A conditionally positive definite kernel function for possibilistic clustering

Jyotsna Nigam and Meena Tushir*

Department of Information Technology,
Maharaja Surajmal Institute of Technology,
Delhi, India
Email: jyotsnaroy81@gmail.com
Email: meenatushir@yahoo.com
*Corresponding author

Dinesh Rai

Department of Computer Science and Engineering,
Ansal University School of Engineering and Technology,
Gurgaon, India
Email: dineshrai@ansaluniveristy.edu.in

Abstract: In the past few years, the kernel-based clustering methods have overpowered the conventional clustering techniques in the field of unsupervised learning due to its strength and effectiveness to deal with nonlinearly separable data and mapping it into higher dimensional feature space by preserving the inner structure of the data. Many kernel functions exist in the literature which works effectively depending on the type of dataset to be used. In this paper, we have proposed a new log kernel function which is embedded in the unsupervised possibilistic clustering and this kernel function is not explored much in research. We have done extensive comparison of the proposed algorithm with few clustering techniques over a test suite of several synthetic and real life datasets. Based on the experimental results, we have proved that our algorithm gives better performance than the previous methods on various comparative parameters like ideal centroids, error rate, misclassification, accuracy and elapsed time.

Keywords: unsupervised learning; possibilistic c-means; conditionally positive definite function; fuzzy c-means; kernel functions.

Reference to this paper should be made as follows: Nigam, J., Tushir, M. and Rai, D. (2017) 'A conditionally positive definite kernel function for possibilistic clustering', *Int. J. Artificial Intelligence and Soft Computing*, Vol. 6, No. 1, pp.75–91.

Biographical notes: Jyotsna Nigam received her BTech in Computer Science from Kurukshetra University and MTech in Information and Technology from Guru Gobind Singh Indraprastha University, Delhi, India. She has 12 years of experience as an Assistant Professor. Her area of interests includes data mining, soft computing, computer networking and artificial intelligence.

Meena Tushir graduated in Electrical Engineering from Sir Chhotu Ram Engineering College in Murthal, Sonepat, India in 1993. She received her ME from Delhi College of Engineering in 2000. She received her PhD from Delhi University in India in 2013. Since August 2003, she has been with the

Department of Electrical at MSIT in New Delhi, India, where she is currently an Associate Professor. Her current research interests include neural networks, fuzzy logic, modelling, identification, and control of nonlinear systems.

Dinesh Rai has completed his Masters in Computer Technology from National Institute of Technology, Raipur in 2003. Before that he did MSc in Computer Science and BSc (Maths and Compter Science) from Jiwaji University Gwalior. He has been working as an Assistant Professor in Ansal University (earlier AIT) for more than nine years. His ares of interest includes computer architecture and organisation, data structures, analysis and design of algorithms, theory of computer science and distributed operating system.

This paper is a revised and expanded version of a paper entitled 'A new log based possibilistics clustering' presented at CSI-2015; 50th Global Jubilee Annual Convention on "Digital Life", New Delhi, 2–5 December 2015.

1 Introduction

Clustering, also known as unsupervised learning, is a fundamental technique of automatic classification and typological analysis of the unlabeled data objects in such a way that the objects are categorised into group of similar clusters. Clustering (Xu and Wunsch, 2005; Backer and Jain, 1981) is commonly used as one of the methodical techniques in the fields of pattern recognition, modelling of system, image segmentation and analysis, Artificial Intelligence, data mining and so on. The hard clustering methods confine each point of the dataset to exactly one cluster leading to tightly bound clusters. Compared with the concept of hard bound clustering, the notion of fuzzy clusters describe the data patterns more efficiently in which data vectors belong to more than one cluster producing overlapping clusters. This concept is depicted by the Fuzzy set theory proposed by Zadeh (1965) that invented the idea of partial membership described by a membership function. In contrast to hard clustering, fuzzy clustering (Ruspini, 1969; Dunn, 1973) has a superior clustering performances and capabilities. This primitive approach was generalised by Bezdek to fuzzy c means algorithm (Bezdek, 1981) by using a weighted exponent on the fuzzy memberships. Although FCM is a very competent clustering method, it can only be effective in clustering 'spherical' clusters. To reduce this weakness to some extent, Krishnapuram and Keller (1993) proposed possibilistic c means algorithm, PCM which uses a possibilistic membership function to describe the degree of belonging. In contrast to FCM, possibilistic approach appeared to be more resilient to noise and outliers. However PCM has a limitation of producing overlapping clusters. i.e., it depends highly on a good initialisation and it tends to produce overlapped clusters which are undesirable. To overcome the limitations imposed by FCM and PCM, many other algorithms have been proposed with certain improvements over conventional fuzzy clustering techniques (Dave, 1991; Pal et al., 1997). In past few years, kernel-based clustering has appeared as an interesting and effective alternative over fuzzy clustering. The effectiveness of kernel-based techniques comes from the notion that it efficiently transforms the nonlinearly separable data in input space to linearly separable high dimension space by replacing the inner products with positive definite kernel functions. The superiority of the kernel-based techniques that depends primarily on the type of kernel function used and its associated parameters have been proved by many researchers (Camastra and Verri, 2005;

Wu and Xie, 2003; Zhang and Chen, 2002). In literature, many different types of kernel functions exist that are data specific.

In this paper, we have proposed a Log kernel-based unsupervised possibilistic clustering with the use of Log kernel function which is an extension of the work unsupervised possibilistic clustering (Yang and Wu, 2006) proposed by Yang et al. The kernelised version of unsupervised clustering using Gaussian kernel function (Hu et al., 2012) was proposed by Hu et al. We have presented the comprehensive comparative analysis of our proposed algorithm with well-known conventional clustering techniques like FCM, PCM, KFCM, UPC, UKPC-Gaussian. The experimental results are carried out on various synthetic and real datasets and performances are evaluated on various factors like ideal centroids, misclassification, elapsed time, accuracy and error rate.

The remainder of this paper is organised as follows. Section 2 provides the literature work on the fuzzy c-means, possibilistic c-means and unsupervised possibilistic clustering. In Section 3, the kernel-based approach and kernel-based algorithms KFCM and UKPC-Gaussian are discussed. In Section 4, the proposed log kernel function and the new clustering algorithm is presented in detail. Section 5 highlights the potential of the proposed approach through various synthetic and real datasets. Concluding remarks and future scope are presented in Section 6.

2 Literature work

2.1 Possibilistic C-means clustering

PCM clustering proposed by Krishnapuram and Keller (1993) works with the viewpoint of possibility theory. According to possibility theory, the membership values can be interpreted as the degree of compatibility or possibility of the points belonging to the classes. The objective function of PCM is defined as

$$Jpcm(U,V) = \sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^{m} d_{ik}^{2} + \sum_{i=1}^{c} \gamma_i \sum_{k=1}^{n} (1-u_{ik})^{m} \tag{1}$$

where γ_i are some suitable positive numbers. The first term of the objective function imposed the distance from the data points to the cluster centres to be as low as possible, whereas the second term forces the u_{ik} be as large as possible. The typicality of data point x_k and the centre of cluster v_i can be obtained by updating the following equations:

$$u_{ik} = \frac{1}{\left(1+\frac{d_{ki}}{\gamma_i}\right)^{\frac{1}{m-1}}}, \forall i \tag{2}$$

$$v_i = \frac{\sum_{k=1}^{n} u_{ik}^{m} x_k}{\sum_{k=1}^{n} u_{ik}^{m}} \tag{3}$$

PCM does not hold the probabilistic constraint that the membership of the data point across classes sums to one. It overcomes sensitivity to noise and overcomes the need to specify number of clusters. However, there are still some disadvantages in the PCM, i.e., it depends highly on a good initialisation and it tends to produce overlapped clusters which are undesirable.

2.2 Unsupervised possibilistic c-means clustering

In 2006, Yang and Wu proposed a new possibilistic clustering algorithm called unsupervised possibilistic clustering (UPC). The objective function of UPC is an improvement to FCM algorithm with the inclusion of partition coefficient (PC) and partition entropy (PE) validity indexes to its objective function to make it robust to noise and outliers. This generalisation makes each validity index workable in both fuzzy and possibilistic clustering models.

The objective function of UPC is:

$$J_{upc}(U,V)=\sum_{i=1}^{c}\sum_{j=1}^{n}u_{ij}^{m}d_{ij}^{2}+\frac{\beta}{m^{2}\sqrt{c}}\sum_{i=1}^{c}\sum_{j=1}^{n}\left(u_{ij}^{m}\log u_{ij}^{m}-u_{ij}^{m}\right) \tag{4}$$

where m is the fuzzy factor and c is the cluster numbers. The first term is identical to FCM objective function. The second term is constructed by analogue of PE and PC validity indexes.

Parameter β is the sample covariance. That is

$$\beta=\frac{\sum_{j=1}^{n}\left\|xj-\bar{x}\right\|}{n} \tag{5}$$

Minimising the objective function with respect to u_{ij} and setting it to zero we get equation for membership value, i.e.,

$$u_{ij}=\exp\left(-\frac{m\sqrt{c}\left\|xj-v_{i}\right\|^{2}}{\beta}\right) \tag{6}$$

where $i = 1, \ldots, c, j = 1, \ldots, n$.

Minimising the objective function with respect to v_i, the equation for membership value cluster centres i.e.,

$$v_{i}=\frac{\sum_{j=1}^{n}\mu_{ij}^{m}x_{j}}{\sum_{j=1}^{n}\mu_{ij}^{m}} \tag{7}$$

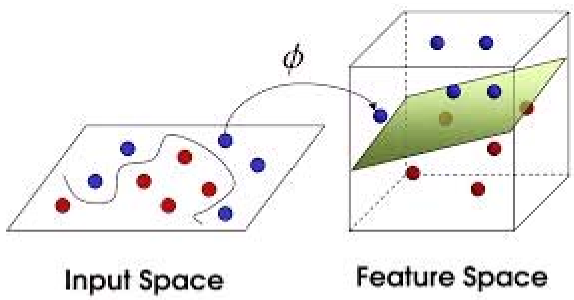
3 Kernel-based techniques

3.1 Kernel approach

Various types of kernel-based learning methods have been proposed so far by many researchers. A kernel function is generalisation of the distance matrix that measures the distance between two data points as the data points are mapped into high dimensional space in which they are more clearly separable. Kernel-based clustering methods in the higher dimension space perpetuates the internal structure of clusters in the input space.

Girolami first invented the kernel k-means (Girolami, 2002) clustering technique for unsupervised classification. Several studies (Camastra and Verri, 2005; Wu and Xie, 2003; Zhang and Chen, 2002; Dhillon et al., 2004; Ding and He, 2004) have demonstrated the superiority of kernel clustering algorithms over other techniques.

Figure 1 Mapping nonlinear data to higher dimension space by using linear hyper plane (see online version for colours)



Kernel trick:

The Kernel trick is the heart of kernel-based learning methods that provides a bridge from linearity to nonlinearity to any algorithm that can express solely in terms of dot products between two vectors. Kernel trick is interesting because that mapping does not need to be ever computed.

3.1.1 General definition of kernel function

A function $K: N \times N \rightarrow \Im\Re$ is called a positive (semi)-definite kernel if and only if it is

1 symmetric, that is, $K(x,x') = K(x',x)$ for any two objects $x, x' \in L$

2 positive (semi)-definite, that is, $c^T Kc = \sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j K(x_i, x_j) \geq 0$ with matrix K of all elements $K(x_i, x_j)$, for any $n > 0$, any choice of n objects $x_i \ldots x_n \in L$, any choice of vectors $c \in \Im\Re^n$, and any choice of numbers $c_1 \ldots c_n \in \Im\Re$ resp.

Given an unlabelled dataset $X = [x_1, x_2 \ldots\ldots\ldots x]$ in the p-dimensional space R^p, let nonlinear transformation function ϕ maps the data into higher dimension space and the explicit form of ϕ is not necessarily known.

$$\phi : R^p \rightarrow H, x \rightarrow \phi(x)$$

The dot product in the higher space can be calculated through the kernel function $K(xi, x_j)$ in the input space R^p:

$$K(x_i, x_j) = \phi(x_i).\phi(x_j)$$

Consider the following example. For $p = 2$ and a mapping function ϕ.

$$\varnothing : R^2 \rightarrow H = R^3 (x_{i1}, x_{i2}) \rightarrow \left(x_{i1}^2, x_{i2}^2, \sqrt{1 x_{i1} x_{i2}}\right)$$

Then the dot product in the feature space H is calculated as:

$$\varnothing(x_i).\varnothing(x_j) = \left(x_{i1}^2, x_{i2}^2, \sqrt{2x_{i1}x_{i2}}\right).\left(x_{j1}^2, x_{j2}^2, \sqrt{2x_{j1}x_{j2}}\right)$$
$$= \left((x_{i1}x_{i2}), (x_{j1}x_{j2})\right)^2$$
$$= (x_i, x_j)^2 = K(x_i, x_j)$$

where K-function is the dot product in the input space.

Some examples of kernel function are as follows:

Example1 (linear kernel): $k(x,y) = x^T y + c$

Example 2 (polynomial kernel): $K(x_i, x_j) = (x_i \,.\, x_j + c)^d$ where, $c \geq 0$, $d \in N$

Example 3 (Gaussian basis kernel):

$$K(x_i, x_j) : \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$$

where $\sigma > 0$.

Example 4 [hyperbolic tangent (Sigmoid) kernel]:

$$K(x, y) = \tanh\left(ax^T + c\right)$$

3.2 Kernel-based fuzzy C-means algorithm

The kernelised fuzzy c-means adopts a kernel induced metric different from the Euclidean norm in original FCM. KFCM minimises the following objective function:

$$J(U,V) = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^m \|\varnothing(x_k) - \varnothing(v_i)\|^2 \tag{8}$$

where $\|\varnothing(x_k) - \varnothing(v_i)\|^2$ is the square of the distance between $\varnothing(x_k)$ and $\varnothing(v_i)$. The distance in the feature space is calculated through the kernel in the input space as follows:

$$\|\varnothing(x_k) - \varnothing(v_i)\|^2 = (\varnothing(x_k) - \varnothing(v_i)).(\varnothing(x_k) - \varnothing(v_i))$$
$$= \varnothing(x_k).\varnothing(x_k) - 2\varnothing(x_k).\varnothing(v_i) + \varnothing(v_i).\varnothing(v_i)$$
$$= K(x_k, x_k) - 2K(x_k, v_i) + K(v_i v_i)$$

If we adopt the Gaussian kernel function as a kernel function, i.e.,

$$K(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right),$$

where σ defined as kernel width, is a positive number then $K(x, x) = 1$. Thus, equation (8) can be written as.

$$J_{kfcm}(U,V) = 2\sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^m \left(1 - K(x_k, v_i)\right) \tag{9}$$

Given a set of points X, we minimise $J_m(U, V)$ in order to determine U, V.

The necessary conditions for minimising J_{kfcm} under the constraint of U, V are:

$$u_{ik} = \frac{\left(\frac{1}{1-K(x_k, v_i)}\right)^{\frac{1}{m-1}}}{\sum_{j=1}^{c}\left(\frac{1}{1-K(x_k, v_i)}\right)^{\frac{1}{m-1}}} \tag{10}$$

$$v_i = \sum_{k=1}^{n} u_{ik}^{m}.K \frac{(x_k, v_i)\, x_k}{\sum_{k=1}^{n} u_{ik}^{m}.K(x_k, v_i)} \tag{11}$$

The common ground of kernel-based FCM is to first map the input data element into feature space with higher dimension via a nonlinear transformation and then perform FCM in feature space.

3.3 Unsupervised gaussian kernel-based possibilistic clustering

To the extension of the work proposed by Yang and Wu (2006), the Gaussian kernel-based possibilistic clustering (UKPC-G) was proposed by Hu et al. which is convenient for non-convex shaped clusters also.

The objective function of the unsupervised kernel possibilistic clustering (Hu et al., 2012) model is the following:

$$J_{UKPC}(U, V) = \sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^{m} D_{ij}^{2} + \frac{\beta}{m^2\sqrt{c}}\sum_{i=1}^{c}\sum_{j=1}^{n}\left(u_{ij}^{m}\log u_{ij}^{m} - u_{ij}^{m}\right) \tag{12}$$

where $W = [w_1, w_2, \ldots w_c]$ denote the cluster centres in the feature space and

$$D_{ij}^{2} = \left\|\Phi(x_j) - w_i\right\|_H \sqrt{\left\langle \left(\Phi(x_j) - w_{i,}\right), \left(\Phi(x_j) - w_{i,}\right)\right\rangle}$$

denotes distance from data points, which is mapped into higher dimension from x_j by mapping function using the Gaussian kernel function

$$K(x_i, x_j) : \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\sigma^2}\right)$$

where $\sigma > 0$.

By using Lagrange's multiplier and minimising the objective function, we get following equations for membership functions and the centres:

$$u_{ij} = \exp\left(\frac{m\sqrt{c}\left\|\Phi(x_j) - w_i\right\|_H^{2}}{\beta}\right), 1 \le i \le c, 1 \le j \le n \tag{13}$$

$$w_i = \frac{\sum_{j=1}^{n} u_{ij}^{m}.\Phi(xj)}{\sum_{j=1}^{n} u_{ij}^{m}}, 1 \le i \le c \tag{14}$$

In contrast to UPC, UKPC-Gaussian is more robust to noise and outliers and gives better performance for non-convex and different shaped datasets.

4 Proposed unsupervised kernel-based possibilistic clustering using log kernel function

To improve the efficiency of previously defined clustering algorithms, we have introduced a log kernel function into unsupervised clustering (UPC). Log kernel function is conditionally positive definite function i.e. it is positive for all values of greater than 0. The use of a positive definite kernel ensures that the results will be optimised in convex structure. We named our proposed algorithm as UKPC-L.

Definition of conditionally positive definite function:

Let X is a non-empty set.

A kernel K is called conditionally positive definite if and only if it is symmetric and

$$\sum\nolimits_{j,k=1}^{n} c_j c_k K\left(x_j, x_k\right) \geq 0$$

For $n \geq 1$, $c_1, \ldots, c_n \in R$ with

$$\sum\nolimits_{j=1}^{n} c = 0$$

and $x_1, \ldots, x_n \in X$.

The log kernel function that we have proposed for our algorithm is:

$$K(x, y) = \log\left(1 + \alpha \left\| x_j - v_i \right\|^2\right)$$

To the extension of unsupervised possibilistic clustering (UPC) (Yang and Wu, 2006), the objective function for UKPC-L clustering using log kernel function is as follows:

$$J_{UKPC-L} = \sum\nolimits_{i=1}^{c} \sum\nolimits_{j=1}^{n} u_{ij}^{m} D_{ij}^{2} + \frac{\beta}{m^2 \sqrt{c}} \sum\nolimits_{i=1}^{c} \sum\nolimits_{j=1}^{n} \left(u_{ij}^{m} \log u_{ij}^{m} - u_{ij}^{m}\right) \tag{15}$$

where D_{ij} denotes the distance from the data, which is mapped from x_j by φ using kernel function and is computed as follows:

$$D_{ij}^{2} = \left\| \Phi\left(x_i\right) - \Phi\left(v_i\right) \right\|^2 = -2 \log\left(1 + \alpha \left\| x_j - v_i \right\|^2\right) \tag{16}$$

The value of β (Yang and Wu, 2006) is defined by the co-variance of the data points in higher dimension space, is defined as follows:

$$\beta = \frac{-2 \sum\nolimits_{j=1}^{n} \log\left(1 + \alpha \left\| x_j - v_i \right\|^2\right)}{n} \tag{17}$$

The detailed computation steps for UKPC-L clustering are as follows:

1 Fix the number of clusters C; fix(m) >1; Set the termination parameter ε.

2 Choose the kernel function k and its parameters.

3 Initialise the cluster centroids v_i and membership degree u_{ik} of data point x_k in the i^{th} cluster $k = 1, 2, \ldots, N$ and $i = 1, 2, \ldots, C$ using fuzzy c-means clustering.

4 Compute β using

$$\beta = \frac{-2\sum_{j=1}^{n} \overline{\log}\left(1+\alpha\left\|x_j - v_i\right\|^2\right)}{n}.$$

5 Calculate the objective function $J(U, V)$ using

$$J_{UKPC-L} = \sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^m D_{ij}^2 + \frac{\beta}{m^2\sqrt{c}}\sum_{i=1}^{c}\sum_{j=1}^{n}\left(u_{ij}^m \log u_{ij}^m - u_{ij}^m\right)$$

where

$$\begin{aligned}\left\|\Phi(x_k)-\Phi(v_i)\right\|^2 &= \left(\Phi(x_k)-\Phi(v_i)\right)\cdot\left(\Phi(x_k)-\Phi(x_k)\right)\\ &= \Phi(x_k)\cdot\Phi(x_k)-2\Phi(x_k)\Phi(v_i)\cdot\Phi(v_i)\\ &= K(x_k,x_k)-2K(x_k,v_i)+K(v_i,v_i)\end{aligned}$$

If we use log function $K(x, y) = \log(1+\alpha \| x_j - v_i \|^2)$ as a kernel function the formulation would be as follows:

$$K(x_k, x_k) = 0,$$

hence,

$$\left\|\Phi(x_k)-\Phi(v_i)\right\|^2 = -2\left(K(x_k,v_i)\right)$$

$$\begin{aligned}D_{ij}^2 &= \left\|\Phi(x_i)-\Phi(v_i)\right\|^2\\ &= -2\log\left(1+\alpha\left\|x_j - v_i\right\|^2\right).\end{aligned}$$

6 Compute the new centroids by taking the Partial derivative of J_{UKPC-L} with respect to v_j is and equating to zero

$$\frac{\partial J}{\partial v_i} = \sum_{i=1}^{n} u_{ij}^m\left(\frac{1}{1+\left(\alpha\left\|x_j-v_i\right\|^2\right)}\cdot\left(x_j - v_i\right)\right) = 0.$$

We get

$$v_i = \frac{\sum_{j=1}^{n} u_{ij}^m x_j\left(\frac{1}{1+\left(\alpha\left\|x_j-v_i\right\|^2\right)}\right)}{\sum_{j=1}^{n}\frac{u_{ij}^m}{1+\left(\alpha\left\|x_j-v_i\right\|^2\right)}} \tag{18}$$

7 Compute the membership degree u_{ij} by taking the partial derivative of $J_{UKPC\text{-}L}$ with respect to v_j is and equating to zero

$$\frac{\partial J}{\partial u} = mu_{ij}^{m-1} D_{ij}^2 + \frac{\beta}{m^2 \sqrt{c}} \cdot \left(mu_{ij}^{m-1} \log u_{ij}^m \right) = 0$$

We get

$$u_{ij} = \exp\left(\frac{2m\sqrt{c}}{\beta} \log\left(1 + \alpha \left\| x_j - v_i \right\|^2 \right) \right) \tag{19}$$

8 Repeat Steps 4 to 7 until the following termination criterion is satisfied:

$$\left\| J^{Present} - J^{Previous} \right\| < \varepsilon$$

5 Experimental results

A series of experiments were carried out for a variety of datasets using the UKPC-L clustering. The objective of this comprehensive suite of experiments is to come up with a thorough comparison of the performance of the UKPC-L clustering with conventional clustering techniques like FCM, PCM, UPC, KFCM, and UKPC-G. In this section, we conducted experiments on two artificial datasets and four real datasets to test the performance of various clustering algorithms. The datasets employed are IRIS data, Seed data, Abalone data and Zoo dataset. The selection of datasets includes variety in the shape of clusters, number of data points and count of features of each datum. For all datasets, we choose $m = 2$ which is a common choice for fuzzy clustering, maximum iteration = 100 and termination constant $\varepsilon = 0.001$.

5.1 Synthetic datasets

Example 1: R4 Gaussian random data (identical data with noise)

Algorithms: FCM, PCM, KFCM, UPC, UKPC-G, UKPC-L

Number of clusters: four clusters with two-dimensional data with noise

The dataset generated is spherical with unequal radii. There are four clusters and the data points in each cluster are normally distributed over two-dimensional space. The Ideal centroids for this dataset are:

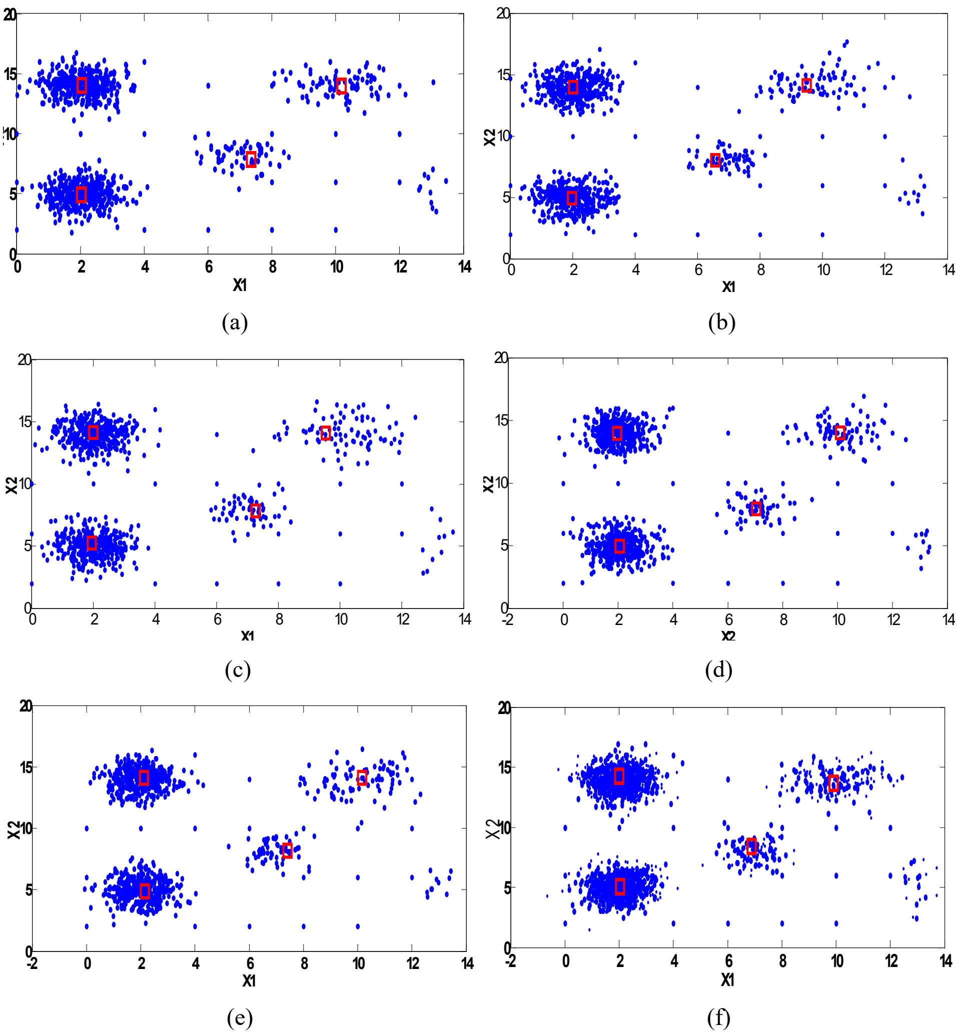
$$V_{ideal} = \begin{bmatrix} 2 & 5 \\ 7 & 8 \\ 2 & 14 \\ 10 & 14 \end{bmatrix}$$

The respective covariance matrices are:

$$\begin{bmatrix} 0.4 & 0 \\ 0 & 0.8 \end{bmatrix}, \begin{bmatrix} 0.4 & 0 \\ 0 & 0.8 \end{bmatrix}, \begin{bmatrix} 0.4 & 0 \\ 0 & 0.8 \end{bmatrix} \text{and} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Let V_{FCM}, V_{PCM}, V_{KFCM}, V_{UPC}, $V_{UKPC\text{-}G}$, $V_{UKPC\text{-}L}$ be the final centroids identified by their respective algorithms. Figure 2 displays the clustering results of different algorithms. The results of the final centroids identified by their respective algorithms are tabulated in Table 1.

Figure 2 (a) Clustering result of FCM (b) Clustering result of PCM (c) Clustering result of KFCM(d) Clustering result of UPC (e) Clustering result of UKPC-G (f) Clustering result of UKPC-L (see online version for colours)



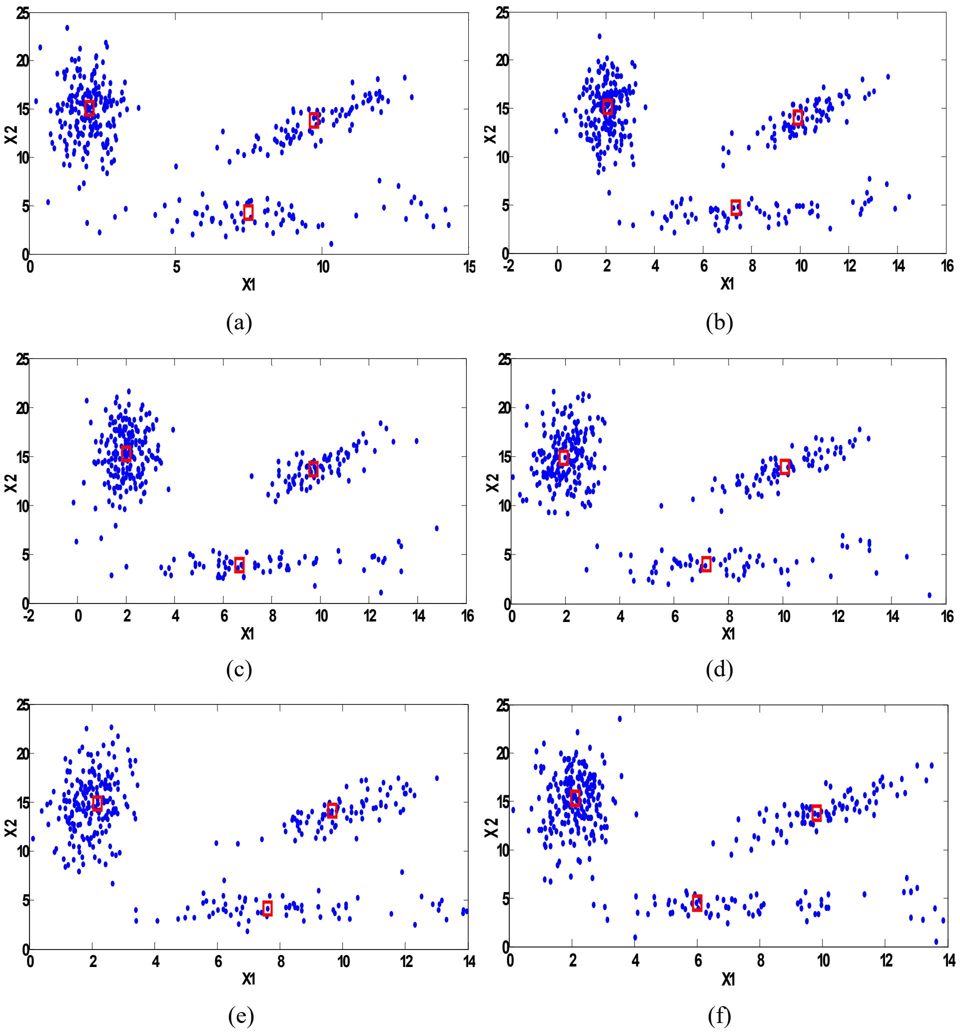
Note: Centroids are shown by '■'.

Analysing Figure 2 closely, our proposed algorithm gives desired results with cluster centres located at their prototypical locations, while in other cases, the cluster centres are slightly shifted from the ideal locations.

Table 1 Terminal centroids produced by FCM, PCM, KFCM, UPC, UKPC-G and UKPC-L for R3 and R4 randomly generated datasets and their corresponding error rate

	Name of clustering algorithm					
	Terminal centroids generated by clustering techniques					
DATASETS	*FCM* (V_{FCM})	*PCM* (V_{PCM})	*KFCM* (V_{KFCM})	*UPC* (V_{UPC})	*UKPC-G* ($V_{UKPC\text{-}G}$)	*UKPC-L (PROPOSED)* ($V_{UKPC\text{-}L}$)
R3	$\begin{pmatrix} 9.75 & 13.6 \\ 2.10 & 15.46 \\ 6.91 & 4.23 \end{pmatrix}$	$\begin{pmatrix} 9.75 & 14.08 \\ 2.07 & 15.4 \\ 7.10 & 4.69 \end{pmatrix}$	$\begin{pmatrix} 9.95 & 14.36 \\ 1.93 & 14.41 \\ 8.30 & 3.93 \end{pmatrix}$	$\begin{pmatrix} 10.2 & 14.0 \\ 2.0 & 14.80 \\ 7.2 & 3.9 \end{pmatrix}$	$\begin{pmatrix} 9.68 & 13.94 \\ 2.0 & 15.5 \\ 7.44 & 4.19 \end{pmatrix}$	$\begin{pmatrix} 10.08 & 14.04 \\ 1.875 & 15.30 \\ 7.60 & 3.95 \end{pmatrix}$
Error rate	0.876	0.4899	0.8524	0.5066	0.4419	0.4273
R4	$\begin{pmatrix} 10.13 & 13.86 \\ 2.015 & 13.95 \\ 1.949 & 4.980 \\ 7.322 & 7.742 \end{pmatrix}$	$\begin{pmatrix} 10.035 & 14.037 \\ 1.996 & 13.993 \\ 2.052 & 4.973 \\ 6.466 & 7.885 \end{pmatrix}$	$\begin{pmatrix} 9.529 & 14.079 \\ 2.009 & 14.138 \\ 1.955 & 5.251 \\ 7.256 & 7.830 \end{pmatrix}$	$\begin{pmatrix} 10.13 & 13.86 \\ 2.00 & 14.08 \\ 1.98 & 4.95 \\ 7.1421 & 8.1931 \end{pmatrix}$	$\begin{pmatrix} 10.001 & 14.006 \\ 1.98 & 13.91 \\ 2.145 & 5.052 \\ 7.434 & 8.1111 \end{pmatrix}$	$\begin{pmatrix} 9.99 & 14.00 \\ 1.91 & 13.95 \\ 2.097 & 5.03 \\ 7.095 & 8.09 \end{pmatrix}$
Error rate	0.2393	0.2759	0.3190	0.1621	0.2407	0.1465

Figure 3 (a) Clustering result of FCM (b) Clustering result of PCM (c) Clustering result of KFCM (d) Clustering result of UPC (e) Clustering result of UKPC-G (f) Clustering result of UKPC-L (see online version for colours)



Note: Centroids are shown by '■'.

To show the effectiveness of the proposed algorithm, we also compute the error rate

$$E^* = \left\| V_{ideal} - V^* \right\|^2$$

where * corresponds to FCM, PCM, KFCM, UPC, UKPC-G,UKPC-L as shown in Table 1.

Example 2: R3 Non-Identical Dataset with noise

Algorithms: FCM, PCM, KFCM, UPC, UKPC-G, UKPC-Log

Number of Clusters: 3 Clusters with 2-dimensional data

In the next experiment, we generate a dataset with unequal sized clusters. This dataset consists of three clusters in 2-D space which satisfy the standard normal distribution, whose centres are (10, 14), (2, 15) and (7, 4) and their respective covariance matrices are

$$\begin{bmatrix} 2 & 2 \\ 1 & 3 \end{bmatrix}, \begin{bmatrix} 0.4 & 0.2 \\ 1 & 8 \end{bmatrix}, \begin{bmatrix} 4 & 0 \\ 0 & 0.8 \end{bmatrix}$$

In the next experiment, we generate a dataset with unequal sized clusters. This dataset consists of three clusters in 2-D space which satisfy the standard normal distribution, whose centres are (10, 14), (2, 15) and (7, 4) and their respective covariance matrices are Besides this, there are noisy points which are randomly distributed over the region whose centres is (13, 5) in order to obtain a difference in the volume between the clusters. Figure 3 displays the clustering results of different clustering techniques on R3 dataset. The results of the final centroids and the error rate identified by their respective algorithms are tabulated in Table 1.

5.2 Real life datasets

A number of real datasets were experimented with including: Iris, Abalone, Seed and Zoo datasets. The clustering results were assessed using Huang's accuracy measure (Huang and Ng, 1999):

$$r = \frac{\sum_{i=1}^{k} n_i}{n} \tag{20}$$

where n_i is the number of data occurring in both the i^{th} cluster and its corresponding true cluster and n is the number of data points in the dataset. The higher value of accuracy measure r proves superior clustering results with perfect clustering generating a value $r = 1$.

Example 1: Iris dataset

Algorithms: FCM, PCM, KFCM, UPC, UKPC-G, UKPC-L

Number of clusters: 3 clusters (4-dimensional data)

This four-dimensional dataset contains 50 samples each of three types of Iris flowers. One of the three clusters (class 1) is distinguished from the other two, while classes 2 and 3 have some overlap. We executed several runs of UPC and our proposed method when these algorithms are initialised with FCM terminal prototypes. Table 1 compares the algorithms based on the optimisation efficiency as judged by the above equation. As indicated in Table 2, the typical result of comparing FCM partitions to the physically correct labels of Iris is 16 errors. PCM gives 14 errors. KFCM gives 11 errors. UPC also gives 12 errors. UKPC-G gives maximum 17 errors. Our proposed algorithm, UKPC-L gives only 10 errors with an accuracy of 93.33%.

Example 2: ABALONE DATASET

Algorithms: FCM, PCM, KFCM, UPC, UKPC-G, UKPC-L

Number of clusters: 3 clusters (8-dimensional data)

The age of abalone is determined by cutting the shell through the cone, staining it, and counting the number of rings through a microscope – a boring and time-consuming task. Physical measurements, which are easier to obtain, are used to predict the age and the given data is obtained. As seen from Table 2, the best result is given by UKPC-L in which 1,979 points are mis-clustered with an accuracy value of 52.62%. FCM and PCM have same performance with 2,026 misclassifications. Number of misclassifications given by KFCM is 2,040. UPC gives the maximum, 2,112 errors and UKPC-G gives 2,072 errors.

Example 3: Zoo dataset

Algorithms: FCM, PCM, KFCM, UPC, UKPC-G, UKPC-L

Number of clusters: 2 clusters (17-dimensional data)

Table 2 Comparative analysis of clustering techniques on parameters like number of misclassified data, elapsed time, accuracy and error (see online version for colours)

Datasets	*Clustering*	*Misclassification*	*Elapsed time*	*Accuracy*	*Error*
Iris dataset	FCM	16	0.76	89.33%	10.67%
	PCM	14	1.45	90.67%	9.33%
	KFCM	11	0.85	92.67%	7.33%
	UPC	12	0.23	92.00%	8.00%
	UKPC-G	17	0.73	88.67%	11.33%
	UKPC-LOG (Proposed)	10	0.71	93.33%	6.67%
Abalone dataset	FCM	2026	1.17	51.50%	48.50%
	PCM	2026	2.34	51.50%	48.50%
	KFCM	2040	1.8	51.16%	48.84%
	UPC	2112	0.72	49.44%	50.56%
	UKPC-G	2072	2.09	50.44%	49.56%
	UKPC-LOG (Proposed)	1972	1.97	52.62%	47.38%
Zoo dataset	FCM	21	0.21	79.22%	20.79%
	PCM	20	0.47	80.20%	19.80%
	KFCM	21	0.28	79.21%	20.79%
	UPC	20	0.24	80.20%	19.80%
	UKPC-G	21	0.24	79.21%	20.79%
	UKPC-LOG (Proposed)	19	0.24	81.20%	18.80%
Seed dataset	FCM	22	0.71	89.52%	10.48%
	PCM	25	0.76	88.10%	11.90%
	KFCM	22	0.28	89.52%	10.48%
	UPC	23	0.45	89.05%	10.95%
	UKPC-G	27	0.78	86.14%	13.86%
	UKPC-LOG (Proposed)	20	0.79	91.55%	8.45%

It is a simple database containing 17 Boolean-valued attributes. The 'type' attribute appears to be the class attribute. The animals are of the following types: buffalo, chicken, sea snake, carp, frog, flea and crab. 101 different animals are assigned a class belonging to the above types. Table 2 shows the misclassifications results of different clustering algorithms. From the table, it can be seen that all clustering algorithms give almost same performance for zoo dataset. The best results are again shown by UKPC-L with misclassification of 19 points and accuracy of 81.19%.

Example 4: Seed dataset

Algorithms: FCM, PCM, KFCM, UPC, UKPC-G, UKPC-L

Number of clusters: 3 clusters (7-dimensional data)

The examined group comprised kernels belonging to three different varieties of wheat: Kama, Rosa and Canadian, 70 elements each, randomly selected for the experiment. High quality visualisation of the internal kernel structure was detected using a soft X-ray technique. From Table 2, it can be seen that all clustering algorithms give almost same performance for seed dataset. The best results are shown by UKPC-L with misclassification of 20 points and accuracy of 91.54%.

We have analysed and compared our proposed UKPC-L algorithm with spectral clustering algorithm on real life datasets like Iris, Abalone, Zoo and Seed defined and explained above. We have taken the misclassification criteria and observed that our proposed algorithm is giving better performance as compared to spectral clustering technique. Table 3 demonstrates the misclassified data on real life datasets using UKPC-L and spectral clustering technique.

Table 3 Misclassification data on real datasets

Datasets	*Number of misclassified data*	
	Using UKPC-L algorithm	*Using spectral clustering*
Iris	10	14
Abalone	1,979	1,991
Zoo	19	25
Seed	20	27

6 Conclusions

This paper has demonstrated the detailed comparative analysis of our proposed log kernel-based unsupervised possibilistic clustering with various types of typical conventional clustering methods. In literature, most kernel-based clustering algorithms use Gaussian kernel functions. We have presented use of log-based kernel function which is conditionally positive definite function. We incorporated this kernel function in unsupervised possibilistic clustering and applied this clustering algorithm on a wide variety of synthetic data as well as real life datasets. From the experiments, we have seen that the log-based kernel function have worked better than FCM, PCM, KFCM, UPC and UKPC-G on various comparative factors like ideal centroid, misclassification, accuracy and error rate. We have also proved the efficiency of our proposed UKPC-L algorithm with spectral clustering technique. In future, we can extend this proposed log kernel-based possibilistic clustering on Image Segmentation to prove its efficiency.

References

Backer, E. and Jain, A.K. (1981) 'A clustering performance measure based on fuzzy set decomposition', *IEEE Trans. Pattern Anal. Mach Intell.*, Vol. 3, No. 1, pp.66–74.

Bezdek, J.C. (1981) *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York.

Camastra, F. and Verri, A. (2005) 'A novel kernel method for clustering', *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 27, No. 5, pp.801–805.

Dave, R. (1991) 'Characterization and detection of noise in clustering', *Pattern Rec. Lett.*, Vol. 12, No. 11, pp.657–664.

Dhillon, I.S., Guan, Y. and Kulis, B. (2004) 'Kernel kmeans', *Spectral Clustering and Normalized Cuts*, KDD.

Ding, C. and He, X. (2004) 'K-means clustering via principal component analysis', *Proc. of Int'l Conf. Machine Learning*, pp.225–232.

Dunn, J.C. (1973) 'A fuzzy relative of the ISODATA process and its use in detecting compact well-separated cluster', *J. Cybern.*, Vol. 3, No. 3, pp.32–57.

Girolami, M. (2002) 'Mercer Kernel-based clustering in feature space', *IEEE Trans. Neural Networks*, Vol. 13, No. 3, pp.780–784.

Hu, Y., Zuo, C., Qu, F. and Shi, W. (2012) 'Unsupervised possibilistic clustering based on kernel methods', *Proc. of International Conference on Solid State Devices and Materials Science*, pp.1084–1090.

Huang, Z. and Ng, M.K. (1999) 'A fuzzy k-modes algorithm for clustering categorical data', *IEEE Trans. Fuzzy Syst.*, Vol. 7, No. 4, pp.446–452.

Krishnapuram, R. and Keller, J.M. (1993) 'A possibilistic approach to clustering', *IEEE Trans. Fuzzy Syst.*, Vol. 1, No. 2, pp.98–110.

Pal, N.R., Pal, K. and Bezdek, J.C. (1997) 'A mixed c-means clustering model', *Proceedings of the IEEE Int. Conf. on Fuzzy Systems*, pp.11–21, Spain.

Ruspini, E.H. (1969) 'A new approach to clustering', *Inform. Control*, Vol. 15, No. 1, pp.22–32.

Wu, Z.D. and Xie, W.X. (2003) 'Fuzzy c-means clustering algorithm based on kernel methods', *Proc. of fifth Intl. Conf. on Computational Intelligence and Multimedia Applications*, pp.47–54.

Wu, Z.D. and Xie, W.X. (2003) 'Fuzzy c-means clustering algorithm based on kernel methods', *Proc. of fifth Intl. Conf. on Computational Intelligence and Multimedia Applications*, pp.47–54.

Xu, R. and Wunsch II, D. (2005) 'Survey of clustering algorithms', *IEEE Trans. Neural Networks*, Vol. 16, No. 3, pp.645–678.

Yang, M.S. and Wu, K.L. (2006) 'Unsupervised possibilistic clustering', *Pattern Recognition*, Vol. 39, No. 1, pp.5–21.

Zadeh, L.A. (1965) 'Fuzzy sets', *Inf. Control*, Vol. 8, No. 3, pp.338–353.

Zhang, D.Q. and Chen, S.C. (2002) 'Fuzzy clustering using kernel methods', *Proc. of Intl. Conf. on Control and Automation*, pp.123–128, China.