Kasetsart J. (Nat. Sci.) 43 : 378 - 388 (2009)

Comparison of Clustering Techniques for Cluster Analysis

Piyatida Rujasiri* and Boonorm Chomtee

ABSTRACT

Cluster analysis is important for analyzing the number of clusters of natural data in several domains. Various clustering methods have been proposed. However, it is very difficult to choose the method best suited to the type of data. Therefore, the objective of this research was to compare the effectiveness of five clustering techniques with multivariate data. The techniques were: hierarchical clustering method; K-means clustering algorithm; Kohonen's Self-Organizing Maps method (SOM); K-medoids method; and K-medoids method integrated with Dynamic Time Warping distance measure (DTW). To evaluate these five techniques, the root mean square standard deviation (RMSSTD) and r^2 (RS) were used. For RMSSTD, a lower value indicates a better technique and for RS, a higher value indicates a better technique. These approaches were evaluated using both real and simulated data which were multivariate normally distributed. Each dataset was generated by a Monte Carlo technique with 100 sample sizes and repeated 1,000 times for 3, 5 and 7 variables. In this research, 2, 3, 4, 5, 6, 7 and 8 clusters were studied. Both real and simulated datasets provided the same result, with the K-means clustering method having the closest RMSSTD and RS results to the SOM method. These two methods yielded the lowest RMSSTD and highest RS in all simulations. Hence, both K-means and SOM were considered to be the most suitable techniques for cluster analysis.

Key words: cluster analysis, multivariate data, Kohonen's Self-Organizing Maps, K-medoids, Dynamic Time Warping, K-means

INTRODUCTION

Statistical methods are used in current research in several domains including: social sciences, management, medicine, agriculture and other sciences (Arms and Arms, 1978; David *et al.*, 1996; Clatworthy *et al.*, 2005). Almost all research needs to collect large amounts of data and manage it systematically in order to analyze processes or systems. Data clustering is one of the important analytical techniques and will become increasingly useful in the future, for visualizing data and searching for hidden trends in the data.

Cluster analysis is a class of statistical technique used to separate data into appropriate groups. It is most important in unsupervised learning problems since these techniques deal with finding structure in a collection of unlabeled data. Clustering algorithms can be divided into two types: hierarchical algorithms and partitional algorithms (Jardine and Sibson, 1968). Hierarchical algorithms, such as hierarchical clustering, begin with matching each object with similar ones that are placed in a separate cluster and then merged into larger clusters. On the other hand, partitional algorithms, such as K-means

Department of Statistics, Faculty of Science, Kasetsart University, Bangkok 10900, Thailand.
* Corresponding author, e-mail: kittylovely555@hotmail.com

Received date : 15/12/08 Accepted date : 16/03/09

clustering, classify the whole object into smaller clusters. Many researchers have proposed other clustering techniques for various data.

Kaufman and Rousseeuw (1990) considered it can be a challenging problem to choose the best clustering algorithm from the many available. Therefore, the main purpose of this work was to compare the effectiveness of five techniques: the hierarchical clustering method (Johnson, 1967); the K-means clustering algorithm (Hartigan and Wong, 1979); Kohonen's Self-Organizing Maps Method (SOM) (Kohonen, 1988); K-medoids method (Sheng and Liu, 2006); and K-medoids method together with an integration of Dynamic Time Warping distance measure (DTW) (Nienattrakul and Ratanamahata, 2006). These approaches were tested using simulated data. Each of the datasets had 3, 5 and 7 variables.

In addition, this work particularly focused on real datasets to verify the results, which could give researchers additional confidence in the results of this paper.

MATERIALS AND METHODS

The materials used in this work consisted of the following:

1. Computer: Intel Core 2 Duo 1.66 GHz, 1.50 Gb. of RAM
2. Microsoft Windows XP SP2
3. MATLAB version 7.0

The methodology of this paper is shown in Figure 1.

The methodology was separated into three parts, involving collection of datasets, testing the methods and evaluating the results. The first step of collecting datasets involved using two sources: simulated datasets and real datasets.

The real dataset (Boonmung *et al*., 2006) was based on pineapple data, with a sample size of 149 pineapples. Three independent variables were measured for each pineapple: average firmness, brix and resonant frequency. The pineapples were classified into two groups: immature and mature, based on their internal color and external qualities. The classification resulted in 95 and 54 pineapple samples in group1 (immature) and group 2 (mature), respectively. Some sample pineapple data is shown in Table 1.

Each sample consisted of measurements of resonant frequency (Freq), average firmness (Firmness) and soluble solids content (Brix %).

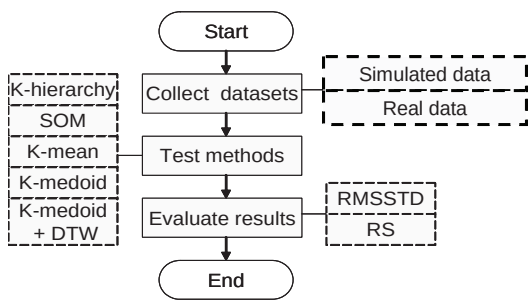

Figure 1 Organization of the methodology used.

Table 1 Some samples of the pineapple dataset.

No.	Group	Freq	Firmness	Brix (%)
1	1	420.141	2291.83	13.48
2	1	453.130	3418.80	11.74
3	1	454.864	2048.75	12.16
.	.	.	.	.
.	.	.	.	.
147	2	251.738	625.23	10.74
148	2	239.583	318.37	15.1
149	2	222.223	281.93	14.22

Freq = resonant frequency; Firmness = average firmness; Brix (%) = soluble solids content.

The second sample dataset consisted of multivariate normally distributed data (Morrison, 2002) generated from the function:

$$f(\mathrm{X}) = \frac{1}{(2\pi)^{p/2}|\sum|^{1/2}} \cdot \exp\left(-\frac{1}{2}(\mathrm{X}-\mu)^T \sum^{-1}(X-\mu)\right)$$

where p is the number of independent variables,
μ is a mean vector of size $p \times 1$,
Σ is a covariance matrix of size $p \times p$,

Each multivariate normally distributed dataset was generated 1,000 times for 3, 5 and 7 variables, with each having 100 samples. The number of variables (3, 5 and 7 variables) was varied using the following parameters:

For 3 variables, μ = [1000 2000 3000] and $\Sigma = \begin{bmatrix} 300 & 350 & 400 \\ 350 & 600 & 700 \\ 400 & 700 & 1400 \end{bmatrix}$.

For 5 variables, μ = [1000 2000 3000 4000 5000] and $\Sigma = \begin{bmatrix} 30 & 35 & 40 & 48 & 52 \\ 35 & 60 & 75 & 81 & 96 \\ 40 & 75 & 140 & 156 & 187 \\ 48 & 81 & 156 & 230 & 290 \\ 52 & 96 & 187 & 290 & 400 \end{bmatrix}$.

For 7 variables, μ = [100 200 300 400 500 600 700] and $\Sigma = \begin{bmatrix} 2 & 3 & 4 & 5 & 7 & 9 & 10 \\ 3 & 12 & 13 & 15 & 17 & 19 & 21 \\ 4 & 13 & 25 & 27 & 29 & 32 & 34 \\ 5 & 15 & 27 & 38 & 40 & 43 & 47 \\ 7 & 17 & 29 & 40 & 51 & 53 & 58 \\ 9 & 19 & 32 & 43 & 53 & 64 & 69 \\ 10 & 21 & 34 & 47 & 58 & 69 & 75 \end{bmatrix}$.

Before running the clustering algorithm both the simulated and real datasets were pre-processed using Z-normalization which is defined by Equations 1,2 and 3:

$$Z - Normalzation = \frac{x_i - \bar{x}}{\hat{\sigma}_x} \quad (1)$$

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (2)$$

$$\hat{\sigma}_x = \sum_{i=1}^{n}\sqrt{\frac{(x_i - \bar{x})^2}{n-1}} \quad (3)$$

where x_i is the data point in position i^{th},
$\bar{x}$ is the average value from x_i through x_n,
$\hat{\sigma}_x$ is the standard deviation of x_i.

After producing all datasets, the five clustering methods were applied to these data.

Hierarchical clustering

The hierarchical clustering method, a classical clustering method, has been used for a long time.

This method defines each of the objects in a dataset as a cluster and tries to merge them into larger clusters by constructing a hierarchical tree or dendrogram. The method applied in this algorithm is shown in Table 2.

From step 1 of Table 2, the similarity or dissimilarity of a pair of data was investigated in order to build the larger cluster in the hierarchical tree by using the four criteria (Single, Average, Complete and Ward) shown in Table 3.
where a is an object in cluster A, b is an object in cluster B, min and max is the minimum and

Table 2 Hierarchical clustering method.

Algorithm: Hierarchical clustering

while (the number of clusters is more than 1 cluster)

1. Find similarity or dissimilarity of a pair of objects in the dataset.
2. Group the pair of objects from step 1 as the same cluster.

end while

3. Determine the position of the hierarchical tree that is suitable for classifying into clusters.

Table 3 Criteria for matching objects.

Criterion	Description
Single	min(dist(*a*,*b*))
Average	$\frac{\sum_{a\in A}\sum_{b\in B} dist(a,b)}{\|A\|\|B\|}$
Complete	max(dist(*a*,*b*))
Ward	$\left(\frac{\|A\|\|B\|}{\|A\|+\|B\|}\right)^{1/2} dist(\frac{\sum a}{\|A\|},\frac{\sum b}{\|B\|})$

maximum function respectively, and dist is the Euclidean distance between (a,b) calculated as:

$$Euclidean\ Distance = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2} \quad (4)$$

where n is the number of object variables and i^{th} is the position of variables.

K-means clustering

The K-means clustering method was first described in 1956. It is an iterative algorithm. This method randomizes a cluster center and chooses an object, which is closest to the cluster center, as a member in the cluster. The algorithm is run until every object in a cluster is not changed to be a member of another cluster. The details of the K-means clustering algorithm are shown in Table 4. where x is an input vector, w is a random weight vector, m is the number of objects in w, n is the number of objects in x, and η is a learning rate.

Kohonen's Self-Organizing Maps (SOM)

SOM or Kohonen's network was introduced in 1982. It is a special class of artificial neural network. SOM is used to find hidden patterns in data, such as in clustering and classification tasks. The algorithm for SOM is shown in Table 5.

K-medoids method

The K-medoids method uses a partitional clustering algorithm, the same as the K-means clustering method.

This method chooses an object from all objects in the dataset as medoids of a cluster (a cluster center), in contrast to the K-means method, which selects a random value as the center of the cluster.

The K-medoids method is more robust to noise and outliers, compared to the K-means method. The steps involved in clustering data using the K-medoids clustering algorithm are shown as Table 6.

Table 4 K-means clustering method.

Algorithm: K-means clustering

1. Decide a value of k.
2. Randomize k positions as cluster centers.

while (an object in some clusters is changed to other clusters)
  for every object
  3. Find the cluster center which gives the minimum distance.
  4. Assign that object in step 3 into that cluster.
  end for
  for every cluster
  5. Adjust a cluster center as the mean of every object in that cluster.
  end for
end while

Table 5 SOM approach.

Algorithm: SOM

1. Randomly initialize all weights $w = [w_1, w_2, w_3,..., w_j,...w_m]$.

while (all weights are changed or training cycles have not passed)

2. Select input vector $x = [x_1, x_2, x_3,..., x_i,...x_n]$.
3. Calculate distance(Euclidean distance) between input vector x_i and all weights w_j in order to find the nearest output node $\text{dist}(w_j,x_i) = \sqrt{\sum_{j=1}^{m} (w_j - x_i)^2}$.
4. Update the winner's weight so that it becomes closer to x_i.
   $w_{j.new} = w_{j.current} + \eta(x_i - w_{j.current})$
5. Adjust η.

end while

Table 6 K-medoids method.

Algorithm: K-medoids method

1. Decide a value of k.
2. Choose k objects randomly from all objects in dataset as medoids.

while (medoids of every cluster change position)

for every object

3. Find medoids which give the minimum distance.
4. Assign that object in step 3 into that cluster.

end for

for every cluster

5. Adjust medoids by setting the object which has the minimum average distance between itself and other objects in the same cluster as new medoids.

end for

end while

K-medoids method integrated with Dynamic Time Warping (DTW)

Nienattrakul and Ratanamahata (2006) integrated DTW, replacing the Euclidean distance with a K-medoids clustering approach in their work.

Dynamic Time Warping distance measure (DTW) is well known in the speech recognition community. It is more a powerful distance measure than the Euclidean distance since it tries to find the minimum distance between two objects.

Suppose that a and b are two objects having n variables as shown in Equations 5 and 6:

$$a = [a_1,a_2,a_3,...,a_i,...,a_n] \quad (5)$$

$$b = [b_1,b_2,b_3,...,b_j,...,b_n] \quad (6)$$

Calculating the DTW distance involves constructing an n × n matrix that keeps the distance in each cell (i,j) calculated from data point a_i and b_j, and the minimum distance in the cell which is adjacent to itself in three directions, $(i\text{-}1,j)$,$(i,j\text{-}1)$,$(i\text{-}1,j\text{-}1)$. Each cell in the matrix can be defined by Equation 7:

$$\gamma(i, j) = dist(i, j) + minb \quad (7)$$

where $minb = \min(\gamma(i\text{-}1,j), \gamma(i,j\text{-}1), \gamma(i\text{-}1,j\text{-}1))$, $\gamma_{(i, j)}$ is distance in cell (i, j), $minb$ is the minimum distance from the adjacent matrix and $dist(i, j)$ is calculated from $(a_i - b_j)^2$.

Then, to find the minimum distance from the shortest path in the matrix $n \times n$, the path must be chosen using Equation 8:

$$DTW(a,b) = \min_{\forall w \in P} \sqrt{\sum_{k=1}^{K} dist_{w_k}} \quad (8)$$

where w is a warping path which has length K,

P is a set of all possible warping paths, and

$dist_{w_k}$ is a distance of a warping path w in position k.

More detail about DTW can be found in Ratanamahatana and Keogh (2004).

When all clustering algorithms had been run, the experimental results were evaluated using RMSSTD (Root Mean Square Standard Deviation), and RS (R-squared) (Halkidi *et al.,* 2002a; Halkidi *et al.,* 2002b).

RMSSTD

The root mean square standard deviation is an evaluation method used to measure the quality of the clustering algorithm (Equation 9). The lower the value of RMSSTD, the better the separation of clusters.

$$RMSSTD = \sqrt{\frac{\sum_{i=1..k}^{} \sum_{j=1..p}^{} \sum_{a=1}^{n_{ij}} (x_a - \bar{x}_{ij})^2}{\sum_{i=1..k}^{} \sum_{j=1..p}^{} (n_{ij} - 1)}} \quad (9)$$

where k is the number of clusters,

p is the number of independent variables in dataset,

$\bar{x}_{ij}$ is the mean of data in variable j and cluster i, and

n_{ij} is the number of data which are in variable p and cluster k.

For RMSSTD values, the average RMSSTD was calculated based on 1,000 iterations of each simulated datasets (Equation 10):

$$Average\ RMSSTD = \frac{Summation\ of\ RMSSTD\ values\ from\ 1,000\ simulated\ datasets}{1,000} \quad (10)$$

RS

The R-squared value is used to determine whether there is a significant difference among objects in different groups and that objects in the same group have high similarity (Equations 11,12 and 13). If RS equals zero, then there is no difference between the groups. On the other hand, if RS equals one, then the partitioning of clusters is optimal.

$$RS = \frac{SS_t - SS_w}{SS_t} \quad (11)$$

$$SS_t = \sum_{j=1}^{p} \sum_{a=1}^{n_j} (x_a - \bar{x}_j)^2 \quad (12)$$

$$SS_w = \sum_{j=1}^{i=1..k} \sum_{a=1}^{n_{ij}} (x_a - \bar{x}_{ij})^2 \quad (13)$$

where SS_t is the summation of the distance squared among all variables,

SS_w is the summation of the distance squared among all data in the same cluster,

k is the number of clusters, p is the number of independent variables in the dataset,

$\bar{x}_j$ is the mean of data in variable j,

$\bar{x}_{ij}$ is the mean of the data in variable j and cluster i and

n_{ij} is the number of data which are in variable p and cluster k.

The average RS was calculated based on 1,000 replications of each simulated dataset (Equation 14):

$$Average\ RS = \frac{Summation\ of\ RS\ values\ from\ 1,000\ simulated\ datasets}{1,000} \quad (14)$$

RESULTS AND DISCUSSION

The objective of the study was to compare the five clustering techniques (Hierarchical, K-means, SOM, K-medoids and K-medoids integrated with DTW) using the criteria of RMSSTD and RS. Both simulated and real datasets were used.

In the first experiment, the simulated datasets were clustered using all approaches to find the average RMSSTD and RS values. This experiment was repeated 1,000 times to provide stable and reliable results and the number of clusters (k) was also varied in order to study any trends.

The results of the first experiment based on the simulated datasets are shown in Tables 7 to 12 and in Figure 2, in which the RMSSTD and RS values are used to evaluate the results. It should be noted that a higher RS value means that the clustering method is better.

In Table 7, showing the average RMSSTD values for 3 variables, the K-means method had the closest RMSSTD results to the SOM method. The RMSSTD values of the K-means and SOM methods were the lowest, even though the number of clusters was varied. This implied that these two methods outperformed the others. The results of this table are plotted in Figure 2(A).

From Table 10, showing the average RS values for 3 variables, using the K-means method had the closest RS results to the SOM method. The RS values of the K-means and SOM clustering approaches were the highest, even though the number of clusters was varied. This suggests that the K-means and SOM methods were the best for 3 variables. The result of this table are plotted in Figure 2(B).

Tables 8 and 11 show the RMSSTD and RS values, respectively, resulting from clustering for the simulated datasets with 5 variables. The results in Table 8 show that the RMSSTD values of the K-means and SOM clustering methods were the lowest compared with the other methods. Table 11 shows the same outcome as Table 10, with the

Table 7 Average RMSSTD value of all approaches tested on simulated data for 3 variables.

Number of clusters	RMSSTD							
	Hier single	Hier average	Hier complete	Hier ward	K-mean	SOM	K-med	K-med (DTW)
2	0.96743	0.85753	0.73992	0.71113	**0.68785**	**0.69337**	0.75515	0.76016
3	0.94344	0.68745	0.62333	0.59387	**0.57429**	**0.57772**	0.64688	0.64698
4	0.92254	0.59418	0.559	0.53166	**0.51893**	**0.52142**	0.58587	0.58451
5	0.90317	0.5477	0.51281	0.48851	**0.4783**	**0.47893**	0.54215	0.53935
6	0.8851	0.5105	0.47657	0.45454	**0.44747**	**0.44815**	0.50579	0.50507
7	0.86862	0.48127	0.44736	0.42715	**0.42258**	**0.4239**	0.47896	0.47837
8	0.85247	0.45425	0.42309	0.40459	**0.40153**	**0.40552**	0.45755	0.45479

Table 8 Average RMSSTD value of all approaches tested on simulated data for 5 variables.

Number of clusters	RMSSTD							
	Hier single	Hier average	Hier complete	Hier ward	K-mean	SOM	K-med	K-med (DTW)
2	0.97191	0.85966	0.75641	0.73013	**0.70714**	**0.71271**	0.77502	0.77214
3	0.94899	0.71353	0.64879	0.62133	**0.60187**	**0.60625**	0.67287	0.6732
4	0.92947	0.63211	0.58744	0.56315	**0.54983**	**0.55201**	0.61245	0.61256
5	0.91187	0.58333	0.54366	0.52031	**0.50957**	**0.51039**	0.5689	0.57041
6	0.89465	0.5461	0.50871	0.48695	**0.47869**	**0.47989**	0.53739	0.5368
7	0.88045	0.51461	0.47996	0.46055	**0.45463**	**0.45693**	0.51186	0.51197
8	0.86544	0.4872	0.45706	0.43922	**0.43526**	**0.4393**	0.4902	0.48968

K-means and SOM methods providing the highest RS values. The values in Table 8 and 11 are plotted in Figures 2(C) and 2(D), respectively.

Tables 9 and 12 show the RMSSTD and RS values, respectively, resulting from clustering for the simulated datasets with 7 variables. Even

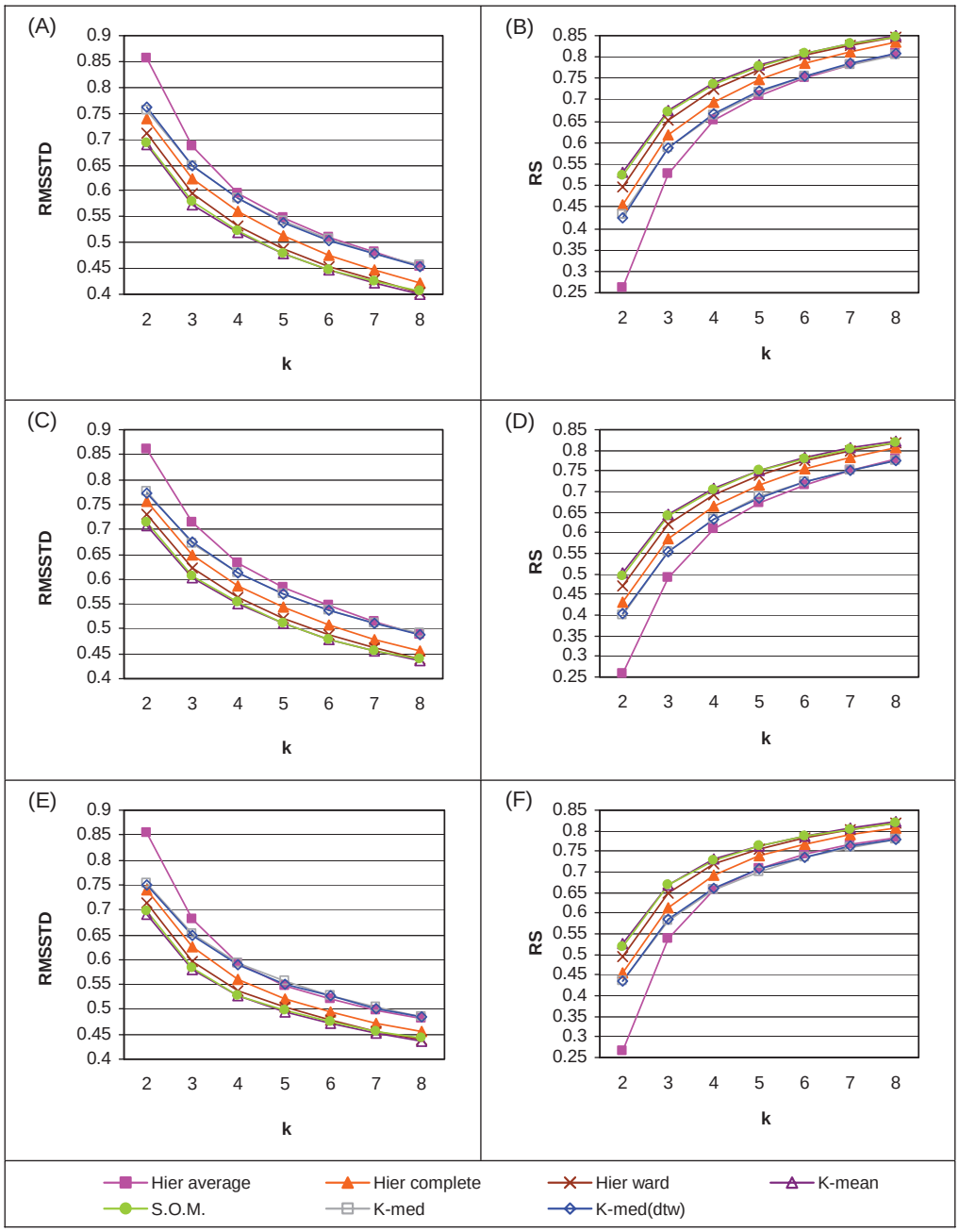

Figure 2 Chart of average RMSSTD and RS values.
(A) average RMSSTD of the 3-variable dataset. (B) average RS of the 3-variable dataset.
(C) average RMSSTD of the 5-variable dataset. (D) average RS of the 5-variable dataset.
(E) average RMSSTD of the 7-variable dataset. (F) average RS of the 7-variable dataset.

though the number of variables was increased to 7, the results in these tables still gave the same results as for 3 and 5 variables, which indicated that K-means and SOM were the best clustering algorithm compared to the others.

The results of the simulated data overall differ from recent work that reported the K-medoids method gave a better clustering result than K-means method (Nienattrakul and Ratanamahata, 2006). This can be explained by the fact that the simulated datasets used in this research did not include noise or outlier data. As a

Table 9 Average RMSSTD value of all approaches tested on simulated data for 7 variables.

Number of clusters	RMSSTD							
	Hier single	Hier average	Hier complete	Hier ward	K-mean	SOM	K-med	K-med (DTW)
2	0.96742	0.85489	0.74039	0.71343	**0.6914**	**0.69635**	0.75288	0.7513
3	0.94349	0.68015	0.62669	0.59672	**0.58014**	**0.58297**	0.65064	0.64722
4	0.92605	0.59267	0.56129	0.53871	**0.52596**	**0.52873**	0.59444	0.59032
5	0.90952	0.5481	0.5211	0.50348	**0.49614**	**0.49807**	0.55666	0.55171
6	0.89173	0.52039	0.49458	0.47789	**0.47307**	**0.47517**	0.52774	0.5265
7	0.87773	0.49788	0.47298	0.45714	**0.45324**	**0.45634**	0.50502	0.50293
8	0.86386	0.48026	0.4545	0.43965	**0.43698**	**0.44195**	0.48648	0.48504

Note: a lower RMSSTD value means that the clustering method is better.

Table 10 Average RS value of all approaches tested on simulated data for 3 variables.

Number of clusters	RS							
	Hier single	Hier average	Hier complete	Hier ward	K-mean	SOM	K-med	K-med (DTW)
2	0.072963	0.26164	0.45495	0.49842	**0.53116**	**0.52353**	0.43083	0.42284
3	0.12687	0.52797	0.61727	0.65374	**0.67627**	**0.67238**	0.58708	0.58679
4	0.17334	0.65413	0.69568	0.7253	**0.73825**	**0.73576**	0.66475	0.66667
5	0.21545	0.70999	0.7467	0.77049	**0.77996**	**0.7794**	0.71625	0.71925
6	0.25399	0.75087	0.78362	0.80338	**0.80939**	**0.80884**	0.75576	0.75644
7	0.28881	0.78104	0.81138	0.8282	**0.83184**	**0.83079**	0.78335	0.78379
8	0.32184	0.80709	0.83312	0.84752	**0.84977**	**0.84678**	0.80447	0.80673

Table 11 Average RS value of all approaches tested on simulated data for 5 variables.

Number of clusters	RS							
	Hier single	Hier average	Hier complete	Hier ward	K-mean	SOM	K-med	K-med (DTW)
2	0.064476	0.25868	0.43064	0.47139	**0.50449**	**0.49659**	0.40112	0.40536
3	0.11672	0.49273	0.58564	0.62098	**0.64443**	**0.63921**	0.55356	0.55343
4	0.16087	0.60832	0.66416	0.69182	**0.70626**	**0.70391**	0.63431	0.6343
5	0.20023	0.6709	0.7154	0.73968	**0.75034**	**0.74951**	0.68794	0.68618
6	0.23788	0.71498	0.75349	0.7744	**0.78197**	**0.78086**	0.72456	0.72511
7	0.26949	0.74978	0.78294	0.80033	**0.80542**	**0.8034**	0.75273	0.75268
8	0.30129	0.77822	0.8053	0.82033	**0.82353**	**0.82022**	0.77574	0.77619

result, K-means was the superior algorithm compared to K-medoids in this case.

In the second experiment, the effectiveness of the clustering methods on a real dataset was measured using an accuracy value calculated from the percentage of matching contents of all clusters, when each resultant cluster was compared to the corresponding original cluster. The real dataset had two clusters, while five clustering techniques were used to classify this dataset into two clusters. The results in Figure 3 indicate that the K-means clustering method and the SOM method had the highest accuracy (83.89%) and the hierachical clustering method with Single criterion has the lowest accuracy (64.43%).

CONCLUSION

In this work, five clustering techniques were compared based on RMSSTD and RS criteria for simulated and real datasets with a multivariate normal distribution. The results showed that the K-means clustering algorithm and Kohonen's Self-Organizing Maps method (SOM) yielded the lowest RMSSTD and highest RS, for the 3, 5 and 7 variables, and for 2, 3, 4, 5, 6, 7 and 8 clusters. In addition, increasing the number of clusters tended to increase the efficiency of the five clustering methods. However, the number of variables did not affect the efficiency of the five clustering methods, which indicated that K-means and SOM were the most suitable algorithms. Moreover, clustering the real dataset, produced the same results with both the study and the simulated data.

Table12 Average RS value of all approaches tested on simulated data for 7 variables.

Number of clusters	RS							
	Hier single	Hier average	Hier complete	Hier ward	K-mean	SOM	K-med	K-med (DTW)
2	0.072984	0.2658	0.45458	0.49518	**0.52627**	**0.51939**	0.43451	0.43713
3	0.12666	0.53918	0.61341	0.65047	**0.66969**	**0.66645**	0.58218	0.58676
4	0.1669	0.65633	0.69336	0.71801	**0.73117**	**0.72832**	0.65523	0.65983
5	0.2042	0.70984	0.73864	0.75621	**0.76322**	**0.7614**	0.70086	0.70636
6	0.24253	0.74151	0.76708	0.78266	**0.7870**	**0.7851**	0.73419	0.73543
7	0.27354	0.76605	0.78925	0.80323	**0.80656**	**0.80389**	0.75932	0.76136
8	0.30342	0.7847	0.8075	0.81995	**0.82213**	**0.81804**	0.77914	0.78037

Note : A higher RS value means that the clustering method is better.

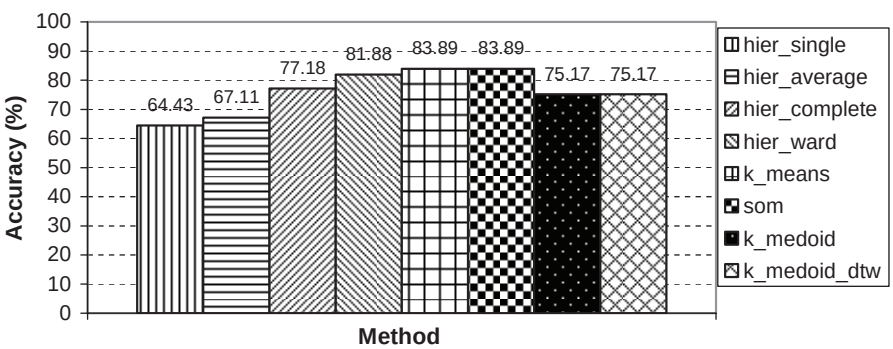

Figure 3 Accuracy of each approach with the real dataset.

LITERATURE CITED

Arms, W. Y. and C. R. Arms. 1978. Cluster analysis used on social science journal citations. **J. Doc.** 34 (1): 1-11.

Boonmung, S., B. Chomtee and K. Kanlayasiri. 2006. Evaluation of artificial neural networks for pineapple grading. **J. Text. Stud.** 37 (5): 568-579.

Clatworthy, J., D. Buick, M. Hankins, J. Weinman and R. Horne. 2005. The use and reporting of cluster analysis in health psychology: A review. **Br. J. Health Psychol.** 10 (3): 329-358.

David, J., Jr. Ketchen and C. L. Shook. 1996. The application of cluster analysis in strategic management research: An analysis and critique. **Strat. Mgmt. J.** 17 (6): 441–458.

Halkidi, M., Y. Batistakis and M. Vazirgiannis. 2002a. Cluster validity methods: Part I. **SIGMOD Rec.** 31(2): 40-45.

Halkidi, M., Y. Batistakis and M. Vazirgiannis. 2002b. Clustering validity checking methods: Part II. **SIGMOD Rec.** 31(3): 19-27.

Hartigan, J. A. and M. A. Wong. 1979. A K-means clustering algorithm. **Applied Stat.** 28 (1): 100–108.

Jardine, N. and R. Sibson. 1968. The construction of hierarchic and non-hierarchic classifications. **Comp. J.** 11 (2): 177-184.

Johnson, S. C. 1967. Hierarchical clustering schemes. ***Psychometrika*** 32 (3): 241-254.

Kaufman, L. and P. J. Rousseeuw. 1990. **Finding Groups in Data: An Introduction to Cluster Analysis.** John Wiley and Sons, Inc., New York. 368p.

Kohonen, T. 1988. **Self-Organization and Associative Memory.** 2nd ed. Springer-Verlag., New York. 312p.

Morrison, D. F. 2002. **Multivariate Statistical Methods.** 4th ed. McGraw- Hill Book Co., New York. 480p.

Niennattrakul, V. and C. A. Ratanamahatana. 2006. **Clustering multimedia data using time series**. **ICHIT.** 1: 372-379.

Ratanamahatana, C. A. and E. Keogh. 2004. **Making Time-series classification more accurate using learned constraints**. **SDM.** 4: 11-22.

Sheng, W. and X. Liu. 2006. A genetic k-medoids clustering algorithm. **J. Heuristics** 12 (6): 447-466.