the event that we receive the Gaussian function as a kernel function, i.e., $K(x,y)=\exp(-\|x-y\|^2/\sigma^2)$, then $K(x,x)=1$, according to Eq. (3), Eq. (2) can be rewritten as

$$J(U,V)=2\sum\sum u^m (1-K(x,v)) . \qquad (4)$$

Minimizing Eq. (4) under the constraint of u_{ik}, we have

$$u_{ik} = \frac{(1/(1-K(x_k, v_i)))^{1/(m-1)}}{\sum_{j=1}^{c}(1/(1-K(x_k, v_j)))^{1/(m-1)}} \quad (5)$$

$$v_i = \frac{\sum_{k=1}^{n} u_{ik}^m K(x_k, v_i) x_k}{\sum_{k=1}^{n} u_{ik}^m K(x_k, v_i)} \quad (6)$$

KFCM ALGORITHM

Step 1: Fix *c*, t_{max}, $m>1$ and $\varepsilon>0$ for some positive constant;

Step 2: Initialize the memberships u^i.

Step 3: For t=1,2…. t_{max}, do:

(a) Update all prototypes v_i^t with Eqs. (6);

(b) Update all memberships uwith Eqs. (5);

(c) Compute $E^t = \max(u^t - u^{t-1})$, if $E^t \leq \varepsilon$,

Stop; else t=t+1;

III. Kernel Possiblistic C-Means Clustering

The first FCM utilizes the probabilistic imperative that the enrollments of an information indicate crosswise over classes entirety one. While this is helpful in making segments, the participations coming about because of FCM and its subsidiaries, regardless, don't by and large identify with the characteristic thought of level of having a spot or likeness. Krishnapuram and Keller unwind this imperative and propose a possibilistic way to deal with grouping (PCM) by minimizing the accompanying item work

$$J_m(U,V) = \sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^m \| x_k - v_i \|^2 + \sum_{i=1}^{c} \eta_i \sum_{k=1}^{n} (1-u_{ik})^m \quad (7)$$

where $\eta_i$are suitable positive numbers. The principal term requests that the separations from information focuses to the models be as low as could be expected under the circumstances,whereas the second term forces the $u_{ik}$to be as far reaching as could sensibly be normal, in this manner evading the trifling arrangement. As in KFCM, we adopt the Gaussian kernel function. Then the updating of memberships is

$$u_{ik} = \frac{1}{1+(2(1-K(x_k, v_i))/\eta_i)^{1/(m-1)}} \quad (8)$$

Here we use the Gaussian function as the kernel function, and $\eta_i$are estimated using

$$\eta_i = K \frac{\sum_{k=1}^{n} u_{ik}^m 2(1 - K(x_k, v_i))}{\sum_{k=1}^{n} u_{ik}^m} \quad (9)$$

Typically, *K* is chosen to be 1.

KPCM ALGORITHM

Step 1: Fix *c*, $t_{max,}$ m>1 and ε>0 for some positive constant;

Step 2: Initialize the memberships u^0 using KFCM algorithm.

Step 3: Estimate $\eta_i$using equation (9).

Step 4: For t=1,2…. t_{max}, do

(a) Update all prototypes v_i^t with Eqs. (6);

(b) Update all memberships $u_{ik}$with Eqs. (8);

(c) Compute E^t =max(u^t-u^{t-1}), if $E^t \leq \varepsilon$,

Stop; else t=t+1;

IV. Rough Kernel Possibilistic C-Means Clustering

Kernel methodologies can enhance the execution of traditional bunching or order calculations for complex appropriated information. This is achieved by using a kernel function, which is characterized as the inward result of two qualities got by a change capacity. In doing as such, this permits calculations to work in a higher dimensional space (i.e., more degrees of opportunity for information to be seriously parceled) without computing the change. KPCM incorporates a difference redesigning strategy for Gaussian portions for every grouping cycle. A few exploratory results demonstrate that the proposed calculation can outflank different calculations for general information with added substance clamor.

In the proposed algorithm, each cluster is treated as rough set. The rough cluster has lower and upper approximations. To stay away from the instatement affectability and low computational proficiency issues of the kernelized possibilistic c-means clustering algorithm (KPCM), hybridization of rough set is applied and RKPCM is proposed.

The algorithm forms three types of clusters where data patterns may lie in any of the following:

- In both upper and lower approximations which can be denoted as

$\underline{B}X_i \neq \emptyset \wedge BX_i - \underline{B}X_i \neq \emptyset$ by using the formula -

$$w_{low}\frac{\sum_{\vec{x}_k\in\underline{B}X_i}\vec{x}_k\mu_{ik}^m K(\vec{x}_k,\vec{v}_i)}{\sum_{\vec{x}_k\in|\underline{B}X_i|}\mu_{ik}^m K(\vec{x}_k,\vec{v}_i)}+w_{up}\frac{\sum_{\vec{x}_k\in\overline{B}U_i}\vec{x}_k\mu_{ik}^m K(\vec{x}_k,\vec{v}_i)}{\sum_{\vec{x}_k\in|\overline{B}U_i|}\mu_{ik}^m K(\vec{x}_k,\vec{v}_i)} \quad (10)$$

Only in upper approximation which can be denoted by

$$\underline{B}X_i=\emptyset\wedge\overline{B}X_i-\underline{B}X_i\neq\emptyset$$

by using the formula –

$$\frac{\sum_{\vec{x}_k\in\overline{B}X_i-\underline{B}X_i}\vec{x}_k\mu_{ik}^m K(\vec{x}_k,\vec{v}_i)}{\sum_{\vec{x}_k\in\overline{B}U_i-\underline{B}X_i}\mu_{ik}^m K(\vec{x}_k,\vec{v}_i)} \quad (11)$$

Only in lower approximation by using the formula-

$$\frac{\sum_{\vec{x}_k\in\underline{B}X_i}\vec{x}_k\mu_{ik}^m K(\vec{x}_k,\vec{v}_i)}{\sum_{\vec{x}_k\in\underline{B}X_i}\mu_{ik}^m K(\vec{x}_k,\vec{v}_i)} \quad (12)$$

The patterns for lower approximation is controlled by w_low while for upper approximations is controlled by w_up. $\overline{B}X_i-\underline{B}X_i$ is the set of patterns lying in rough boundary between the two approximations. When a cluster contains objects only in its lower or only in its upper approximation, the cluster center is computed in the traditional manner without scaling factor w_{low} or w_up. The rough kernel possibilistic c-means is as below-

RKPCM ALGORITHM

Step 1: Fix c, t_{max}, $m>1$ w_{low} and $\varepsilon>0$ for some positive constant;

Step 2: Initialize the memberships u^0 using KFCM algorithm.

Step 3: Estimate $\eta_i$using equation (9).

Step 4: For t=1,2…. t_{max}, do

(a) Update all prototypes v_i^t with Eqs. (6);

(b) Update all memberships u_{ik}^twithEqs. (8);

(c) Compute $E^t = \max(u^t-u^{t-1})$, if $E^t\leq\varepsilon$,

Stop; else t=t+1;

Step 5: for each, $\vec{x}_k, k=1:N$ do

Step 6: Assign maximum membership grade to x_k, $\mu_{pk}\leftarrow\max_{j=1:C}(\bar{\mu}_{jk})$

Step 7: for j=1:c, do

Step 8: if $\mu_{jk}/\mu_{pk} >$ threshold, then

Step 9: Assign x_k to upper approximation

end if

end for

Step 10: if $x_k$doesn'tbelong to upper approximation, then

Step 11: Assign $x_k$to lower approximation.

end if

end for

Step 12: Update clusters using the formulas (10), (11) and (12).

Step 13: End while.

V. Results

DB Index:

Its defined as the ratio of sum of intra-cluster distance to inter-cluster distance. The formula used is-

$$DB = \frac{1}{c}\sum_{i=1}^{c} max_{k\neq i}\left\{\frac{S(v_i)+S(v_k)}{d(v_i,v_k)}\right\} \quad for\ 1<i,k<c$$

It helps in reducing the intra-cluster distance and increasing the inter-cluster distance.

Dunn Index:

It is used to check the compactness of the clusters by using the formula-

$$Dunn = min_i\left\{min_{k\neq i}\left\{\frac{d(v_i,v_k)}{max_l S(v_l)}\right\}\right\} \quad for\ 1<k,i,l<c$$

1.) Clusters

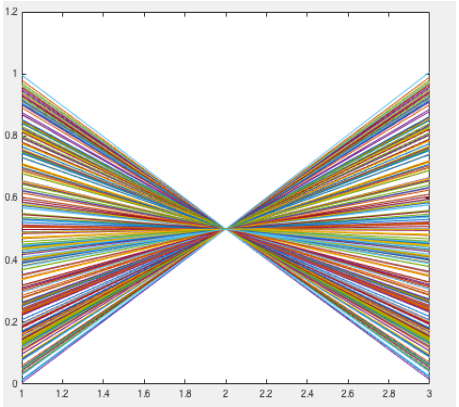

Fig.1 KFCM Cluster

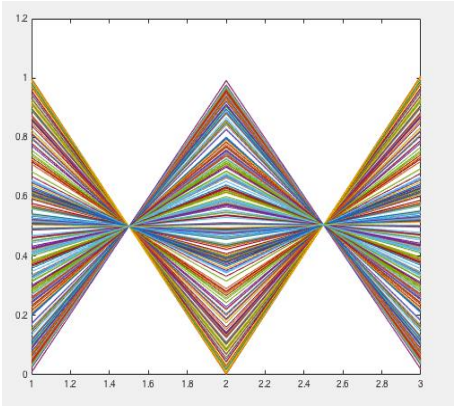

Fig.2 KPCM cluste

2.) DB and Dunn Index of KFCM and KPCM algorithms.

Table-I. DB and Dunn Index.

		DB	Dunn
Heart Disease	KFCM	0.7394	2.7049
	KPCM	0.7329	2.7291
Diabetes	KFCM	0.1902	10.5151
	KPCM	0.1958	10.2165

3.) Reduction

Table-II: Diabetes Data Reduction of RKFCM.

	Rule	LHS Support	RHS Support	RHS Accuracy	LHS Coverage	RHS Coverage	RHS Stability	LHS Length	RHS Length
1	a2(0) => dec(0) OR dec(0.1)	42	26, 16	0.619048, 0.380952	0.054759	0.052, 0.059925	1.0, 1.0	1	2
2	a0(0.1) => dec(0.1) OR dec(0)	96	42, 54	0.4375, 0.5625	0.125163	0.157303, 0.108	1.0, 1.0	1	2
3	a0(0.8) => dec(0.1) OR dec(0)	74	29, 45	0.391892, 0.608108	0.09648	0.108614, 0.09	1.0, 1.0	1	2
4	a0(0.7) => dec(0.1) OR dec(0)	73	24, 49	0.328767, 0.671233	0.095176	0.089888, 0.098	1.0, 1.0	1	2
5	a0(0.5) => dec(0) OR dec(0.1)	69	52, 17	0.753623, 0.246377	0.089961	0.104, 0.06367	1.0, 1.0	1	2
6	a0(0.4) => dec(0.1) OR dec(0)	61	24, 37	0.393443, 0.606557	0.079531	0.089888, 0.074	1.0, 1.0	1	2
7	a0(0.2) => dec(0.1) OR dec(0)	73	36, 37	0.493151, 0.506849	0.095176	0.134831, 0.074	1.0, 1.0	1	2
8	a0(0.9) => dec(0.1) OR dec(0)	77	24, 53	0.311688, 0.688312	0.100391	0.089888, 0.106	1.0, 1.0	1	2
9	a0(1) => dec(0.1) OR dec(0)	31	7, 24	0.225806, 0.774194	0.040417	0.026217, 0.048	1.0, 1.0	1	2
10	a0(0.6) => dec(0.1) OR dec(0)	86	20, 66	0.232558, 0.767442	0.112125	0.074906, 0.132	1.0, 1.0	1	2
11	a0(0.3) => dec(0) OR dec(0.1)	79	51, 28	0.64557, 0.35443	0.102999	0.102, 0.104869	1.0, 1.0	1	2
12	a0(0) AND a2(0.1) => dec(0)	6	6	1.0	0.007823	0.012	1.0	2	1

Table-III: Diabetes Data Reduction of RKPCM.

	Rule	LHS Support	RHS Support	RHS Accuracy	LHS Coverage	RHS Coverage	RHS Stability	LHS Length	RHS Length
1	a0(0.8) => dec(0) OR dec(0.1)	65	42, 23	0.646154, 0.353846	0.084746	0.084, 0.086142	1.0, 1.0	1	2
2	a0(0.7) => dec(0) OR dec(0.1)	65	39, 26	0.6, 0.4	0.084746	0.078, 0.097378	1.0, 1.0	1	2
3	a0(0.1) => dec(0.1) OR dec(0)	81	26, 55	0.320988, 0.679012	0.105606	0.097378, 0.11	1.0, 1.0	1	2
4	a0(0.4) => dec(0) OR dec(0.1)	74	43, 31	0.581081, 0.418919	0.09648	0.086, 0.116105	1.0, 1.0	1	2
5	a0(0) => dec(0.1) OR dec(0)	39	14, 25	0.358974, 0.641026	0.050847	0.052434, 0.05	1.0, 1.0	1	2
6	a0(0.5) => dec(0.1) OR dec(0)	85	39, 46	0.458824, 0.541176	0.110821	0.146067, 0.092	1.0, 1.0	1	2
7	a0(0.3) => dec(0.1) OR dec(0)	68	25, 43	0.367647, 0.632353	0.088657	0.093633, 0.086	1.0, 1.0	1	2
8	a0(0.9) => dec(0) OR dec(0.1)	79	57, 22	0.721519, 0.278481	0.102999	0.114, 0.082397	1.0, 1.0	1	2
9	a0(1) => dec(0.1) OR dec(0)	41	11, 30	0.268293, 0.731707	0.053455	0.041199, 0.06	1.0, 1.0	1	2
10	a2(0.2) => dec(0.1) OR dec(0)	82	28, 54	0.341463, 0.658537	0.10691	0.104869, 0.108	1.0, 1.0	1	2
11	a2(0.6) => dec(0.1) OR dec(0)	88	27, 61	0.306818, 0.693182	0.114733	0.101124, 0.122	1.0, 1.0	1	2
12	a0(0.2) AND a2(0.3) => dec(0)	6	6	1.0	0.007823	0.012	1.0	2	1
13	a0(0.6) AND a2(0.7) => dec(0)	9	9	1.0	0.011734	0.018	1.0	2	1

Table-IV: Heart Disease Reduction of RKFCM.

	Rule	LHS Support	RHS Support	RHS Accuracy	LHS Coverage	RHS Coverage	RHS Stability	LHS Length	RHS Length
1	a0(0.5) => dec(0) OR dec(1)	17	4, 13	0.235294, 0.764706	0.066148	0.03252, 0.097015	1.0, 1.0	1	2
2	a0(0.7) => dec(0) OR dec(1)	39	13, 26	0.333333, 0.666667	0.151751	0.105691, 0.19403	1.0, 1.0	1	2
3	a0(0.6) => dec(1) OR dec(0)	24	6, 18	0.25, 0.75	0.093385	0.044776, 0.146341	1.0, 1.0	1	2
4	a0(1) => dec(0) OR dec(1)	9	5, 4	0.555556, 0.444444	0.035019	0.04065, 0.029851	1.0, 1.0	1	2
5	a2(0.8) => dec(1) OR dec(0)	31	20, 11	0.645161, 0.354839	0.120623	0.149254, 0.089431	1.0, 1.0	1	2
6	a2(0) => dec(0) OR dec(1)	7	3, 4	0.428571, 0.571429	0.027237	0.02439, 0.029851	1.0, 1.0	1	2
7	a0(0.3) AND a2(0.3) => dec(1) OR dec(0)	23	13, 10	0.565217, 0.434783	0.089494	0.097015, 0.081301	1.0, 1.0	2	2
8	a0(0.4) AND a2(0.4) => dec(0) OR dec(1)	30	20, 10	0.666667, 0.333333	0.116732	0.162602, 0.074627	1.0, 1.0	2	2
9	a0(0.1) AND a2(0.1) => dec(0) OR dec(1)	33	12, 21	0.363636, 0.636364	0.128405	0.097561, 0.156716	1.0, 1.0	2	2
10	a0(0.9) AND a2(0.9) => dec(0) OR dec(1)	14	8, 6	0.571429, 0.428571	0.054475	0.065041, 0.044776	1.0, 1.0	2	2
11	a0(0.9) AND a2(1) => dec(1)	1	1	1.0	0.003891	0.007463	1.0	2	1
12	a0(0.2) AND a2(0.2) => dec(1) OR dec(0)	28	14, 14	0.5, 0.5	0.108949	0.104478, 0.113821	1.0, 1.0	2	2
13	a0(0.1) AND a2(0.2) => dec(0)	2	2	1.0	0.007782	0.01626	1.0	2	1
14	a0(0.2) AND a2(0.3) => dec(0)	1	1	1.0	0.003891	0.00813	1.0	2	1
15	a0(0.4) AND a2(0.5) => dec(1)	2	2	1.0	0.007782	0.014925	1.0	2	1
16	a0(0.8) AND a2(0.9) => dec(0)	1	1	1.0	0.003891	0.00813	1.0	2	1
17	a0(0) AND a2(0.1) => dec(0)	1	1	1.0	0.003891	0.00813	1.0	2	1
18	a0(0.3) AND a2(0.4) => dec(0)	1	1	1.0	0.003891	0.00813	1.0	2	1

Table-V: Heart Disease Reduction of RKPCM.

	Rule	LHS Support	RHS Support	RHS Accuracy	LHS Coverage	RHS Coverage	RHS Stability	LHS Length	RHS Length
1	a2(0.8) => dec(0) OR dec(1)	29	15, 14	0.517241, 0.482759	0.11284	0.121951, 0.104478	1.0, 1.0	1	2
2	a2(0.6) => dec(1) OR dec(0)	28	11, 17	0.392857, 0.607143	0.108949	0.08209, 0.138211	1.0, 1.0	1	2
3	a0(0.7) => dec(0) OR dec(1)	32	13, 19	0.40625, 0.59375	0.124514	0.105691, 0.141791	1.0, 1.0	1	2
4	a0(0.4) => dec(1) OR dec(0)	24	13, 11	0.541667, 0.458333	0.093385	0.097015, 0.089431	1.0, 1.0	1	2
5	a0(0.5) => dec(1) OR dec(0)	24	17, 7	0.708333, 0.291667	0.093385	0.126866, 0.056911	1.0, 1.0	1	2
6	a0(0.3) => dec(0) OR dec(1)	26	14, 12	0.538462, 0.461538	0.101167	0.113821, 0.089552	1.0, 1.0	1	2
7	a0(0.2) => dec(0) OR dec(1)	19	8, 11	0.421053, 0.578947	0.07393	0.065041, 0.08209	1.0, 1.0	1	2
8	a0(1) => dec(1) OR dec(0)	15	9, 6	0.6, 0.4	0.058366	0.067164, 0.04878	1.0, 1.0	1	2
9	a0(0) => dec(1) OR dec(0)	14	5, 9	0.357143, 0.642857	0.054475	0.037313, 0.073171	1.0, 1.0	1	2
10	a0(0.1) => dec(1) OR dec(0)	23	11, 12	0.478261, 0.521739	0.089494	0.08209, 0.097561	1.0, 1.0	1	2
11	a0(0.9) => dec(1) OR dec(0)	26	16, 10	0.615385, 0.384615	0.101167	0.119403, 0.081301	1.0, 1.0	1	2
12	a0(0.8) AND a2(0.9) => dec(0)	1	1	1.0	0.003891	0.00813	1.0	2	1
13	a0(0.6) AND a2(0.7) => dec(0)	3	3	1.0	0.011673	0.02439	1.0	2	1

Table-VI: The confusion Matrix of RKFCM and RKPCM.

CONFUSION MATRIX				
	DIABETES		HEART DISEASE	
Measure	RKFCM	RKPCM	RKFCM	RKPCM
Sensitivity	0.6543	0.7078	0.6871	0.6913
Specificity	0.3607	0.6818	0.8	0.8148
Precision	0.844	0.93	0.8211	0.8374
Negative Predictive Value	0.1648	0.2809	0.6567	0.6567
False Positive Rate	0.3182	0.6393	0.1852	0.2
False Discovery Rate	0.07	0.156	0.1626	0.1789
False Negative Rate	0.2922	0.3457	0.3087	0.3129
Accuracy	0.6076	0.804	0.7354	0.8432
F1 Score	0.0115	0.8038	0.5001	0.9481

As the images fig.1 and fig.2 show that the formation of clusters in KPCM helps to get a more classified view and it helps in detecting outliers easily which was not possible in KFCM algorithm and from Table I, it is verified that KPCM for diabetes and heart disease data sets work better in DB and Dunn indexes as their intra-cluster distance is very less and inter-cluster distance is more which makes the clusters uniquely identified. Thus KPCM is better than KFCM algorithm. After the application of reduction on KFCM and KPCM, the tables show that the reduced rules generated are less in RKPCM on comparing with RKFCM algorithm.

Table VI shows the results of the confusion matrix for diabetes and heart disease data sets. The results show that the accuracy of RKPCM is more as compared to the accuracy of RKFCM algorithm. Therefore, Rough Kernel Possibilistic C-Means has more advantages in utilizing it as a part of constant information sets a sit gives come about more exact conclusion. The hybridized algorithm Rough Kernel Possibilistic C-Means when applied on diabetes and heart disease data sets provides more accurate and uniquely identified clusters. This helps in detecting the diseases in an appropriate way. In future perspective, more hybridized algorithms can be used and can be applied on different data sets and can be checked for its validity on large and real life data sets to get more precise and accurate results which are helpful in detecting numerous real life problems in a better way.

References:

1. Theodoridis and Koutroumbas. *Pattern Recognition*. Academic Press, New York, third edition, 2006.
2. Pawlak. Rough sets. *International Journal of Computer and Information Sciences*, 11:341–356, 1982.
3. PawlakRo*ugh Sets, Theoretical Aspects of Reasoning About Data.* Kluwer, Dordrecht, Netherlands, 1991.
4. Lingras and West. Interval set clustering of web users with rough k-means. *Intelligent Information Systems*, 23(1):5–16, 2004.
5. T. Hofmann, Scholkopf, and Smola. Kernel methods in machine learning. *The Annals of Statistics*, 36(3):1171–1220, 2008
6. S. Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, USA, 1st edition, 2004.
7. Scholkopf and Smol. *Learning with Kernels*. The MIT Press,Cambridge, 2002.
8. Cristianini, Taylor. An Introduction to SVMs and otherkernel-based learning methods. Cambridge Univ. Press, 2000.
9. Krishnapuram, J.M. Keller. A possibilistic approach toclustering, IEEE Trans. Fuzzy Systems 1: 98-110, 1993.
10. Barni, V. Cappellini, Mecocci. Comments on "A possibilisticapproach to clustering". IEEE Trans. Fuzzy Systems 4:393-396, 1996.
11. Krishnapuram, Keller. The possibilistic c-means: insights and recommendations. IEEE Trans. Fuzzy Systems 4: 385-393, 1996.
12. http://onlineconfusionmatrix.com/

Corresponding Author:

Utkarsha Tiwari*,

Email: *utkarsha.0913@gmail.com*