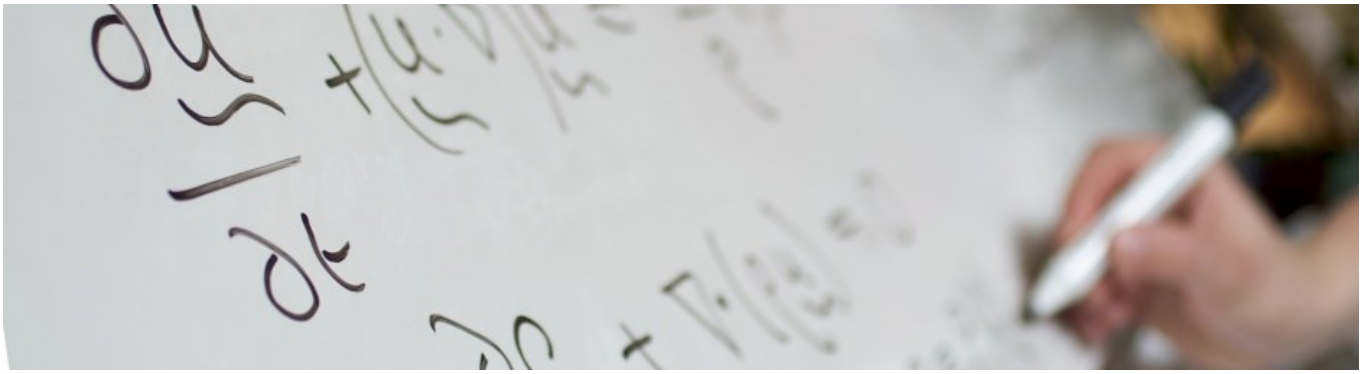
AN IMPROVED FUZZY C-MEANS CLUSTERING ALGORITHM FOR RBF NETWORKS



Original Research Article

ISSN : 2456-1045 (Online)
(ICV-MTS/Impact Value): 63.78
(GIF) Impact Factor: 4.126
Publishing Copyright @ International Journal Foundation
Journal Code: ARJMD/MTS/V-31.0/I-1/C-1/NOV-2018
Category : MATHEMATICAL SCIENCE
Volume : 31.0 / **Chapter**- I / **Issue** -1 (NOVEMBER-2018)
Journal Website: www.journalresearchijf.com
Paper Received: 13.11.2018
Paper Accepted: 23.11.2018
Date of Publication: 05-12-2018
Page: 01-05

Name of the Author (s):

Lim Eng Aik [1]*
Tan Wei Hong [2]
Ahmad Kadri Junoh[3]

[1, 3] Institut Matematik kejuruteraan, Universiti Malaysia Perlis, 02600 Ulu Pauh, Perlis, Malaysia

[2]School of Mechatronic Engineering, Universiti Malaysia Perlis, 02600 Ulu Pauh, Perlis. Malaysia

Citation of the Article

Aik LE; Hong TW; Junoh AK; (2018) An improved fuzzy C-Means clustering Algorithm for RBF networks.; *Advance Research Journal of Multidisciplinary Discoveries.31(1)pp. 01-05*

ABSTRACT

In fuzzy C-means (FCM) clustering, each data point belongs to a cluster to a degree specified by a membership grade. FCM partitions a collection of vectors in c fuzzy groups and finds a cluster center in each group such that the dissimilarity measure is minimized. This paper presented a training algorithm for the radial basis function (RBF) network using improved Fuzzy C-means (IFCM) clustering method which is the modified version of FCM clustering method based on weight readjustment for each attributed. The training algorithm which uses IFCM clustering method to train the network gain better accuracy in predictions and reduced network architecture compared to the standard RBF networks. The proposed training algorithm was implemented with RBF networks in MATLAB, therefore the new network will undergo a hybrid learning process. The networks called improved Fuzzy C-means Clustering–Radial Basis Function Network (IFCM/RBF) was tested against the standard RBF network and the networks called standard Fuzzy C-means Clustering-RBF network (FCM/RBF) in predictions. The experimental models were tested on three real world application problems, particularly in Air pollutant problem, Biochemical Oxygen Demand (BOD) problem, and Phytoplankton problem, which yield promising results.

KEYWORD: Fuzzy C-means clustering, Radial Basis Function Networks, Clustering, Prediction accuracy.

I. INTRODUCTION

Radial Basis Function (RBF) networks form a type of Artificial Neural Networks (ANNs), which has certain advantages over other kinds of ANNs, such as for instance better approximation capabilities, simpler network structures and faster learning algorithms. As a result of popularity of RBF networks, there are many researchers whom have been working to produce more effective training algorithms, set alongside the standard techniques (A. Alexandridis, Chondrodima, Giannopoulos, & Sarimveis, 2016; Alex Alexandridis, Chondrodima, Giannopoulos, & Sarimveis, 2017; da Silva, Maia, & Cabacinha, 2018; Hu, You, Liu, & He, 2018; Sarimveis, Alexandridis, & Bafas, 2003; Shan & Xu, 2017; Sun et al., 2013).

RBF networks are useful in approximation problems, but it requires quite a long time to teach the networks as it pertains to a huge number of training data, yet create a high error because of possible invalid data in the training data. Even though a combination of clustering methods in RBF networks has been proven by Sarimveis (Sarimveis et al., 2003) to be faster in training, it still produces a more substantial error. This is due to the standard clustering algorithms which still lack the ability to choose the most accurate and informative centers. By using IFCM clustering method, we are able to fix the problem stated above. As we know, the more accurate the centers chosen, the more accurate the information that feeds to the train network, this leads to more accurate result.

In this paper, a fast algorithm for training RBF networks which produce high accuracies is presented, which selects the input centers using the IFCM method. The speed of the presented method is due to the fact that it does not involve the formulation and solution of a nonlinear optimization problem, while it requires only one pass of the training data. The methodology is illustrated through the application of the experimental models by forecasting the pollutant trend at Forth Worth City, Texas with air quality data from Texas Resource Conservation Commission database, BOD concentration and Phytoplankton growth and death rates, both with data from Lim (Aik, 2006). The advantages of the presented learning strategy (IFCM/RBF) are identified and the results are compared with standard RBF networks and FCM/RBF.

II. IMPROVED FUZZY C-MEANS CLUSTERING ALGORITHM (IFCM)

The improved fuzzy C-means (IFCM) algorithm is based on the attribute weight assignment with its weight learning is mainly based on the similarity between samples. Motivated by simplicity and easy-manipulation of similarity measure based on Euclidean distance (X. Wang, Wang, & Wang, 2004), the similarity measure $\rho_{ij}^{(w)}$ is defined as follows:

$$\rho_{ij}^{(w)} = \frac{1}{1+\beta * d_{ij}^{(w)}} \quad \text{..... (1)}$$

Since similarity measure (1) is associated with the weighted Euclidean distance, it has well analytic properties and intuitive meaning (X. Wang et al., 2004). The value of similarity measure $\rho_{ij}^{(w)}$ is referred as the similarity degree. When w = (1,1,…,1), the similarity degree $\rho_{ij}^{(1)}$ is uniform distributed in [0,1]. However, most real data sets may not meet the requirement of uniformly distribution in [0,1]. To adjust the mean of the distribution of ρ_{ij}, the positive parameter β (>0) is used. Remarking that 0.5 is the mean of the uniform distribution in [0,1], hence it is preferable to select a β such that:

$$\frac{2}{n(n-1)}\sum_{j<i}\frac{1}{1+\beta * d_{ij}} = 0.5 \quad \text{..... (2)}$$

where d_{ij} is the commonly used Euclidean distance, and $d_{ij}^{(w)}$ is the weighted Euclidean distance defined as follows:

$$d_{ij}^{(w)} = \sqrt{\left(\sum_{k=1,\ldots,S} w_k^2 (x_{ik} - x_{jk})^2\right)} \quad \text{..... (3)}$$

where $w = (w_1, w_2, \ldots, w_S)$ is the attribute weight vector. Its component is the important degree corresponding to each attribute. The larger the w_k, the more important the k-th attribute is in FCM. When $w = (1,\ldots,1)$, the space $\{\|d_{ij}^{(w)} \le r\|\}$ is a hyper-sphere with radius r in the well-known Euclidean space. In the Euclidean space, $d_{ij}^{(w)}$ is denoted by d_{ij} and $\rho_{ij}^{(w)}$ by ρ_{ij}. When $w \neq (1,\ldots,1)$, it means that the axes would be extended or contracted in accordance with w_k. The space $\{\|d_{ij}^{(w)} \le r\|\}$ is hyper-ellipse and is called the transformed space. The more lower the value of w_k, the broader the flattening extent.

According to De Luca and Termini (1972) (De Luca & Termini, 1972), the fuzziness of similarity degrees $\{\rho_{ij} \mid i < j\}$ can be defined as

$$Fuzziness = \frac{-2}{n(n-1)}\sum\left(\rho_{ij}\log\rho_{ij} + (1-\rho_{ij})\log(1-\rho_{ij})\right) \quad \text{.....(4)}$$

It is clear that fuzziness shown in equation (4) attains its maximum when all similarity degrees are close to 0.5. It will attain its minimum when all similarity degrees are close to either 0 or 1 (W. Wang, Wang, Cui, & Wang, 2008). A good partition should have the following properties; the samples within one cluster are close to the centre and different centres are more separate, which implies that the samples within one cluster are more similar, i.e. $\rho_{ij}^{(w)} \to 1$. Dissimilar samples are more separate, i.e. $\rho_{ij}^{(w)} \to 0$, so that the fuzziness given in equation (4) is low. The hope is that by adjusting w, similar objects ($\rho_{ij} > 0.5$) in the Euclidean space are more similar ($\rho_{ij}^{(w)} \to 1$) in the transformed space, and the dissimilar objects ($\rho_{ij} < 0.5$) in the Euclidean space are more separate ($\rho_{ij}^{(w)} \to 0$) in the transformed space.

Based on the above discussion, notice that the learning of attribute-weight value can be done by minimizing an evaluation function $E(w)$ first introduced in Basak (1998) (Basak, De, & Pal, 1998) and then applied to clustering performance improvement (Yeung & Wang, 2002). $E(w)$ is defined as:

$$E(w) = \frac{2}{n(n-1)}\sum_i\sum_{j\neq i}\frac{1}{2}\left(\rho_{ij}^{(w)}(1-\rho_{ij}) + \rho_{ij}(1-\rho_{ij}^{(w)})\right) \quad \text{....(5)}$$

The gradient descent technique can be used to minimize $E(w)$. Let Δw_k be the change of w_k, compute as follows:

$$\Delta w_k = -\eta \frac{\partial E(w)}{\partial w_k} (0 < k < s) \quad \text{..... (6)}$$

For the procedure and related details, one can refer to Yeung and Wang (2002) (Yeung & Wang, 2002).

After obtaining attribute-weight values by the above learning, the weighted Euclidean distance can be used to replace the common Euclidean distance in FCM. In this way, the objective function $J^{(w)}$ given as follows:

$$J(U, c_1, ..., c_C) = \sum_{i=1}^{C} \sum_{j=1}^{n} u_{ij}^m d_{ij}^2 \quad \text{..... (7)}$$

then become the following:

$$J^{(w)}(U, c_1, ..., c_C) = \sum_{i=1}^{C} \sum_{j=1}^{n} u_{ij}^m (d_{ij}^{(w)})^2 \quad \text{..... (8)}$$

Minimizing equation (8) subject to

$$\sum_{i=1}^{C} u_{ij} = 1, \forall j = 1, ..., n. \quad \text{..... (9)}$$

then obtain u_{ij} and c_i as follows:

$$c_i = \frac{\sum_{j=1}^{n} u_{ij}^m x_j}{\sum_{j=1}^{n} u_{ij}^m} \quad \text{..... (10)}$$

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(d_{ij}^{(w)} / d_{kj}^{(w)} \right)^2} \quad \text{..... (11)}$$

The other parts of the algorithm are the same as FCM and are described in section 3.

The only difference between FCM and IFCM is the different of their respective distance matrix. Similar as FCM, the IFCM algorithm also input all samples and outputs the cluster centres and the partition matrix $U^{(w)}$.

III. THE PROPOSED IFCM/RBF TRAINING METHOD

The RBF network can be considered as a three layer network. The input nodes pass the input values to the connection arcs. The internal units form a single layer of *L*-RBF nodes, where the Gaussian function was used in this layer that localized response functions in the input space. The hidden node responses are weighted and the output nodes are simple summations of the weighted responses. The formulation of the training algorithm involves a set of input-output pairs [*x(i), y(i)*], *i = 1,...,K*, where *x(i)* is the *N*-dimensional input vector, *y(i)* is the corresponding target or desired *M*-dimensional output vector and *K* is the number of training examples.

The set of input-output examples is the information base, which is used to determine the values of the unknown parameters, i.e. the hidden node centers and radii and the connection weights between the hidden and the output layer. An approach to simplify the original input data set is the crucial step in developing a successful RBF networks model. The innovation in this work is the proposed algorithm which is used for selecting the most significant input centers, based on the IFCM method. The rest of the network parameters are calculated using standard methods.

Note that all the u mentioned here onwards is the u from equation (11). The following is the algorithm for IFCM that would meet this objective:

Step 1: Fix *c* and *m*. Initialise *U* to some $U^{(1)}$. Select $\varepsilon > 0$ for stopping condition.

Step 2: Update midpoint values v_i for each cluster c_i.

Step 3: Compute the set $\mu_k = \{i : 1 \le i \le c : \|x_k - v_i\| = 0\}$, and update $U^{(\ell)}$ according to the following: if $\mu_k = \phi$, then $\mu_{ik} = 1 \Big/ \left[\sum_{j=1}^{c} \left(\|x_k - v_i\| / \|x_k - v_j\| \right)^{2/(m-1)} \right]$, otherwise $\mu_{ik} = 0, \forall i \notin \mu_k$ and $\sum_{i \in \mu_k} \mu_{ik} = 1$. ($\phi \in \{i : 1 \le i \le c : \|x_k - v_i\| = 0\}$).

Step 4: Stop if $J < \varepsilon$, where $J = \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^m \|x_k - v_i\|^2$, otherwise go to step 2.

In step 1, $c(\ge 1)$ is set to a fixed number of clusters. In the rule generation phase, each cluster will be the basis for one rule. Usually we keep *c* as small as possible in order to keep the number of rules within reasonable bounds. Further, the matrix $U = \langle u_{ik} \rangle$ is to be initialised. A crisp, and even random, partition of *X* into *c* subsets can be sufficient to provide a good starting point for the algorithm.

In step 2, midpoint values v_i are computed, and respective midpoints will of course move towards points with higher membership values in their clusters. Note that a midpoint can coincide with some x_λ. In such a case we will have $u_{i\lambda} = 1$, and then, for all $\xi \ne i$, we will have $u_{\xi\lambda} = 0$.

Step 3 is the core of the algorithm. There, membership values u_{ik} are updated. Note that we must distinguish between cases depending on whether or not midpoints coincide with data points. The variable ℓ denotes the iteration number.

In step 4, we compute the difference between present and previous matrices of membership values. If the stopping condition is met, we are done.

IV. RESULTS AND DISCUSSION

The proposed methodology was tested by forecasting the pollutant trend at Forth Worth City, Texas, BOD concentration and Phytoplankton growth rate and death rate. The experimental result for IFCM/RBF, standard RBF network and FCM/RBF on the pollutant trend at Forth Worth City, Texas, BOD concentration and Phytoplankton growth rate and death rate, will be shown at the end of this section. . For the pollutant trend at Forth Worth City problem, the training set consists of 480 sets of air data and the test set comprises of 72 sets of air data which both were taken from hourly air data. Meanwhile, for BOD concentration problem and Phytoplankton growth and death rates problem, both training set consists of 100 sets of data and the test set comprises of 100 sets of data.

The experiment used the *newrb* function because it represents the general form of a RBF network. Furthermore, the proposed clustering method was implemented by using MATLAB's function. Gaussian basis function was used for both networks with other parameters such as spread was set to default value, so that the performance of the proposed network can be evaluated effectively. Performance of IFCM/RBF, standard RBF network and FCM/RBF in this experiment was measured by comparing the computation time taken for training and the Root Mean Squared Error (RMSE) was used to measure how well both networks approximates the chosen functions and it is given as follow:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(ti - yi)^2}{n}}$$, where t is target output, yi is actual output, and n is total number of data.

The number of centers chose for both FCM clustering method and SFCM clustering method are based on Lim (Aik, 2006) where the taken number of centers can perform the best result compared to other number of centers. Here we chose the number of center for both clustering method as 417 for air pollutant problem, 53 for Phytoplankton problem and 13 for BOD problem.

The database of air quality monitored at Forth Worth city, Texas of United States, data from BOD and Phytoplankton data were selected to test the developed IFCM/RBF network model. For air pollutant problem, the type of pollutant monitored includes carbon monoxide, nitric oxide, nitrogen dioxide, ozone, and oxides of nitrogen. For experimental purposes, hourly updated air quality data which were obtained from Texas Natural Resource Conservation Commission's homepage were used to predict the trend of interested pollutants which are Nitric Oxide, Nitrogen Dioxide and Oxides of Nitrogen. While for Phytoplankton problem, growth rate and death rate were used as the interested values. As for the BOD problem, the BOD concentration was taken as the interested value.

Table I : Performance comparison for IFCM/RBF network, FCM/RBF network and standard RBF network prediction results for each pollutant in air pollutant problem.

Method	NO		NO_2		NOx	
	CPU time (s)	RMSE	CPU time (s)	RMSE	CPU time (s)	RMSE
FCM/RBF	51.609	0.4819	52.719	0.3741	51.828	0.4804
IFCM/RBF	51.407	0.2098	51.792	0.2532	50.823	0.4127
Standard RBF network	93.281	0.1875	93.890	0.3909	94.218	0.4776

Table II: Performance comparison for IFCM/RBF network, FCM/RBF network and Standard RBF network prediction results for Growth rate and Death rate in Phytoplankton problem.

Method	Growth rate, Gr		Death Rate, Dr	
	CPU time (s)	RMSE	CPU time (s)	RMSE
FCM/RBF	0.203	0.1297	0.203	0.1630
IFCM/RBF	0.203	0.0333	0.188	0.0380
Standard RBF network	1.031	0.5443	1.000	0.6106

Table III: Performance comparison for IFCM/RBF network, FCM/RBF network and standard RBF network prediction results for bod concentration in bod problem.

Method	BOD concentration	
	CPU time (s)	RMSE
FCM/RBF	0.094	0.4502
IFCM/RBF	0.078	0.4502
Standard RBF network	1.313	0.4642

Results from Table I and Table II showed that IFCM/RBF networks approximate the chosen functions very well and it outperform the Standard RBF network and FCM/RBF network in the experiments.

From Table I, IFCM/RBF network surpasses the standard RBF and FCM/RBF network in terms of accuracy, learning speed and the network architecture by using training set which consists only 417 centers compared to 480 centers because significant data were chosen as center successfully. This means that, it is possible to find a number of centers such that it will provide a network with reduced complexity, faster training time yet improved accuracy.

From Table II, result shows that IFCM/RBF network once again outperform both the standard RBF and FCM/RBF network in term of accuracy. Even using only half of total 100 centers, it was able to perform such satisfying result. Finally, result from Table III showed that IFCM/RBF network are able to show better training time compare with both FCM/RBF network and Standard RBF network. However, in term of accuracy, IFCM/RBF obtained similar results as FCM/RBF, but still outperformed standard RBF RMSE value.

Generally, both IFCM/RBF network and Standard RBF network performed well in the experiments. The IFCM/RBF network is superior in terms of learning speed and the architecture of the network but it requires a proper value of number of centers for determining the number of input centers.

Reduced number of training set take less computation time and it means shorter training time compared to actual number of training set. From table I, the computation time is reduced to about 44%.

From the results above, we may infer that smaller training sets would use less computation time but produce larger errors compared to actual training set, because there is a significant loss of information which is represented by the original data especially when we try to reduce it into smaller sizes. However, due to the IFCM clustering method able to provide more accurate center for the network, the accuracy and learning speed of the network improve and even outperform the standard RBF itself. Furthermore, a large training set does not guarantee desirable accuracies because it might contain much invalid data that could jeopardize the desired accuracy, not mentioning the size of network it would create and the time taken for training.

There is no denial on the learning speed of the IFCM/RBF network, but it comes with a hefty compensation for the accuracy if the proper value of number of centers is not selected. As the number of centers for the network becomes lesser and it results much simpler network architecture and faster training time. Although the three models provide good results, the network structure, learning speed and accuracy of the IFCM/RBF network is superior compared to the standard RBF network and FCM/RBF network.

V. CONCLUSION

Experiments and a real world problem were simulated in this paper, where we applied on a real case study on forecasting for air pollution problem. Performances of these networks were compared by using the training time and Root Mean Squared Error (RMSE) as criteria for performance measurement.

Results from the experiments and case study showed that the IFCM/RBF network is better than the standard RBF in the context of learning speed and network architecture. IFCM/RBF network also better than FCM/RBF in the context of accuracy. It is possible to improve the accuracy of the proposed network by using statistical methods to choose the best value of number of center to be used. As conclusion, the proposed network is far more superior to the standard RBF network when it comes to learning speed, network architecture and accuracy.

Since self-organized selection of centers which can be performed by clustering algorithms to select meaningful centers for the hidden nodes was used, it would be interesting if the network would be tested with noisy training data to verify the efficiency of the chosen clustering algorithm.

REFERENCE

[1]. **Aik, L. E. (2006).** Modified Clustering Algorithms for Radial Basis Function Networks. Universiti Sains Malaysia.

[2]. **Alexandridis, A., Chondrodima, E., Giannopoulos, N., & Sarimveis, H. (2016).** A Fast and Efficient Method for Training Categorical Radial Basis Function Networks. IEEE Transactions on Neural Networks and Learning Systems, 1–6. https://doi.org/10.1109/TNNLS.2016.2598722

[3]. **Alexandridis, A., Chondrodima, E., Giannopoulos, N., & Sarimveis, H. (2017).** A fast and efficient method for training categorical radial basis function networks. IEEE Transactions on Neural Networks and Learning Systems, 28(11), 2831–2836. https://doi.org/10.1109/TNNLS.2016.2598722

[4]. **Basak, J., De, R. K., & Pal, S. K. (1998).** Unsupervised feature selection using a neuro-fuzzy approach. Pattern Recognition Letters, 19(11), 997–1006. https://doi.org/10.1016/S0167-8655(98)00083-X

[5]. **da Silva, E. M., Maia, R. D., & Cabacinha, C. D. (2018).** Bee-inspired RBF network for volume estimation of individual trees. Computers and Electronics in Agriculture, 152. https://doi.org/10.1016/j.compag.2018.07.036

[6]. **De Luca, A., & Termini, S. (1972).** A definition of a nonprobabilistic entropy in the setting of fuzzy sets theory. Information and Control, 20(4), 301–312. https://doi.org/10.1016/S0019-9958(72)90199-4

[7]. **Hu, Y., You, J. J., Liu, J. N. K., & He, T. (2018).** An eigenvector based center selection for fast training scheme of RBFNN. Information Sciences, 428, 62–75. https://doi.org/10.1016/j.ins.2017.08.092

[8]. **Sarimveis, H., Alexandridis, A., & Bafas, G. (2003).** A fast training algorithm for RBF networks based on subtractive clustering. Neurocomputing, 51, 501–505. https://doi.org/10.1016/S0925-2312(03)00342-4

[9]. **Shan, D., & Xu, X. (2017).** Multi-label Learning Model Based on Multi-label Radial Basis Function Neural Network and Regularized Extreme Learning Machine. Moshi Shibie Yu Rengong Zhineng/Pattern Recognition and Artificial Intelligence, 30(9). https://doi.org/10.16451/j.cnki.issn1003-6059.201709007

[10]. **Sun, Y., Du, W., Feng, C., Liu, D., Lu, J., Cui, Q., … Song, Z. (2013).** The application of RBF neural network based on ant colony clustering algorithm to pressure sensor. Chinese Journal of Sensors and Actuators, 26(6). https://doi.org/10.3969/j.issn.1004-1699.2013.06.010

[11]. **Wang, W., Wang, C., Cui, X., & Wang, A. (2008).** Improving fuzzy C-means clustering based on adaptive weighting. In Proceedings - 5th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2008 (Vol. 1). https://doi.org/10.1109/FSKD.2008.160

[12]. **Wang, X., Wang, Y., & Wang, L. (2004).** Improving fuzzy c-means clustering based on feature-weight learning. Pattern Recognition Letters, 25(10), 1123–1132. https://doi.org/10.1016/j.patrec.2004.03.008

[13]. **Yeung, D. S., & Wang, X. Z. (2002).** Improving performance of similarity-based clustering by feature weight learning. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(4), 556–561. https://doi.org/10.1109/34.993562

*****