Criterion in selecting the clustering algorithm in Radial Basis Functional Link Nets

ANG SAU LOONG[1], ONG HONG CHOON[2] & LOW HENG CHIN[3]
Department of Mathematical Sciences
Universiti of Sains Malaysia
Universiti of Sains Malaysia, 11800, Pulau Pinang
MALAYSIA
Dellphin1223@yahoo.com

***Abstract:* -** K-means and k-median clustering algorithms can help in the selection of centres for the Radial Basis Functional Link Nets. Radial Basis Functional Link Nets is used to classify the data. In this paper, we will show the importance of knowing the skewness of the data in deciding to choose between k-means or k-median clustering algorithm in finding the centre of Radial Basis Functional Link Nets and we will also show that this initial selection criterion will result in the improvement of efficiency in terms of speed and accuracy in data classification. Two sets of real data are used to demonstrate our results.

Key-Words: - Radial Basis Functional Link Nets; Radial Basis Function Network; k-means; k-median; clustering; skewness.

1 Introduction

This paper shows the importance of calculating the skewness of data and in making a choice between k-means and k-median clustering method in the centre selection of the Radial Basis Functional Link Nets (RBFLN). We use the Mardia's skewness to obtain the information about the nature of the data [1]. If the data is skewed, it is better to select k-median as our clustering algorithm in centre selection for Radial Basis Functional Link Nets. However, if the data is almost symmetrical, then k-means will be a better choice. There are many clustering methods to find the centre of a Radial Basis Functional Link Nets like DBSCAN [2], Dynamic Clustering [3] and many other methods. However, among these clustering methods, k-means and k-median algorithms usually give the solution in the quickest and more efficient way. K-means algorithm is linked to the Radial Basis Function centre selection to enhance the performance [4].

The k-means algorithm self organizes to create partitions, which act as the clusters [5]. The data which fall into any cluster will be averaged and finally we will obtain the centre of the cluster. The k-median will do the same process but instead of averaging, we will use the median in finding the centre of every cluster [6].

By using these clustering methods, we can reduce the distances between input data and the centre of the Gaussian function. This will enable the sum of squared error to converge faster so that the optimum solution can be obtained once the sum of squared error exceeds the stopping criteria [7]. K-means and k-median algorithms also help in cutting down the number of input data that are out of the coverage of the feature space region by clustering them into suitable partitions.

Radial Basis Functional Link Nets used by Looney [8] was originated by Pao [9]. It had been proven to be more efficient than Radial Basis Function Network [10] and [11]. The core function in Radial Basis Functional Link Nets is the Gaussian function that had been proven as universal robust approximators [12]. By applying Radial Basis Function, we can use the network in function approximation, interpolation, classification and pattern recognition [13]. However, in this study, we use two real data sets in classification to determine the accuracy and speed in the performance results.

2 Methodology

In this study, we apply K-means and K-median clustering algorithms to obtain the centres of the hidden nodes according to the Mardia's multivariate skewness calculation. The most important property of Mardia's multivariate skewness calculation is that it can indicate the nature of the input data as to whether the input data are skewed or nearly symmetrical [14]. Thus, Mardia's multivariate skewness can be considered as one of the statistical methods that analyzes the distribution of the given multidimensional data. Under this circumstance, Mardia's multivariate skewness calculation is useful in getting the preprocess information on the input data.

Since our input data are mostly in multidimensional form, Mardia's multivariate skewness is used in determining the nature of distribution for the input data so that choice of the selection of centres between K-means or K-median clustering algorithm can be done objectively. Here, the measurement of Mardia's multivariate skewness gives the information about the spread of the input data based on the multivariate normality so we can select the clustering algorithm for finding the centre of radial basis functional link nets according to the results obtained.

In other words, the input data will be tested for normality to check whether the input data is skewed or symmetrical. If the data sets are highly skewed, K-median is sufficient to be chosen for computing the centres from the input data while if the data sets are symmetrical, K-means becomes the better choice.

The Mardia's multivariate skewness is defined as

$$skewness = 1 - probchi\ (\ kappa1,\ dfchi\ ) \qquad (1)$$

where *probchi* is the probability of chi square [15].

The first parameter, *kappa1* is defined as

$$kappa1 = n * Beta1hat / 6 \qquad (2)$$

where n is the number of dataset while *Beta1hat* is Mardia's sample skewness.

The second parameter *dfchi* is the degree of freedom for the chi square approximation of multivariate skewness.

$$dfchi = p * (\ p+1\ ) * (\ p + 2\ ) / 6 \qquad (3)$$

where p is the dimension of the dataset.

In k-means algorithm, we must decide the value of k to be used first. Since the k is priori knowledge, we will set it to be the same as the number of centres in the hidden nodes of Radial Basis Functional Link Nets. In order to get the best solution for k-means, it is better to run the k-means algorithm many times. This will increase the probability that the data will be classified to the nearest cluster and yield the optimum solution. In this study, the k-means algorithm is applied to run for 100 times and we will use the result as our centre of hidden nodes in Radial Basis Functional Link Nets.

Generally, the k-means function is defined as

$$J = \sum_{j=1}^{K} \sum_{i=1}^{N} \left\| x_i^{(j)} - c_j \right\| \qquad (4)$$

where $\left\| x_i^{(j)} - c_j \right\|$ is the distance between one of the data x_i and a centre of a cluster, c_j.

K-median uses the same formula as in Equation (4) but the difference is that instead of using averaging, it will take the median of the data to be the centre for each cluster. Radial Basis Functional Link Nets is a hybrid or extension of the Radial Basis Function Network. The basic idea for Radial Basis Function is that the feature space [0, 1] is covered with M overlapping circular hyper ball regions. In every region there exists a continuous Radial Basis Function that assumes its maximum value at the centre of the region but reduces in value to zero when away from it. In this study, we use Radial Basis Functional Link Nets to classify multidimensional data into their respective classes.

The structure of the Radial Basis Function Network consists of three layers. The first layer is the input layer. The second layer will be the hidden layer where the hidden neurons are located. The last layer will be the output layer. In this layer, the output is compared with the target values to form the value for the sum of squared error. We apply Radial Basis Function Network with Gaussian function because it can perform exact interpolation of a set of data points which is in a high dimensional space [16]. Exact interpolation is the process to map every input vector exactly to its corresponding target vector [17]. If the data sets which we feed into the neural networks are non-linear, it is better if the neural networks are able to perform a non-linear mapping which can transform a non-linear separable classification problem into a linear separable problem [18]. The linear separable problem is easier to be solved than the non-linear problem [18]. The Radial Basis Function Network which has the Gaussian function as the kernel function possesses the ability to perform the task.

Generally, Radial Basis Function is mapping functions, s, as shown below:

$$s : R^n \rightarrow R \qquad (5)$$

where n is the number of the dimension in the input.

From Equation (5), we observe that the multi-dimensional input R^n are transformed into single dimensional output, R.

When a set of data which consists of Q number of input vectors $\boldsymbol{x}^{(n)}$ where n=1,…, N with the corresponding outputs, $\boldsymbol{t}$, we aim to find a function $g(\mathbf{x})$ in such a way that

$$g(x^n) = t^{(n)} \qquad n = 1,..., N \qquad (6)$$

Radial Basis Function has the form of

$$g(x^n) = \sum_n w_n \phi(\|x - x^n\|) \qquad n = 1,..., N \qquad (7)$$

where each data point $\boldsymbol{x}^n$ belongs to a basis function and the basis function takes the form $\phi(\|x - x^n\|)$ where $\phi(\cdot)$ is the non-linear function such as the Gaussian function. $\|x - x^n\|$ is the distance between the input vectors, $\boldsymbol{x}^n$ from the centres of Gaussian function, $\boldsymbol{x}$. $\boldsymbol{w}_n$ is the weight in the Radial Basis Function [16].

By having the similar form as the generalized linear discriminant function, we can write the interpolation condition of Equation (6) and Equation (7) in matrix form as

$$\boldsymbol{\Phi w}=\boldsymbol{t} \qquad (8)$$

where $\boldsymbol{t} \equiv (\boldsymbol{t}^{(n)})$, $\boldsymbol{w} \equiv (\boldsymbol{w}_\mathbf{n})$, and the square matrix $\boldsymbol{\Phi}$ has element $\boldsymbol{\Phi}_{nn'} = \phi(\|x - x^n\|)$ [17]. If the inverse matrix for $\boldsymbol{\Phi}^{-1}$ exists, the solution to Equation (8) is given by

$$\boldsymbol{w}=\boldsymbol{\Phi}^{-1}\boldsymbol{t} \qquad (9)$$

If the data points are different for a large class of function $\phi(\bullet)$, the matrix $\boldsymbol{\Phi}$ is non singular which also means the matrix has an inverse. By setting the weights in Equation (7) to the values given in Equation (9), the function $g(\mathbf{x})$ represents a continuous differentiable surface that passes exactly through each data point [19].

In our study case, the numbers of hidden nodes M, will be determined heuristically depending on the results. If we use less M than the number of training sets, we will get an under training situation. If M is too large, the computation on-line of unknown vectors will be slower and will build up extraneous error. Therefore we need to choose the M with a larger set than the input data, and try to reduce or increase the M until the data are well classified.

An RBF defined on N-dimensional feature vectors $\boldsymbol{x}$ is

$$y_m^{(q)} = \exp\left[-\left\|x - v^{(m)}\right\|^2 / \left(2\delta_{(m)}{}^2\right)\right] \qquad m = 1,...,M \qquad (10)$$

where q is the dataset number, $\boldsymbol{x}$ is an input vector, $\boldsymbol{v}^{(m)}$ is the centre of m^{th} hidden nodes while the spread of the receptive field is δ. $y_{(m)}$ will be maximum if the distances between $\boldsymbol{x}$ and $\boldsymbol{v}^{(m)}$ equals to zero. The δ will determine how far the spread of the circular disk that covers the interest bounded region in the feature space [10].

The activation function used in Radial Basis Functional Link Nets is the Gaussian function. Once the input vector gets near to the centre, it will start to activate the Gaussian function, y = f(x). In Radial Basis Functional Link Nets, there are weights in between the input layer with hidden layer and also the hidden layer with the output layer. Both weights are adjusted in a steepest descent way. The input weights will influence the input vector to get closer to the centre while the output weights will move the output, $\boldsymbol{z}$ towards the target value, $\boldsymbol{t}$.

The input weights, $\boldsymbol{w}_n^{(j)}$ are updated as

$$w_{\text{nj}}^{k+1} = w_{\text{nj}}^{k} + (2\eta_1 / M)\sum_{(q=1,Q)}\left[\sum_{(j=1,J)}\left(t_j^{(q)} - z_j^{(q)}\right)\right]x_n^{(q)} \qquad (11)$$

where n is the input number, η_1 is the learning rate, Q is the numbers of dataset, $\boldsymbol{z}_j^{(q)}$ is the output, and k is the iteration number

The output weights, $\boldsymbol{u}_m^{(j)}$ are updated as

$$u_{\text{mj}}^{k+1} = u_{\text{mj}}^{k} + (2\eta_2 / M)\sum_{(q=1,Q)}\left[\sum_{(j=1,J)}\left(t_j^{(q)} - z_j^{(q)}\right)\right]y_m^{(q)} \qquad (12)$$

where η_2 is the learning rate.

The differences between the observed value, $\boldsymbol{z}$ and the target value, $\boldsymbol{t}$ will be calculated as the sum of squared error. Our aim is to obtain the minimum total sum squared of error. The output z and the total sum squared of error E are defined as

$$\boldsymbol{z}_j^{(q)} = (1/M)\left[\sum_{(m=1,M)}\boldsymbol{u}_{mj}\boldsymbol{y}_m^{(q)}\right] + (1/N)\left[\sum_{(n=1,N)}\boldsymbol{w}_{nj}\boldsymbol{x}_n^{(q)}\right] \qquad (13)$$

$$\text{E}=\sum_{(\text{q}=1,\text{Q})}\sum_{(\text{j}=1,\text{J})}\Big(\boldsymbol{t}_j^{(q)} - \{((1/M)\sum_{(m=1,M)}\boldsymbol{u}_{mj}\boldsymbol{y}_m^{(q)} + \\ = (1/N)\sum_{(n=1,N)}\boldsymbol{w}_{nj}\boldsymbol{x}_n^{(q)}\}\Big)^2 \qquad (14)$$

In the Radial Basis Functional Link Nets used, the spread parameter will be fixed as it will be easier to compare the results using k-means and k-median in finding the centres for the hidden nodes in the hidden layer. On the other hand, the centres obtained from different clustering methods are used as the centre of each hidden neuron to run when training the data.

3 Results

In this section, we show the criterion in selecting the clustering algorithm in Radial Basis Functional Link Nets based on two data sets. The nature of the data sets is identified and numbers of attribute are stated. We explain the classes for each data set.

Once we identified the multivariate skewness of each data set, then we decide which clustering algorithm to be the better choice for selection of centres in Radial Basis Functional Link Nets. Then, we feed the data sets in Radial Basis Functional Link Nets to show the performance of data training in term of accuracy and speed.

Basically, we separate the data sets into two subgroups which are the testing data set and the training data set. The training data sets are used to train the Radial Basis Functional Link Nets while the test sets are used to show the accuracy of the type of method being applied for the selection of centres.

To determine whether the data sets are correctly classified or wrongly classified, a rule is set. The strict rule is that any output values which fall between 0.4 and 0.6 is considered wrongly classified [8]. This rule gives a tolerance space for the output to be grouped into its class and the rule also filters out those output which do not belong to any class. By applying the rule, we can calculate the number of training misses in the Radial Basis Functional Link Nets when applying the different methods in selecting the centres.

Results are organized in tables form for comparison purposes. We show the results in the type of method used, number of iterations for the method used, and the testing error and training misses in Table 2, Table 3, Table 5 and Table 6.

Next, we then comment on the performances of the different types of method used in Radial Basis Function Network and based on the number of iterations using the sum of squared error and the number of training misses. The sum of squared error in each category for the number of iterations is discussed in terms of speed. The number of training misses is discussed to show the accuracy of the results obtained.

We summarize the results and give the overall performances of the training by Radial Basis Functional Link Nets. We also state the importance of Mardia's multivariate skewness measurement in determining the clustering method which is more suitable to be used in selection of centres for the Radial Basis Functional Link Nets.

The first input data is the geological oil exploratory data set and this data set consists of 4 dimensions [8]. Each dimension represents a feature and all the features analyzed will lead to the determination of which explorations to be made (for example, for oil or coal). There are 69 data points in the geological oil exploratory data set. The first ten data points will be used as the testing set. The rest we will put into the training set to see the results.

The second data set is the fitting contact lenses and this data set consists of 24 data points with four attributes [20]. The data set is noise free, highly simplified problem, and each data point is complete and correct. The application of this data can be found in [21]. The first attribute in the data set is the age of the patient while the second attribute is the spectacle prescription. The third attribute indicates whether the patient is astigmatic or non-astigmatic whereas the fourth attribute is the tear production rate of the patient. The patients are categorized into three classes and these classes are fitted with hard contact lenses, fitted with soft contact lenses and not fitted with contact lenses. The patient should be classified into one of the classes according to their attributes.

Table 1 The skewness measurement for datasets

Dataset	Skewness	
Geological Oil exploratory	Testing	0.5602374
	Training	1.7609×10^{-9}
Fitting contact lenses	Testing	0.9724819
	Training	1

Table 2 Results of training the data for testing based on the 10 data points in the Geological Oil Exploratory data set (skewed)

Type of methods used	No. of iterations	Testing error	Training misses
RBFLN with random centre selection	50	0.031670	0
	100	0.005951	0
	150	0.002375	0
	200	0.001426	0
	250	0.001399	0
	300	0.001382	0
	350	0.001370	0
	400	0.001358	0
	450	0.001348	0
	500	0.001338	0
RBFLN with k-means	50	0.044892	0
	100	0.004031	0
	150	0.002347	0
	200	0.001736	0
	250	0.001277	0
	300	0.001184	0
	350	0.001170	0
	400	0.001160	0
	450	0.001151	0
	500	0.001145	0
RBFLN with k-median	50	0.045567	0
	100	0.010315	0
	150	0.001278	0
	200	0.000885	0
	250	0.000804	0
	300	0.000738	0
	350	0.000686	0
	400	0.000645	0
	450	0.000631	0
	500	0.000619	0

Table 3 Results of training the data based on the 59 data points in the Geological Oil Exploratory data set (skewed)

Type of methods used	No. of iterations	Testing error	Training misses
RBFLN with random centre selection	1000	0.200779	0
	2000	0.189443	0
	3000	0.183701	0
	4000	0.180944	0
	5000	0.178725	0
	6000	0.176626	0
	7000	0.174618	0
	8000	0.172728	0
	9000	0.17089	0
	10000	0.169079	0
RBFLN with k-means	1000	0.242131	0
	2000	0.151816	0
	3000	0.110615	0
	4000	0.086841	0
	5000	0.072506	0
	6000	0.063342	0
	7000	0.055932	0
	8000	0.043138	0
	9000	0.040156	0
	10000	0.037558	0
RBFLN with k-median	1000	0.234793	0
	2000	0.073606	0
	3000	0.050089	0
	4000	0.039521	0
	5000	0.034819	0
	6000	0.031956	0
	7000	0.030061	0
	8000	0.028757	0
	9000	0.027824	0
	10000	0.027132	0

Fig. 1 Iteration versus total sum of squared error for geological oil exploratory testing data set (skewed)

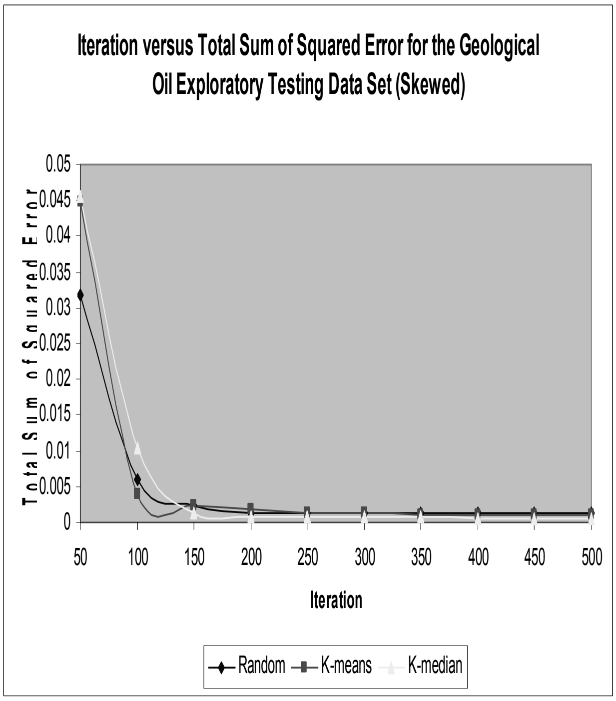


Fig. 2 Iteration versus total sum of squared error for Geological Oil Exploratory training data set (skewed)

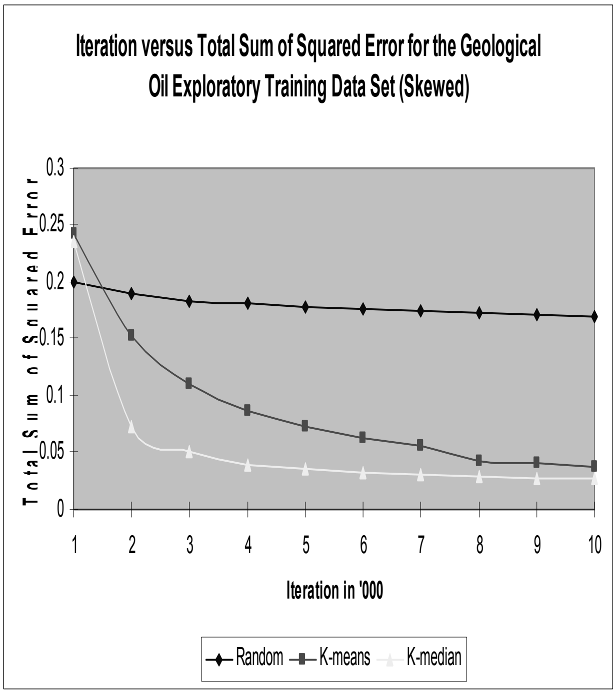


Table 4 Results of training the data for testing based on the 8 data points in the Fitting Contact Lenses data set (symmetrical)

Types of method used	No. of iterations	Testing error	Training misses
RBFLN with random centre selection	10	7.997714	0
	20	6.526225	0
	30	5.612273	0
	40	4.914061	0
	50	4.034773	0
	60	2.889179	0
	70	1.702626	0
	80	0.862680	0
	90	0.420995	0
	100	0.364993	0
RBFLN with k-means	10	8.115248	0
	20	6.257649	0
	30	4.317621	0
	40	2.512642	0
	50	1.137989	0
	60	0.295738	0
	70	0.038922	0
	80	0.008348	0
	90	0.001713	0
	100	0.001713	0
RBFLN with k-median	10	8.008216	0
	20	6.344485	0
	30	4.614012	0
	40	3.007452	0
	50	1.874816	0
	60	1.192491	0
	70	0.682117	0
	80	0.236776	0
	90	0.093433	0
	100	0.093433	0

Table 5 Results of training the data based on the 16 data points in the Fitting Contact Lenses data set (symmetrical).

Types of method used	No. of iterations	Testing error	Training misses
RBFLN with random centre selection	1000	0.777761	0
	2000	0.473322	0
	3000	0.35261	0
	4000	0.275632	0
	5000	0.218692	0
	6000	0.174407	0
	7000	0.139959	0
	8000	0.112778	0
	9000	0.091528	0
	10000	0.074737	0
RBFLN with k-means	1000	0.180488	0
	2000	0.088691	0
	3000	0.055737	0
	4000	0.039372	0
	5000	0.029473	0
	6000	0.023000	0
	7000	0.018468	0
	8000	0.015229	0
	9000	0.012779	0
	10000	0.010871	0
RBFLN with k-median	1000	0.246964	0
	2000	0.206018	0
	3000	0.174594	0
	4000	0.14853	0
	5000	0.126414	0
	6000	0.107656	0
	7000	0.091783	0
	8000	0.07824	0
	9000	0.066707	0
	10000	0.056843	0

Fig. 3 Iteration versus total sum of squared error for the Fitting Contact Lenses testing data set (symmetrical)

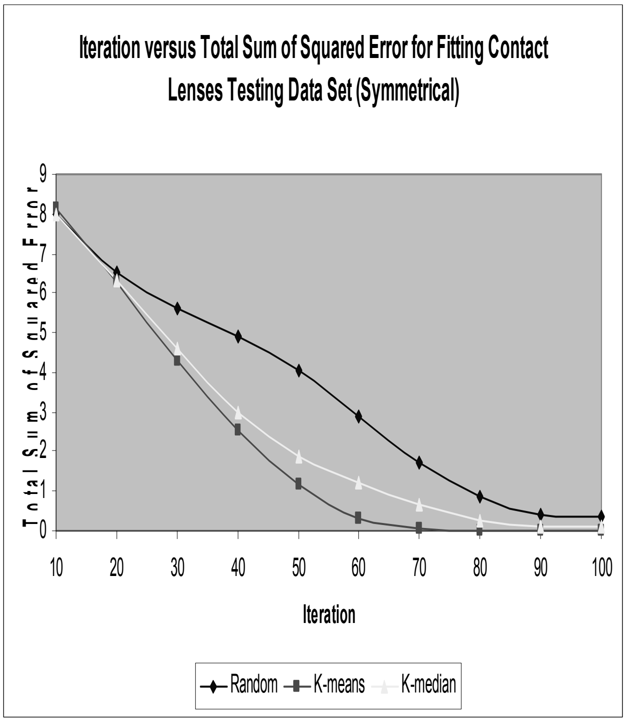
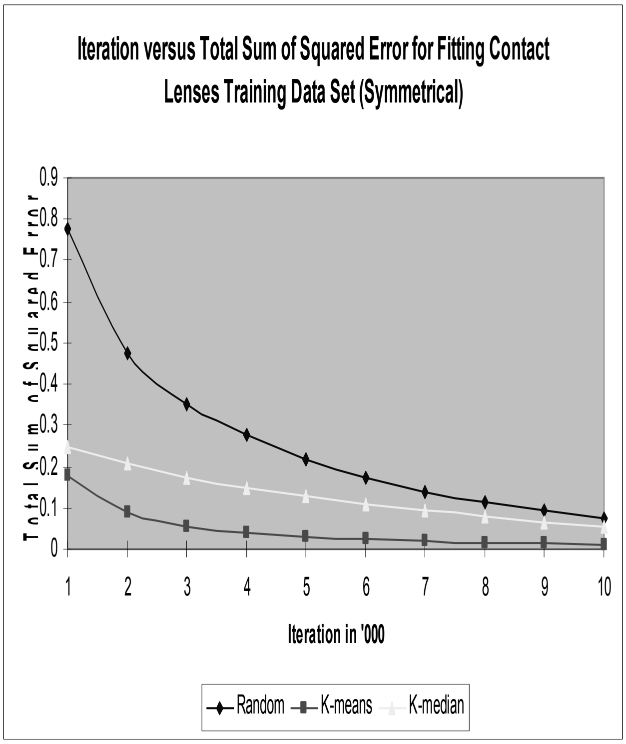


Fig. 4 Iteration versus total sum of squared error for the Fitting Contact Lenses training data set (symmetrical)



4 Discussions

Table 1 shows the value of skewness for both the data sets, namely the oil exploratory data and the fitting contact lenses data. The skewness values range from 0 to 1. If the skewness value is near to 1, the data is almost symmetrical but if the values turn out to be closer to 0, this means the data set is skewed. It is obvious that the the skewness of the datasets has its influence in deciding which clustering method is best used to find the centres.

For the geological oil exploratory testing data set, the skewness is 0.5602374 which is almost in the middle of 0 and 1. Since the data set is only used for testing purpose, therefore we apply both clustering algorithms to verify which one is better to be used for selection of the Gaussian centres. For the training data set of the geological oil exploratory, the value of skewness is 1.7609×10^{-9} and it is almost 0. Theoretically, the data set is highly skewed and K-median will be better in selecting the Gaussian centres. Besides, the fitting contact lenses testing data set takes 0.9724819 as the value of the skewness. That means the testing data set is more symmetrical. For the fitting contact lenses training data set, the skewness is 1 and the data set is symmetrical. When the data set is more symmetrical as in the case of the fitting contact lenses data set, k-means algorithm is the best choice for selecting the Gaussian centre.

Table 2, Table 3, Table 4 and Table 5 show the effect of the preprocess calculation on skewness in selecting the clustering algorithm. For the observations in the testing data based on geological oil exploratory dataset and fitting contact lenses data set, the total sum of squared error reduces very quickly mostly because the data set to be trained is small. We set the stopping criterion to be 0.0001 to halt the training of each data set. The value of the stopping criterion is small which also means that the sum of squared error reach a level where the output values are close to the target values and we stop the training process of data classification for result evaluation. If the sum of squared error is less than 0.0001, then the iteration will stop immediately and our aim is achieved. Overall, the sum of squared error of the training converges in speed. Accuracy of the training is not only based on total sum of squared error but also based on the number of training data missed. We set a strict tolerance to determine the output data that will be considered misclassified or training miss.

Table 2 shows that k-median is the best among the three methods after 100 iterations. This is then followed by K-means and the random selection method. We can also see the comparative performance of these methods in Fig. 1. The sum of squared error of the test data set using three different methods converges in speed. Besides, we found that there is no training miss for the two data sets due to the efficiency of the powerful tool, Radial Basis Functional Link Nets, being used. Therefore, we will justify which clustering method to use based on the results that achieve the lowest sum of squared error. Applications of K-means and K-median clustering clearly improve the performance of the Radial Basis Functional Link Nets but the choice of whether to use K-means or K-median depends on the skewness of the data.

By the way, we can see consistent result in the geological oil exploratory training for both testing and training data set. We observe that k-median converge fastest in Table 3 and this is shown in Fig. 2. As the data is much skewed as in the case of the geological oil exploratory data in Table 3, K-median outperforms other methods such as K-means and random selection. Table 2 and 3 also show that for both testing and training data, K-means performs better than random selection of centres because K-means reduces the number of outliers in our input data so that each output has a better chance to get closer to the centre of the Gaussian function.

Table 4 shows that the k-means has the lowest sum of squared error after 10 iterations in the run for testing among the three methods. All the methods have no training miss and the trend of the total sum of squared error is shown in Fig. 3. Table 5 shows that k-means outperforms other methods for a more symmetrical data set. Therefore, if the data is more symmetrical with the skewness of Mardia near to 1 as in the fitting contact lenses data set, k-means will perform better (Fig. 4).

We have reduced the distances between input data from the centre of the Gaussian Function in both models by using these clustering methods so that the sum of squared error converges faster and the optimum solution can be obtained once the sum of squared error exceeds the stopping criteria. K-means and K-median algorithms also help in reducing the number of input data that are out of the coverage feature space region by clustering them into suitable partitions. In other words, the outlier can be reduced and this will certainly increase the power of training speed and accuracy. If we use random centres for the input vectors, we might lose the best position for the centres which represent the distribution of input data.

5 Conclusion

Application of k-means and k-median clustering clearly improve the performance of the Radial Basis Functional Link Nets in terms of speed and accuracy but the choice of whether to use k-means or k-median depends on the skewness of the data**.** Since the preprocessing of the input data using Mardia's multivariate skewness can show the details of the input data, we can choose the clustering algorithm according to the type of data. By using the clustering algorithm determined by Mardia's multivariate skewness in selection of centres of the Gaussian function, we can see the difference in the performance. K-means are better than K-median for symmetrical data while K-median outperforms the former when the given input data is skewed. Both algorithms are useful in helping the Radial Basis Functional Link Nets in shortening the training iteration and improving the accuracy of the outcomes.

As the initial selection of centres have great impact on the performance of the neural networks, the choice of the different types of clustering algorithms become a very important issue. The measure of skewness is just one of the tools to determine the selection between K-means and K-median but it may not be relevant if we use other clustering methods.

Hence, our future research is to find out more relationships between the input data and properties of the clustering algorithms so that we are able to make a big step forward in improving the performance of Radial Basis Functional Link Nets. Besides, there is still room to improve the efficiency of Radial Basis Functional Link Nets. The determination of the optimum number of hidden nodes, M, in the Radial Basis Functional Link Nets remains a challenge. The type of activation functions in the radial functional link nets can also be investigated for the possibility to be replaced by a more stable and efficient function which is able to boost the performance of the neural network.

Furthermore, several important parameters such as the spreads and the weights which lie in between the layers can be studied to create more advanced Radial Basis Functional Link Nets. The spreads which control the size of the circular disks covered on the feature space of interest are important to make sure all the input data are under their influence. The ranges of the initial weights are also important so that a set of efficient initial weights can be obtained to increase the speed of training.

Acknowledgements

This work was supported in part by U.S.M. Fundamental Research Grant Scheme (FRGS) no. 203/PMATHS/671041.

References:

[1] K. V. Mardia, Measures of multivariate skewness and kurtosis with applications, *Biometrika*, Vol.57, No.3, 1970, pp. 519-530.

[2] M. Ester, H. P. Kriegel, J. Sander, and X. W. Xu, A density-based algorithm for discovering clusters in large spatial database with noise, *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96),* 1996, pp. 226-231.

[3] J. T. Tou, Dynoc-a dynamic optimal cluster seeking technique, *International Journal of Computing and Information Sciences*, Vol.8, 1974, pp. 541-547.

[4] A. Cichocki and R. Unbehauen, *Neural networks for optimization and signal processing*, John Wiley and Sons*,* New York, 1993.

[5] J. Macqueen, Some methods for classification and analysis of multivariate data, *Proceedings of the 5th Berkeley Symposium on Probability and Statistics* Vol.1, 1967, pp. 281-297, University of California, Berkeley.

[6] L. Mico and J. Oncina, An approximate median search algorithm in non-metric spaces, *Pattern Recognition Letters*, Vol.22**,** 2001, pp 1145-1151.

[7] J. K. Sing, D. K. Basu and M. Nasipuri, Improved k-means algorithm in the design of RBF neural networks, *Proceedings of the Conference on Convergent Technologies for Asia-Pasific Region*, *Tencon,* Vol.2, 2003, pp. 841-845, Bangalore, India.

[8] C. G. Looney, Radial Basis Functional Link Nets and fuzzy reasoning, *Neurocomputing* Vol.48, No.(1-4), 2002, pp. 489-509.

[9] Y. H. Pao, G. H. Park and D. J. Sobajic, Learning and generalization characteristic of the Random Vector Functional Link Net, *Neurocomputing*, Vol.6**,** 1994, pp. 163-180.

[10] C. G. Looney, *Pattern Recognition using Neural Networks*, Oxford University Press, New York, 1997.

[11] S. L. Ang, H. C. Ong and H. C. Low, Comparison between modified Radial Basis Functional Link Nets and Radial Basis Function, Proceedings of the 2nd Indonesia-Malaysia-Thailand Growth Triangle (IMT-GT), Regional Conference on Mathematics,

Statistics and Applications, Vol.4, 2006, pp. 120-125.

[12] J. T. H. Lo, Multilayer perceptrons and Radial Basis Function are universal robust approximators, *Neural Networks Proceedings, IEEE World Congress on Computational Intelligence*, Vol.2, 1998, pp. 1311-1314.

[13] M. Sgarbi, V. Cola and L. M. Reyneri, A comparison between weighted Radial Basis Functions and Wavelet Networks, *Proceedings of the 6th European Symposium on Artificial Neural Network*, 1998, pp. 13-19, Bruges, Belgium.

[14] K. V. Mardia, Assessment of Multinormality and the Robustness of Hotelling's T2 Test, *Applied Statistics*, Vol.24, No.2, 1975, pp. 163-171.

[15] R. Khattree and D. N. Naik, *Applied Multivariate Statistics with SAS Software*, SAS Institute Incorporated, Cary, North Carolina, USA, 1995.

[16] M.J.D. Powell, Radial Basis Functions for Multivariable Interpolation: A Review, Clarendon Press, Oxford, 1987.

[17] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, New York, 1995.

[18] S. Haykin, *Neural Networks. A Comprehensive Foundation,* Prentice Hall, New Jersey, 1999.

[19] C. A. Micchelli, Interpolation of Scattered Data: Distances Matrices and Conditionally Positive Definite Functions, *Constructive Approximation*, Vol.2, 1986, pp. 11-22.

[20] C. Blake, Repository of machine learning databases, University of California, Irvine, Downloadable at http://www.ics.uci.edu/~mlearn/MLREpository.html., 1998.

[21] I. H. Witten and B. A. MacDonald, Using Concept Learning for Knowledge Acquisition, *International Journal of Man-Machine Studies*, Vol.27, 1988, pp. 349-370.