Research Journal of Applied Sciences, Engineering and Technology 9(12): 1153-1163, 2015
DOI:10.19026/rjaset.9.2611
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: December 29, 2014 Accepted: January 27, 2015 Published: April 25, 2015

Research Article

Automated Kernel Independent Component Analysis Based Two Variable Weighted Multi-view Clustering for Complete and Incomplete Dataset

[1]M. Kalaiarasu and [2]R. Radhakrishnan
[1]Department of IT, Sri Ramakrishna Engineering College, Coimbatore, India
[2]Vidhya Mandhir Institute of Technology, Tamilnadu, India

Abstract: In recent years, data are collected to a greater extent from several sources or represented by multiple views, in which different views express different point of views of the data. Even though each view might be individually exploited for discovering patterns by clustering, the clustering performance could be further perfect by exploring the valuable information among multiple views. On the other hand, several applications offer only a partial mapping among the two levels of variables such as the view weights and the variables weights views, developing a complication for current approaches, since incomplete view of the data are not supported by these approaches. In order to overcome this complication, proposed a Kernel-based Independent Component Analysis (KICA) based on steepest descent subspace two variables weighted clustering in this study and it is named as KICASDSTWC that can execute with an incomplete mapping. Independent Component Analysis (ICA) which exploit distinguish operations depending on canonical correlations in a reproducing kernel Hilbert space. Centroid values of the subspace clustering approaches are optimized depending on steepest descent algorithm and Artificial Fish Swarm Optimization (AFSO) algorithm for the purpose of weight calculation to recognize the compactness of the view and a variable weight. This framework permits the integration of complete and incomplete views of data. Experimental observations on three real-life data sets and the outcome have revealed that the proposed KICASDSTWC considerably outperforms all the competing approaches in terms of Precision, Recall, F Measure, Average Cluster Entropy (ACE) and Accuracy for both complete and incomplete view of the data with respect to the true clusters in the data.

Keywords: Artificial Fish Swarm Optimization (AFSO) variable weighting, Augmented Lagrangian Cauchy Step computation (ALCS), clustering, data mining, fuzzy centroid, incomplete view data, Kernel-based Independent Component Analysis (KICA), mulitiview data, Steepest Descent Algorithm (SDA), subspace clustering, view weighting

INTRODUCTION

In several real world data mining complications, the identical instance possibly will exist in several datasets with dissimilar representations. Various datasets might highlight different features of instances. An example is clustering the users in a user-oriented recommendation system. For this process, related datasets can be:

- User profile database
- Users' log data
- Users' credit score

Learning with this kind of data is generally referred as multi-view learning (Bickel and Scheffer, 2004). Even though there are some earlier researches on multiple datasets, the entire presume the completeness of the different datasets. Multi-view learning is particularly appropriate for applications that concurrently gather data from several modalities, with each unique modality presenting one or more views of the data.

In the past decade, multi-view data has raised interests in the so-called multi-view clustering (Tzortzis and Likas, 2010; Long *et al*., 2008; Greene and Cunningham, 2009). Different from the traditional clustering methods which take multiple views as a flat set of variables and ignore the differences among different views, multi-view clustering exploits the information from multiple views and take the differences among different views into consideration in order to produce a more accurate and robust partitioning of the data.

Variable weighting clustering has been main research subject in the field of cluster analysis (Deng *et al*., 2010; Cheng *et al*., 2008). It automatically works out a weight for each variable and recognizes significant variables and irrelevant variables through

Corresponding Author: M. Kalaiarasu, Department of IT, Sri Ramakrishna Engineering College, Coimbatore, India

variable weights. The multi-view data may perhaps be regarded as have two levels of variables. In case of clustering the multi-view data, the divergence of views and the significance of individual variables in each view should be considered. The conventional variable weighting clustering techniques only calculate weights for individual variables and pay no attention to the differences in views in the multi-view data. As a result, they are not appropriate for multi-view data. On the other hand, in the real world applications, there are several circumstances in which complete datasets are not available.

Existing multi-view algorithms characteristically presume that there is a complete bipartite mapping among instances in the different views to characterize these correspondences, symbolizing that each object is represented in all views. Subsequently, the mapping among instances in the different views is not complete. Even in certain cases where the connections among views are recorded, sensor availability and scheduling possibly will result in several isolated instances in the various views. Even though it is practical to recognize a partial mapping between the views, the lack of an absolute bipartite mapping presents a complication to most existing multi-view learning approaches. Without a complete mapping, these approaches will be incapable to transmit any information concerning an isolated instance to the other views.

The most important motivation of the proposed approach is to resolve the setback of weight value computation and centroid selection in multi-view data with incomplete data point of view, because the entire existing multi-view clustering data suitable only for clustering complete multi-view data. In the proposed KICASDSTWC approach for clustering both complete and incomplete view data in multi-view data. Incomplete view of data is carried out by proposing Kernel-based Independent Component Analysis (KICA), that differentiate the complete and incomplete view of multi view data, in addition differentiate the impacts of different views and different variables in clustering, the weights of views and individual variables are automatically computed depending on the AFSA. As a result, the view weights replicate the significance of the views in the complete data, at the same time the variable weights in a view only replicate the significance of variables in the view. In Steepest Descent Algorithm is proposed to select or optimize the fuzzy centroid values, Singular Value Decomposition (SVD) to lessen the complexity of clustering. Augmented Lagrangian Cauchy Step computation (ALCS) to score the objects in subspaces where they are homogeneous and have elevated correlated utilities. The proposed KICASDSTWC is extended to support both incomplete and complete view data; it becomes efficient in clustering large high dimensional multi-view data.

LITERATURE REVIEW

In recent times, numerous multi-view clustering algorithms have been developed (Chaudhuri *et al*., 2009). These multi-view clustering approaches have been shown to provide enhanced performance in comparison to single-view approaches. On the other hand, the drawbacks of certain approaches are clear. For example, few approaches presume that the dimensions of the characteristics in multiple views are similar, restricting their applicability to the homogeneous circumstances. Few other approaches simply focus on the clustering of two-view data in order that it might be tough to broaden them to more than a two-view circumstance. Also, a suitable weighting approach is missing for these multiple views, even though coordinating different information is also one critical step in acquiring better clustering outcomes (Tang *et al*., 2009). An integrated framework that can incorporate several categories of multi- view data is lacking (Tang *et al*., 2010).

Conventionally, tensor-based approaches have been exploited to model multi-view data (Kolda and Bader, 2009). Tensors are higher-order generalizations of matrices and certain tensor approaches are incredibly great to analyze the latent pattern unknown in the multi-view data. Tensor decompositions (Kolda and Bader, 2009) obtain multi-linear structures in higher-order data-sets, in which the data have over two modes. Tensor decompositions and multi-way investigation permit naturally obtaining hidden components and examining complex association among them. Sun *et al*. (2006) commenced a Dynamic Tensor Analysis (DTA) approach and its variants and implement them to anomaly detection and multi-way latent semantic indexing. It seems their clustering approach is intended for dynamic stream data. Dunlavy *et al*. (2006) execute Parallel Factor Analysis (PARAFAC) decomposition for examining scientific publication data with multiple linkages. The last two concepts that incorporate multi-view data as a tensor resemble to this approach. However this approach is based on Tucker-type tensor decomposition.

Chaudhuri *et al*. (2009) developed a clustering approach which carries out clustering on lower dimensional subspace of the multiple views of the data, planned by means of canonical correlation investigation. Two approaches for mixtures of Gaussians and mixtures of log concave distributions were provided. Long *et al*. (2008) developed an all-purpose scheme for multi-view clustering in a distributed framework. This scheme commences the idea of mapping function to enable the several patterns from several pattern spaces comparable and therefore a best possible pattern can be learned from the multiple patterns of multiple views.

Greene and Cunningham (2009) developed a clustering approach for multi-view data with the help of a late integration strategy. In this approach, a matrix that includes the partitioning of each individual view is generated and then segmented to two matrices with the help of matrix factorization approach: the one represents the contribution of those partitioning to the concluding multi-view clusters, called metaclusters and the rest represent instances to the metaclusters

Cohn *et al*. (2009) provided an interactive scheme in which a user continuously offers feedback to enhance the quality of a proposed clustering. In both of these situations, the user feedback is integrated in the form of constraints. This interactive scheme is a constructive extension that possibly will allow user knowledge to be brought into a multi-view clustering approach.

On the other hand, with all of the above approaches for multi-view clustering of the complete view, these limitations may be between instances that do not have equivalences in the other views and weight value calculation depends on the view also not supported by these approaches, thus facilitate a challenge to multi-view learning, in particular when the mapping is very limited.

METHODOLOGY

Based For the purpose of multi-view clustering with both complete and incomplete view of the data, a novel fast Kernel-based Independent Component Analysis and Steepest Descent Subspace Two variable Weighted Clustering (KICASDSTWC) with incomplete view methods has been proposed in this study. The proposed KICASDSTWC with incomplete view methods in which incomplete data are transformed into complete data by proposing the KICA in which the subspaces are created in accordance with a set of centroids for total dataset results from KICA based is calculated based Gradient descent method along with user's domain knowledge of utility function. The proposed method distinguishes the impacts of several views and several variables by introducing the weights of views and individual variables to the distance function. The view weights are calculated from the complete variables, at the same time the variable weights in a view are calculated from the subset of the data which comprises only the variables in the view.

As a result, the significance of the views in the complete data is reflected by the view weights while the significance of variables in the view is reflected by the variable weights in a view. The automatic calculations of the centroid value for the specific data through the proposed gradient descent method differentiate it from other existing clustering approaches. At the beginning, the input data results from KIC values are transformed into the fuzzy centroid values followed by which the fuzzy centroid values are optimized using the gradient descent method. With the assistance of the algorithm, the view and variable weights of the KICASDSTWC values in the objective function are optimized by employing Artificial Fish Swarm Optimization (AFSO).

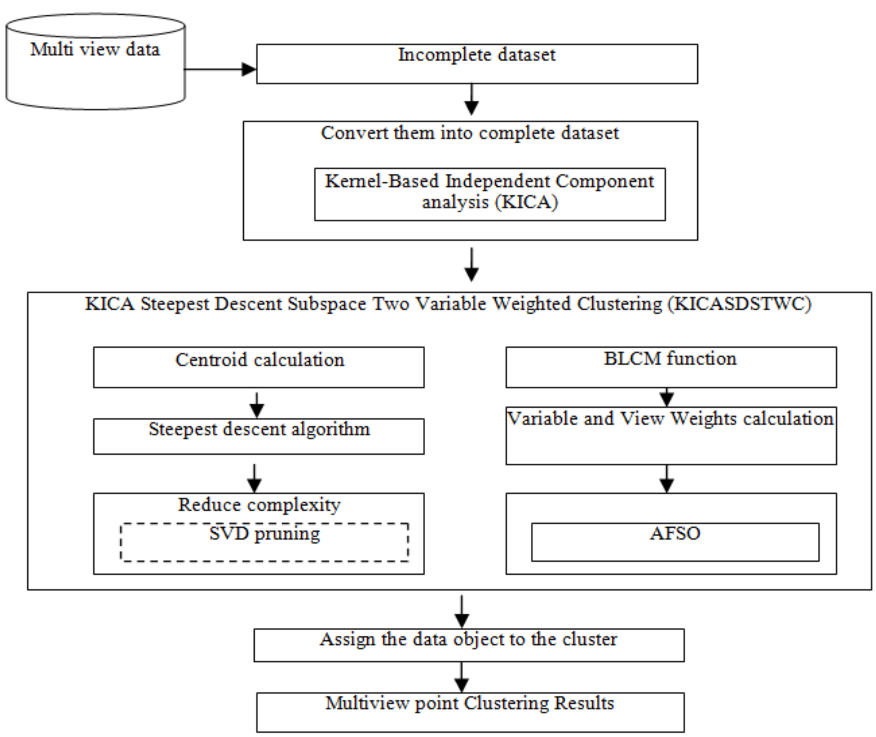

Fig. 1: Flowchart representation of proposed methodology

The complete work representation of the proposed study is shown in Fig. 1.

In order to carry out multi-view clustering for both complete and incomplete dataset, at first the incomplete dataset is transformed into complete dataset by proposing Kernel-based Independent Component Analysis (KICA). For ease of understanding, consider X and Y represent the two number complete and incomplete dataset respectively. Generalization to over two types of complete and incomplete dataset with one complete and remaining incomplete dataset can be performed in a similar way. Assume that complete multi-view data is indicated as X while incomplete multi-view dataset is indicated as Y i.e., the variables values of the multi-view data are available for only a subset of the entire examples. To formalize and discover incomplete data, KICA is proposed to find the values of the variables. KICA learning approaches exploit the following concept: by means of a nonlinear mapping for both complete and incomplete view of the multi-view data samples:

$$\Phi: \Re^{t} \rightarrow \mathrm{MVDV}, \mathrm{X} \rightarrow \Phi(\mathrm{x}), \mathrm{Y} \rightarrow \Phi(\mathrm{y}) \quad (1)$$

The data in the input space$x_1, x_2, \dots, x_N \in \Re^{t}$is plotted to a potentially much elevated dimensional complete multi-view data with variable space V deal, kernel value $\Phi(x_1), \Phi(x_2), \dots, \Phi(x_N)$. $y_1, y_2, \dots, y_N \in \Re^{t}$ is plotted to a potentially much elevated dimensional incomplete multi-view data with variable space V deal with kernel value $\Phi(y_1), \Phi(y_2), \dots, \Phi(y_N)$. In case, if the learning of incomplete and complete multi-view data can be expressed based on inner products with correct nonlinear mapping Φ. In eigen-decomposition of a positive function (the kernel) is exploited to describe the following inner product for the transformation space:

$$\mathrm{K}_{\sigma}(\mathrm{x}-\mathrm{y}) = \sum_{\mathrm{k}=1}^{\infty} \lambda_{\mathrm{k}} \Phi_{\mathrm{k}}(\mathrm{x}) \Phi_{\mathrm{k}}(\mathrm{y}) = \langle \Phi_{\mathrm{k}}(\mathrm{x}) \Phi_{\mathrm{k}}(\mathrm{y}) \rangle \quad (2)$$

where, $\langle .,. \rangle$ represents an inner product, the Φ_{k}s indicates the eigen-functions of the kernel and λ_{k} denotes the related eigen values. Kernel ICA presumes a Reproducing Kernel Hilbert Space (RKHS) of the random variable V with kernel $K(x-y)$ and feature map $\Phi_k(x) = \kappa(\cdot, x)$. Subsequently, the F-correlation among incomplete and complete variable multi-view data is given as the maximal correlation among the two random variables $f_1(x_1)$ and $f_2(y_1)$ in which $f_1$and $f_2$range over random variable V in F correlation as below:

$$\rho = \max_{\mathrm{f}_1, \mathrm{f}_2} \operatorname{corr}(\mathrm{f}_1(\mathrm{x}_1) \mathrm{f}_2(\mathrm{y}_1)) \quad (3)$$

$$\max_{\mathrm{f}_1, \mathrm{f}_2} \frac{\operatorname{cov}(\mathrm{f}_1(\mathrm{x}_1) \mathrm{f}_2(\mathrm{y}_1))}{\sqrt{\operatorname{var} \mathrm{f}_1(\mathrm{x}_1) \mathrm{f}_2(\mathrm{y}_1)}} \quad (4)$$

Noticeably, in case if the random variables x_1 and y_1 are independent, at that time the F-correlation for mutliview data among complete and incomplete data happens to be zero. Furthermore, the reverse is also true provided that the set F is large enough. This indicates that $\rho = 0$ implies x_1 and y_1 are independent. With the intention of obtaining a computationally tractable implementation of F-correlation, the reproducing property of RKHS is exploited to estimate the F-correlation:

$$\mathrm{f}(\mathrm{x}) = \langle \Phi_{\mathrm{k}}(\mathrm{x}), \mathrm{f} \rangle = \langle \kappa(\cdot, x), \mathrm{f} \rangle \quad (5)$$

Consider $S_1$and S_2 represent the linear spaces spanned by the Φ-images of the data samples, then f_1 and f_2 can be fragmented into two parts, i.e.:

$$\mathrm{f}_1 = \sum_{\mathrm{k}=1}^{\mathrm{N}} \alpha_1^{\mathrm{k}} \Phi(\mathrm{x}_1^{\mathrm{k}}) + \mathrm{f}_1^{\perp} \quad (6)$$

$$\mathrm{f}_2 = \sum_{\mathrm{k}=1}^{\mathrm{N}} \alpha_2^{\mathrm{k}} \Phi(\mathrm{x}_2^{\mathrm{k}}) + \mathrm{f}_2^{\perp} \quad (7)$$

where, $f_1^{\perp}$ and $f_2^{\perp}$ are orthogonal to S_1 and S_2 correspondingly. With the help of the empirical complete and incomplete view of mutliview data to approximate the population value, the F correlation can be given as:

$$\hat{\rho} = \max_{\alpha_1, \alpha_2 \in \Re^{\mathrm{N}}} \frac{\alpha_1^{\mathrm{T}} \mathrm{K}_1 \mathrm{K}_2 \alpha_2}{\sqrt{(\alpha_1^{\mathrm{T}} \mathrm{K}_1^2 \alpha_1)(\alpha_2^{\mathrm{T}} \mathrm{K}_2^2 \alpha_2)}} \quad (8)$$

where, K_1 and K_2 represent the gram matrices linked with the complete and incomplete view of multi-view datasets $\{x_1^k\}$ and $\{y_1^k\}$ given as:

$$(\mathrm{K}_{\mathrm{i}})_{\mathrm{a,b}} = \mathrm{k}(x_i^a, y_i^b) \quad (9)$$

The above kernel based system determines the resemblance value between the incomplete and complete view for multi view data. Once multi-view dataset incomplete and complete view of the data are discovered, subsequently carry out centroid value computation with the assistance of Steepest Descent Algorithm (SDA) for that purpose the complete and incomplete view results from KICA is given as $MV\mathrm{D} = \{\mathrm{z}_1, \dots \mathrm{Z}_{\mathrm{n}}\}$, its dimensions are described by set of n objects mo represented by the set A of m variables and view weights VW. Consider the value of object mo on attribute a and in time weight values is indicated by $\mathrm{muv}_{\mathrm{moaw}}$. Also consider csd represent an object chosen as the centroid from SDA. In addition, $h_{csd}(muv_{moawvw}) = Smu_{moawvw}$ is indicated as a homogeneous function to determine the homogeneity among object mo and centroid csd, on attribute a in a multi-view weight value. The users are permitted to define the homogeneous function, however the homogeneous values must be normalized to (0, 1) in order that $\mathrm{Smu}_{\mathrm{moawvw}} = 1$ points out that the value $\mathrm{muv}_{\mathrm{oat}}$ is "perfectly" homogeneous of the centroid $\mathrm{muv}_{\mathrm{cqnoawvw}}$, or else which indicated by value $Smu_{oawvw} = 0$.

The distribution centroid: The fuzzy centroid as developed (Kim *et al*., 2004) simulated the concept of distribution centroid for a improved representation of categorical variables. The cluster centers for the categorical variable part will be better represented by a fuzzy scenario. For $Dom(V_j) = \{v_i^1, v_i^2, v_i^3, \dots v_i^t\}$, the distribution centroid of a cluster mvc is specified as $\mathrm{C}'_{\mathrm{mvc}}$ and given as below:

$$\mathrm{C}'_{\mathrm{mvc}} = \{\mathrm{c}'_{\mathrm{mvc1}}, \mathrm{c}'_{\mathrm{mvc2}}, \dots, \mathrm{c}'_{\mathrm{mvcj}}, \dots \mathrm{c}'_{\mathrm{mvcm}}\} \tag{10}$$

where,

$$\mathrm{c}'_{\mathrm{mvcj}} = \begin{Bmatrix} \{\mathrm{b}_j^1, \mathrm{w}_{\mathrm{mvcj}}^1\}, \{\mathrm{b}_j^2, \mathrm{w}_{\mathrm{mvcj}}^2\}, \dots \\ \{\mathrm{b}_j^k, \mathrm{w}_{\mathrm{mvcj}}^k\}, \dots \{\mathrm{b}_j^t, \mathrm{w}_{\mathrm{mvcj}}^t\} \end{Bmatrix} \tag{11}$$

In the above equation:

$$\mathrm{w}_{\mathrm{mvcj}}^k = \sum_{i=1}^n \mu(\mathrm{x}_{ij}) \tag{12}$$

where,

$$\mu(\mathrm{z}_{ij}) = \begin{cases} \frac{\mathrm{u}_{\mathrm{imvc}}}{\sum_{i=1}^n \mathrm{u}_{\mathrm{imvc}}} \text{ if } \mathrm{z}_{ij} = \mathrm{b}_j^k \\ \text{o if } \mathrm{z}_{ij} \neq \mathrm{b}_j^k \end{cases} \tag{13}$$

At this point, the value of 1 is assigned to $\mathrm{u}_{\mathrm{imvc}}$, if the data object x_i belongs to cluster mvc otherwise which 0 is allocated, if the data object x_i do not belong to cluster mvc. Based on the above mentioned equations 10, 11, 12 and 13, it is obvious that the number of repetitions of each categorical value is been considered by the cluster computation of distribution centroid. As a result, the distribution characteristics of categorical variables are considered to indicate the center of a cluster. In the proposed approach, optimisation of fuzzy membership centroid values is done with the assistance of SDA. SDA becomes an iterative and computations depend on the computation of the objective function $mvfcf$ and $\nabla mvfcf$ at each iteration are commonly concerned. The SDA approach of choosing the best centroid values by maintaining least amount of cluster multi-view datapoints for each cluster, subsequently recur the step until maximum number of points in the cluster is attains, or else go to step 3 and negative direction that is remaining points in the multi-view data are elected to choose optimized centroid value. The optimized centroid values are reduced through calculation of step size in step 4 of the algorithm then revise the chosen optimized fuzzy centroid value results in step 5 and move to step 2. The fundamental form of the algorithm for optimizing the centroid values is given below.

Algorithm 1: Steepest descent algorithm for centroid calculation:

1. Compute distance matrix $Dist_{m\times m}$ in which $dist(Z_i, Z_j)$ indicates distance from Z_i to Z_j
2. Make an initial guess $\mu(z_{ij})$ at the minimum; keep $k = 0$. Choose convergence parameter $\varepsilon > 0$, is calculated from distance matrix
3. Calculate the gradient steepest descent of the centroid objective function $\nabla mvfcf(z)$ at a point Z^k to all other points and centroid is indicated as $c^{(k)} = \nabla \mathrm{mvfcf}(\mathrm{z}^{(k)})$
4. Compute centroid value as $||C|| = \sqrt{C^T C}$, when $||c|| < \varepsilon$ and $utility(u_{mowvw}) > u_{mowvw}(min) > 0.5$ then terminate the iteration process $Z^* = Z^{(k)}$ is minimum number of cluster multi-view data cluster datapoints. Otherwise go to step 3
5. Consider the search direction at the current point $z^{(k)}$ as $d^{(k)} = -c^{(k)}$
6. Compute a step size $\alpha^{(k)}$ to reduce fuzzy centroid value $(z^{(k)} + \alpha^{(k)} d^{(k)})$
7. One dimensional search is exploited to determine $\alpha^{(k)}$
8. Revise the chosen fuzzy centroid values as $z^{(k+1)} = z^{(k)} + \alpha^{(k)} d^{(k)}$
9. Keep $k = k + 1$ and move to step 2

Gaussian function which employed above is the homogeneous function as similarity among data object mo and centroid cqn is been normalized on feature (a, w) to [0; 1]. The homogeneous function is specified as below:

$$h_{csd} = exp\left(-\frac{|v_{csdawvw} - v_{moawvw}|}{2\sigma_{csd}^2}\right) \nabla f\left(\mu(x_{ijk})\right) \tag{14}$$

where, σ_{csd}^2 indicates a parameter which maintains the width of the Gaussian function centered at centroid csd. At this point, the similarity function is not symmetric, i.e., $h_{csd}(v_{moawvw}) \neq h_{mo}(v_{csdawvw})$, as the calculation depends on the distribution of objects centered at the former object. The evaluation of width of the Gaussian function is completed with the help of k-nearest neighbor's heuristic (Nocedal and Wright, 2006) and is given as:

$$\sigma_{\mathrm{csd}} = \frac{1}{\mathrm{k}} \sum_{\mathrm{n} \in \mathrm{Neigh}_{\mathrm{csdawvw}}} \mathrm{dist}_{\mathrm{a}}(\mathrm{csd}, \mathrm{n}) \tag{15}$$

where, $Neigh_{csdawvw}$ represents the set of k-nearest neighbors of object mo on feature (a, vww) and $k = \rho|mo|$ with an supposition that ρ is the neighborhood parameter described by users. In accordance with the distribution of the objects projected in the data space of attribute, the width of the Gaussian function is being implemented by the k-nearest neighbors heuristic, as a result showing that σ_{csd} is more strong than keeping a constant value. Calculating and pruning the homogeneous tensor using SVD for optimized centroid σ_{csd}, a homogeneous tensor $S \in [0,1]^{|mo|\times|a|\times|vww|}$ is characterized containing the homogeneity values MUS_{moawvw} with respect to centroid csd.

Algorithm 2: SVD pruning:
Input |mo| × |a| × |vww| homogenous tensor S
Output: Pruned homogenous S

1. $\mathrm{M = unfold(S)}$
2. Add dummy row and column to M
3. While true do
4. $\mathrm{N \leftarrow}$ zero mean normalization (M)
5. $\mathrm{U\sum V' \leftarrow N}$ //SVD decomposition on N
6. $\mathrm{u \leftarrow principalcomponent(U)}$
7. $\mathrm{v \leftarrow principalcomponent(V')}$
8. Calculate threshold $\tau_u \tau_v$
9. Prune row i of M if $|u(i)| < \tau_u, 1 \le i \le r$
10. Prune column i of M if $|v(j)| < \tau_v, 1 \le j \le n$
11. If there is no pruning then break
12. Remove dummy row and column from M
13. $\mathrm{S = fold(M)}$

Initially, zero mean normalization is carried out on matrix M to get hold of the zero mean normalized matrix N (Line 4), which will later be exploited to compute the covariance matrices. Zero mean normalization is carried out by computing the mean avg_j of the matrix M that $\forall j \in \{1, \dots c\}$ of each column:

$$\mathrm{avg_C} = \frac{1}{r}\sum_{i=1}^{r} \mathrm{M(i,j)} \quad (16)$$

Subsequently, from each entry of M, its equivalent column mean $\forall j \in \{1, \dots c\}$ is been subtracted:

$$\mathrm{N(i,j) = M(i,j) - avg_j} \quad (17)$$

At some point in the performance of the clustering process for the above returned centroid values, the homogeneous tensor S together with the utilities of the objects were exploited to compute the probability of each value $muv_{csdoawvw}$ of the data to be clustered with the centroid csd. Subsequently, the covariance matrices of the homogeneous values in the object space and feature space called NN' and $N'N$ respectively were calculated (N' is the conjugate transpose of matrix N):

$$\mathrm{NN' = U\sum^2 U'} \quad (18)$$

$$\mathrm{N'N = V\sum^2 V'} \quad (19)$$

where, U represents a $r \times r$ orthonormal matrix (its columns are the eigenvectors of NN'), $\sum^2$ represents a $r \times c$ diagonal matrix with the eigen values on the diagonal and V is a $c \times c$ orthonormal matrix (its columns are the eigenvectors of $N'N$). If the magnitude of the pruned objects in their related elements of their principal components is little (Line 9 and 10), a heuristic however parameter-free approach can be proposed to find out the threshold τ_u for pruning objects. For pruned rows (objects) and columns (features) of matrix M, the homogeneous values are fixed to "0". The process of computing SVD and pruning the matrix M is replicated until there is no more pruning. The clustering process for computing the probability value is carried out in which $\mathrm{p_{moawvw}} \in \mathbb{R}$ represent the probability of object mo to be clustered with centroid csdon attribute a. The view weight v, variable weight w for multi-view data is computed with the help of Artificial Fish Swarm Algorithm (AFSA). Consider $P \in \mathbb{R}^{|mo|\times|a|\times|vww|}$ be the probability tensor, such that p_{moawvw} is an element of it provided with the respective indices $|mo| \times |a| \times |vww|$. The following objective function is then maximised to calculate the probabilities: To perform the clustering process, the objective functions are defined as:

$$\mathrm{f(p)} = \sum_{\mathrm{mo \in MO}} \sum_{\mathrm{a \in A}} \sum_{\mathrm{vw \in VW}} \sum_{\mathrm{w \in W}} \frac{\mathrm{p_{moawvw}}}{\mathrm{h(v_{moawvw})}} \mathrm{utility(u_{mowvw})} \quad (20)$$

$$\mathrm{g(p)} = \sum_{\mathrm{mo \in MO}} \sum_{\mathrm{a \in A}} \sum_{\mathrm{vw \in VW}} \sum_{\mathrm{w \in W}} \mathrm{p_{moawvw}} - 1 \quad (21)$$

The Optimization of $f(p)$ under constraint $g(p)$ is a linear programming problem, as $f(p)$ and $g(p)$ is linear functions of the design variable P. Augmented Lagrangian multiplier technique is then exploited to maximize the objective function $f(p)$ for clustering multi-view data in the subspace clustering technique. As a result, the modified objective function is defined as:

$$F(p) = -f(P) - \lambda g(P) + \frac{\mu}{2} g(P)^2 \quad utility(u_{mowvw}) \quad (22)$$

The optimization of $F(p)$ (Algorithm 3) depends on Augmented Lagrangian Cauchy Step computation (ALCS) methods, $f(P)$ and $g(P)$ would be employed by ALCS with the intention that the constrained optimization problem are been replaced with iterations of unconstrained optimization sub problems, Hence, the iterations continue until the solution converges. For algorithm 3, ALCS necessitates three parameters such as μ_k, $\ominus_k$, $\in_k$ to calculate the optimized probability value for clustering process. In the majority of situations, the results are insensitive to these parameters and therefore can be fixed to their default values. The closeness cluster results for multi-view data results is constantly indicated by parameter μ_k. As a result, δ provides the standard tradeoff between accuracy and efficiency, i.e., smaller δ indicates longer computation time however better result. Parameter $\ominus_k$ maintains the level of clustering to the constraint $g(P)$. Parameter $\in_k$ auxiliary nonnegative scalar quantities on $F(P)$ when the constraint is breached.

Algorithm 3: Augmented Lagrangian Cauchy Step computation (ALCS)
Input: Initial probability distribution P_a
Output: The optimal probability distribution P_a^*

1. Initialize $P_a^0, \mu_k > 0, \ominus_k > 0, \in_k > 0, \gamma \in (0,1)$
2. While $P_a^i(z_k) < \mu_k$, true do
3. Set $P_a^* \leftarrow P_a^i(z_k) - z_k$ then return P_a^*
4. If not satisfied do
5. $P_a^* \leftarrow \gamma P_a^i(z_k)$
6. End while
7. If $|g(P_a^i)| < |g(P_a^{i-1})|$ then
8. Return $g(P_a^i)$
9. $\ominus_k \leftarrow \ominus_k . g(P_a^i)$
10. Else $\in_k \leftarrow \in_k . g(P_a^i)$
11. end if
12. $\lambda^i \leftarrow \lambda^i - \in_k . g(P_a^i)$
13. $i \leftarrow i + 1$
14. End procedure

From the outcomes of the optimized probability values for multi-view data both view and variable weights values are calculated using the Artificial Fish Swarm Optimization (AFSO) algorithm. Artificial Fish (AF) is a fictitious entity of true fish, which is exploited to carry on the analysis and explanation of the problem and can be recognized by exploiting an animal ecology conception. The each one of the variable and view weight values of multi-view data to carry out KICASDSTWC results the weight values by external perception by its vision. The number of variable (w) and view weights (vw) values of multi-view data is signified as ZW is the position on AF, $Visual$ indicates the visual distance and ZW_v represents the visual position of the current multi-view data weight values at visual position is superior than the earlier weight values state, it goes forward to the next weight value calculation direction and arrives the ZW_{next} state; or else, current variable (w) and view weights (vw) values maintains an inspecting travel around in the vision until it attains maximum clustering accuracy. Consider ZW represents the current state of the variable (w) and view weights (vw) values and it is indicated as $ZW = \{zw_1, \dots, zw_n\}$ and $ZW_V = \{zw_{1v}, \dots, zw_{nv}\}$ subsequently process can be expressed as given below:

$$zw_i^v = zw_i + Visual.Rand(), i \in (0, n] \tag{23}$$

$$ZW_{next} = ZW + \frac{ZW_v - ZW}{||ZW_v - ZW||} Step.Rand\,() \tag{24}$$

where, $Rand$ () generates random numbers between 0 and 1, Step represents the step length to carry out weight value calculation, n represents the number of multi-view data samples for clustering, δindicates the crowd factor of AFSO algorithm to optimize the Variable and View Weight values is found depending on the input parameters $\eta$and ξ to manage the distribution of the two types of weights VW and W. It can be simply validated whether the objective function (20) can get minimized with regard to VW and W if $\eta \geq 0$ and $\xi \geq 0$. In addition, it is carried out as given below.

$\eta > 0$, based on (25), vw is inversely proportional to E. The smaller E_j and the larger v_j shows that the equivalent variable is more significant.

$\eta > 0$, based on (25), $\eta = 0$ will generate a clustering result with only one significant variable in a view which possibly will not be desirable for high dimensional data. The attributes are presumed to be segmented into T views $\{G_t\}_{t=1}^T$:

$$\delta(vw_j) = \frac{exp\{-E_j/\eta\}}{\sum_{k \in G_t} exp\{-E_k/\eta\}} \tag{25}$$

$$E_j = \sum_{l=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{m} \widehat{N}\,(i, j, mvc) r\widehat{v}w_t \; dist_a(nsd, n) \tag{26}$$

$\xi > 0$, based on (27), w is inversely proportional to D. The smaller D_t, the larger w_t, the more compact the corresponding view.

$\xi > 0$, based on (27), $\xi = 0$ will generate a clustering result with only one significant view. It possibly will not be desirable for multi-view data:

$$\delta(w_t) = \frac{exp\{-D_t/\xi\}}{\sum_{k=1}^{T} exp\left\{-\frac{D_t}{\xi}\right\}} \tag{27}$$

$$D_t = \sum_{l=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{m} \widehat{N}\,(i, j, mvc) \hat{v}_j \; dist_a(nsd, n) \tag{28}$$

The functions multi-view clustering data samples that comprise the behaviors of the AF: AF_Prey, AF_Swarm, AF_Follow, AF_Move. Every fish typically resides in the place with a best objective function (20). The fundamental behaviors of AF are defined (Jiang and Yuan, 2005; Wang *et al*., 2005) as given below for maximum.

AF_Prey: This is a fundamental biological behavior that tends to the each variable (w) and view weights (vw) values is assigned to best variable (w) and view weights (vw) food; commonly the fish give attention to perceives the best variable (w) and view weights (vw) values in water to decide the movement by vision:

$$ZW_j = ZW_i + Visual.Rand() \tag{29}$$

If $ZW_i < ZW_j$ in the higher clustering accuracy it goes to an additional mutliview data samples; if not, choose a state ZW_j arbitrarily yet again to weight calculation and find whether it satisfies the forward condition. If it is not possible to satisfy higher clustering accuracy after maximum number of iterations completed by fish, it travels a step randomly to choose another variable (w) and view weights (vw) values. In case of the maximum number of iterations is small in AF_Prey, the AF can work like a swim approach randomly, which makes it best variable (w) and view weights (vw) values results:

$$ZW_i^{(t+1)} = ZW_i^{(t)} + Visual.Rand() \quad (30)$$

AF_Swarm: The fish will bring together in groups of variable (w) and view weights (vw) values that are obviously assign variable (w) and view weights (vw) values to multi-view data point clustering in the moving procedure, which is a type of living habits to satisfy clustering accuracy and eradicates unnecessary variable (w) and view weights (vw) values. Consider ZW_i represent the AF current state of variable (w) and view weights (vw) values, ZW_c represent the center location of variable (w) and view weights (vw) values and n_f represent the number of its companions in the current neighborhood $(d_{ij} < Visual), n$ represents total number of variable (w) and view weights (vw) values data samples. When $ZW_c > ZW_i$ and $\frac{n_f}{n} > \delta$, which indicates that the companion center has additional clustering accuracy and is not very crowded, it goes forward a step to the companion center:

$$ZW_i^{(t+1)} = ZW_i^{(t)} + \frac{zw_j - zw_i^{(t)}}{\left\|zw_j - zw_i^{(t)}\right\|} Step.Rand() \quad (31)$$

If not, implements the preying behavior. The crowd factor restricts the length of the weight calculation searching space and more AF only cluster at the best possible area, which guarantees that AF move to optimum in a broad field.

AF_Follow: In the moving progression of the weight calculation from one position to many positions then discover best clustering accuracy by comparing the neighbourhood partners will trail and accomplish best clustering accuracy results rapidly:

$$ZW_i^{(t+1)} = ZW_i^{(t)} + Visual.Rand() \quad (32)$$

AF_Move: Fish swim arbitrarily in the water; indeed, they are looking for best weight calculation results food in larger ranges:

$$ZW_i^{(t+1)} = ZW_i^{(t)} + Visual.Rand() \quad (33)$$

AF_Leap: When the objective functions $\delta(vw_i) - \delta(vw_i) < eps$ carried out by fish. It selects fish, certain variable view weights (vw) values arbitrarily in the complete fish swarm and fix parameters arbitrarily to the chosen AF, eps is a smaller constant (Wang *et al.*, 2005) for view weights (vw) values:

$$IF(\delta(vw_i) - \delta(vw_i) < eps) \quad (34)$$

The above mentioned process is also similar for variable weights. The step direction is restructured for next iteration of weight calculation:

$$Step_{t+1} = \frac{\alpha.N-t}{N} step \quad (35)$$

where, α represents parameter to the maximum number of iterations and N indicates the number of clustering data samples, t represents current fishes and $t+1$ next fish (variable and view weights data samples) position. This process is continued until all, the multi-view clustering data samples is all concluded.

EXPERIMENTAL RESULTS AND DISCUSSION

In order to investigate the performance of the KICASDSTWC with incomplete view in classifying real-life data, selected three data sets from UCI Machine Learning Repository (Frank and Asuncion, 2010): They are:

- Multiple Features data set
- Internet Advertisement data set
- Image Segmentation data set

With these data, evaluated the performance of the proposed KICASDSTWC with incomplete view existing, against Quasi Newton's Subspace Two variable Weighted Clustering (QNSTWC), TW-k-means with four individual variable weighting clustering algorithms, i.e., EWKM (Tzortzis and Likas, 2010) and a weighted multi-view clustering algorithm WCMM (Jing *et al.*, 2007).

Characteristics of three real-life data sets: The Multiple Features (MF) data set includes 2,000 patterns of handwritten numerals that were obtained from a collection of Dutch utility maps. These patterns were segmented into 10 classes ("0"-"9"), each comprise 200 patterns. Each pattern was described by 649 characteristics that were segmented again into the following six views:

- **Mfeat-fou view:** Contains 76 Fourier coefficients of the character shapes
- **Mfeat-fac view:** Contains 216 profile correlations
- **Mfeat-kar view:** Contains 64 Karhunen-Love coefficients
- **Mfeat-pix view:** Contains 240 pixel averages in 2×3 windows
- **Mfeat-zer view:** Contains 47 Zernike moments
- **Mfeat-mor view:** Contains 6 morphological variables. At this point, use G_1, G_2, G_3, G_4, G_5 and G_6 to indicate the six views

The Internet Advertisement (IA) data set includes a collection of 3,279 images from various web pages that are classifiedd either as advertisements or non advertisements (i.e., two classes). The instances are expressed in six sets of 1,558 characteristics, which are the geometry of the images (width, height and aspect ratio), the phrases in the url of the pages includes the images (base url), the phrases of the images url (image url), the phrases in the url of the pages the images are directing at (target url), the anchor text and the text of the images alt (alternative) html tags (alt text). The entire views have binary characteristics, apart from the geometry view whose characteristics are uninterrupted.

The Image Segmentation (IS) data set includes 2,310 objects drawn arbitrarily from a database of seven outdoor images. The data set includes 19 characteristics which can be naturally segmented into two views:

- **Shape view:** Includes nine characteristics regarding the shape information of the seven images.
- **RGB view:** Includes 10 characteristics regarding the RGB values of the seven images.

The graphical representations of the clustering results for variable and view weights with different variables and the methods results are shown in Fig. 2. It shows the variation in variable weights for varying $\eta = 8$ values and view weights $w = 1$, $w = 32$ for TW-K means, QNSTWC and proposed KICASDSTWC with Incomplete View (ICV) $w = 1$, $w = 32$ results are shown in Fig. 2. It shows that proposed KICASDSTWC with Incomplete View (ICV) results shows have higher clustering accuracy with less view weights values are automatically calculated using AFSA, the proposed system also supports incomplete view of multi-view dataset, Centroid values are optimized using SDA.

The graphical representations of the clustering results for variable and view weights with different variables and the methods results are shown in Fig. 3 for Internet Advertisement (IA) dataset. It shows the variation in variable weights for varying $\eta = 8$ values and view weights $w = 1$, $w = 32$ for TW-K means and

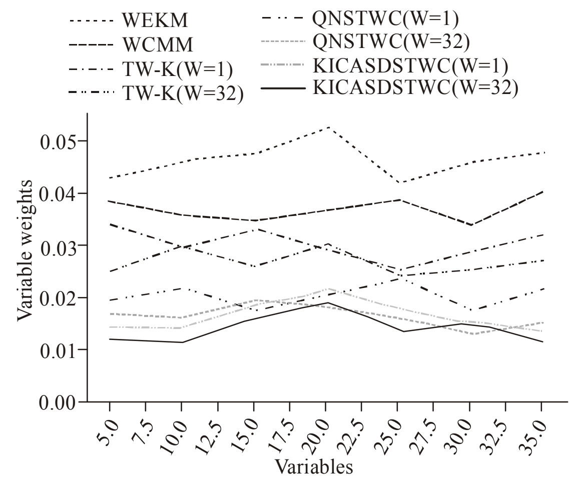

Fig. 2: Comparison of the total variable weights and view weights for methods in Multiple Features (MF) data set

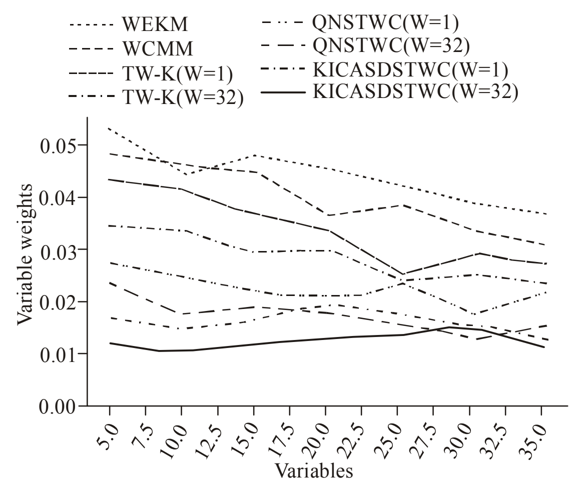

Fig. 3: Comparison of the total variable weights and view weights for methods in Internet Advertisement (IA) data set

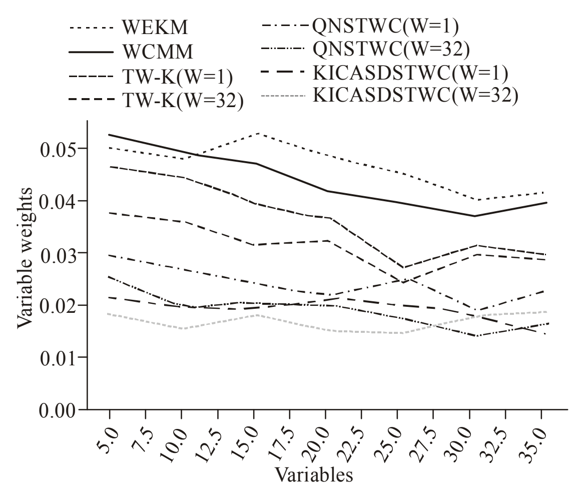

Fig. 4: Comparison of the total variable weights and view weights for methods in Image Segmentation (IS) dataset

Table 1: Summary of clustering results on three real-life data sets by five clustering algorithms

Dataset	Evaluation	EWKM	WCMM	TW-K means	QNSTWC	KICASDSTWC-ICV
MF	Precision	0.25	0.59	0.81	0.83	0.880
	Recall	0.39	0.58	0.83	0.85	0.870
	F measure	0.43	0.64	0.84	0.86	0.890
	Accuracy	0.39	0.62	0.86	0.87	0.895
	ACE	2.42	2.15	1.89	1.56	1.050
IA	Precision	0.26	0.59	0.74	0.76	0.810
	Recall	0.16	0.36	0.75	0.76	0.815
	F measure	0.23	0.49	0.72	0.73	0.820
	Accuracy	0.18	0.39	0.74	0.76	0.830
	ACE	2.12	1.98	1.56	1.05	0.980
IS	Precision	0.05	0.36	0.64	0.72	0.790
	Recall	0.06	0.43	0.65	0.71	0.760
	F measure	0.07	0.45	0.62	0.73	0.780
	Accuracy	0.09	0.42	0.65	0.71	0.790
	ACE	1.98	1.85	1.36	1.25	0.850

QNSTWC results are shown in Fig. 3. It shows that proposed KICASDSTWC with Incomplete View (ICV) $w = 1$, $w = 32$ results shows have higher clustering accuracy.

The graphical representations of the clustering results for variable and view weights with different variables and the methods results are shown in Fig. 4 for Image Segmentation (IS) dataset. It shows the variation in variable weights for varying $\eta = 8$ values and view weights $w = 1$, $w = 32$ for TW-K means, QNSTWC and KICASDSTWC with Incomplete View (ICV) results are shown in Fig. 4. It shows that proposed KICASDSTWC with Incomplete View (ICV) results shows have higher clustering accuracy with less view weights are taken for incomplete and complete view results.

In order to evaluate the clustering accuracy, this study uses Precision, Recall, F-measure, accuracy and average cluster entropy to evaluate the results.

Precision: Precision is computed as the fraction of accurate objects among those that the algorithm considers belonging to the relevant cluster.

Recall: Recall is the fraction of authentic objects that were recognized.

F-measure: F-measure is the harmonic mean of precision and recall and accuracy is the proportion of accurately clustered objects.

The results of the different clustering approaches with the above mentioned parameter results are shown in the Table 1. The performance comparison results of the proposed KICASDSTWC shows higher Precision, Recall, F measure and average accuracy, since the weight and centroid values are automatically calculated rather than using fixed values.

Average Cluster Entropy (ACE): depends on the contamination of a cluster given the true classes in the data. If p_{ij} represents the fraction of class j in obtained cluster i, N_i represents the size of cluster i and N indicates the total number of examples, subsequently the average cluster entropy is given as:

$$E = \sum_{i=1}^{K} \frac{N_i(-\sum_j p_{ij} \log(p_{ij}))}{N} \tag{36}$$

where, K represents the number of clusters.

CONCLUSION

In this study, proposed novel robust KICASDSTWC methods for complete and incomplete view of the data. In order to carry out multi-view data for incomplete data, new approach to ICA depending on kernel methods is given in this study. At the same time, most current ICA algorithms are depending on employing a single nonlinear function, this approach is a more flexible one in which candidate nonlinear functions of incomplete and complete view data are selected adaptively from a reproducing kernel Hilbert space. The proposed clustering is dissimilar to other existing approaches, because incomplete and complete view data are primarily learned then fuzzy centroid values are optimized using SDA. Given multiple-view data, calculate weights for views and individual variables concurrently with the help of AFSO. With the aim of reducing the complexity in the subspace clustering approach Singular Value Decomposition is proposed with ALCS methods for probability distribution optimization. The proposed system have been evaluated using three datasets namely Multiple Features (MF), Internet Advertisement (IA) and Image Segmentation (IS) based on the precision, recall, f measure and accuracy to analysis the properties of two types of weights. It shows that proposed KICASDSTWC achieves better clustering results than the existing clustering methods. Future works must concentrate on the ability to use isolated instances that do not have a corresponding multi-view representation to enhance learning and facilitate multi-view learning to be employed for a wider variety of applications.

REFERENCES

Bickel, S. and T. Scheffer, 2004. Multi-view clustering. Proceeding of the IEEE International Conference on Data Mining (ICDM). IEEE Computer Society, pp: 19-26.

Chaudhuri, K., S. Kakade, K. Livescu and K. Sridharan, 2009. Multi-view clustering via canonical correlation analysis. Proceedings of 26th Annual International Conference on Machine Learning, pp: 129-136.

Cheng, H., K.A. Hua and K. Vu, 2008. Constrained locally weighted clustering. Proceeding of the VLDB Endowment, 1: 90-101.

Cohn, D., R. Caruana and A.K. McCallum, 2009. Semi-supervised Clustering with User Multi-view Constrained Clustering with an Incomplete Mapping Between Views 27 Feedbacks. In: Basu, S., I. Davidson and K. Wagstaff (Eds.), Constrained Clustering: Advances in Algorithms, Theory and Applications. CRC Press, pp: 17-32.

Deng, Z., K. Choi, F. Chung and S. Wang, 2010. Enhanced soft subspace clustering integrating within-cluster and between cluster information. Pattern Recogn., 43(3): 767-781.

Dunlavy, D.M., T.G. Kolda and W.P. Kegelmeyer, 2006. Multilinear algebra for analyzing data with multiple linkages. Technical Report SAND 2006-2079, Sandia National Laboratories.

Frank, A. and A. Asuncion, 2010. UCI Machine Learning Repository. Retrieved from: http://archive.ics.uci.edu/ml.

Greene, D. and P. Cunningham, 2009. A matrix factorization approach for integrating multiple data views. Proceeding of the European Conference on Machine Learning and Knowledge Discovery in Databases, pp: 423-438.

Jiang, M.Y. and D.F. Yuan, 2005. Wavelet threshold optimization with artificial fish swarm algorithm. Proceeding of the IEEE International Conference on Neural Networks and Brain (ICNN&B’2005). Beijing, China, pp: 569-572.

Jing, L., M. Ng and Z. Huang, 2007. An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data. IEEE T. Knowl. Data En., 19(8): 1026-1041.

Kim, W., K.H. Lee and D. Lee, 2004. Fuzzy clustering of categorical data using fuzzy centroid. Pattern Recogn. Lett., 25(11):1263-1271.

Kolda, T.G. and B.W. Bader, 2009. Tensor decompositions and applications. SIAM Rev., 51(3): 455-500.

Long, B., P. Yu and Z. Zhang, 2008. A general model for multiple view unsupervised learning. Proceeding of the 8th SIAM International Conference on Data Mining (SDM ’08).

Nocedal, J. and S.J. Wright, 2006. Numerical Optimization. 2nd Edn., Springer, New York, 22: 664.

Sun, J., D. Tao and C. Faloutsos, 2006. Beyond streams and graphs: Dynamic tensor analysis. Proceeding of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp: 374-383.

Tang, W., Z. Lu and I.S. Dhillon, 2009. Clustering with multiple graphs. Proceeding of the 9th IEEE International Conference on Data Mining, pp: 1016-1021.

Tang, L., X. Wang and H. Liu, 2010. Community detection in multi-dimensional networks. Technical Report TR10-006, School of Computing, Informatics and Decision Systems Engineering, Arizona State University, Tempe, AS, USA.

Tzortzis, G.F. and C.L. Likas, 2010. Multiple view clustering using a weighted combination of exemplar-based mixture models. IEEE T. Neural Networ., 21(12): 1925-1938.

Wang, C.R., C.L. Zhou and J.W. Ma, 2005. An improved artificial fish-swarm algorithm and its application in feed-forward neural networks. Proceeding of the 4th International Conference on Machine Learning and Cybernetics, pp: 18-21.