Pertanika J. Sci. & Technol. 25 (1): 77 - 90 (2017)

PERTANIKA JOURNALS

SCIENCE & TECHNOLOGY

Journal homepage: http://www.pertanika.upm.edu.my/

Modified Kohonen Network Algorithm for Selection of the Initial Centres of Gustafson-Kessel Algorithm in Credit Scoring

Sameer, F.[1, 2]* and Abu Bakar, M. R.[1]

[1]*Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia*

[2]*Faculty of Science, Universiti of Baghdad, 98798 Baghdad, Iraq*

ABSTRACT

Credit risk assessment has become an important topic in financial risk administration. Fuzzy clustering analysis has been applied in credit scoring. Gustafson-Kessel (GK) algorithm has been utilised to cluster creditworthy customers as against non-creditworthy ones. A good clustering analysis implemented by good Initial Centres of clusters should be selected. To overcome this problem of Gustafson-Kessel (GK) algorithm, we proposed a modified version of Kohonen Network (KN) algorithm to select the initial centres. Utilising similar degree between points to get similarity density, and then by means of maximum density points selecting; the modified Kohonen Network method generate clustering initial centres to get more reasonable clustering results. The comparative was conducted using three credit scoring datasets: Australian, German and Taiwan. Internal and external indexes of validity clustering are computed and the proposed method was found to have the best performance in these three data sets.

Keywords: Credit Scoring, Decision-making, Clustering Techniques, Fuzzy Clustering Algorithms, Gustafson-Kessel Algorithm, Kohonen Network

INTRODUCTION

Banks' databases contain information about their customers and the financial history of their payments. The databases can be utilised to assess the credit risk by investigating whether it can be a good basis on which to predict borrowers' ability to repay their loans on time.

The credit-scoring technique is commonly used to evaluate the creditworthiness of credit clients. The credit risk evaluation system plays an important role in decision making to enable faster decisions for credit, lessen the possible risk and reduce the cost

Article history:
Received: 30 December 2015
Accepted: 04 July 2016

E-mail addresses:
fadfhaa55@gmail.com (Sameer, F.),
drrizam@gmail.com (Abu Bakar, M. R.)
*Corresponding Author

ISSN: 0128-7680 © 2017 Universiti Putra Malaysia Press.

of credit analysis. The 2008 financial crisis revealed the importance of credit risk evaluation decisions, not only for financial institutions and banks, but also for both the global and local economy (Wu, 2008).

The credit models built with a credit risk evaluation technique ought to fulfil two essential criteria: precision, which means that they are capable of predicting the behaviour of customers, and transparency, which means that the model is able to describe the input-output relationship in an understandable way. Credit scoring classifies credit applicants as ‘bad’ or ‘good’ customers by considering features like age, monthly income, and marital status (Yang, 2007). Statistical methods have been used most oftenly for assessing credit risk for customers. Logistic regression and linear discriminate analysis are most commonly used (Thomas, 2000). Accuracy of credit scoring for several neural networks was investigated (West, 2000). Results were benchmarked against traditional statistical methods like linear discriminant analysis, logistic regression, k-nearest neighbour, and decision trees. Clustering techniques provide distinct new options for measurable routines for building credit scoring models. Recently, more pragmatic approaches have been adopted, and several classification techniques have appeared to perform well for credit scoring. Since the introduction of Fuzzy logic by Zadeh in 1965, it has successfully been implemented in many fields like credit scoring. Clustering is a technique that is used in data analysis. This method is used to find groups in a data set such that there are the most similarities in each group and the most dissimilarity between different groups. Gustafson–Kessel (GK) algorithm is one of fuzzy clustering techniques used in credit scoring. Gustafson–Kessel (GK) algorithm also has its drawbacks in relation to choosing initial centres (Gustafson & Kessel, 1979). In order to overcome the sensitivity of the initial point choice and increase the accuracy of credit decisions. Another method was used to obtain better initial centres.

In this paper, we modified the kohonen algorithm for selecting the centres of clusters of Gustafson-Kessel algorithm. The paper is organised as follows: In section 2, the Clustering analysis techniques and Self-Organising Map are described. Section 3 describes the modified kohonen algorithm, while Sections 4 provides a brief overview of the measures of cluster validity and Section 5 presents the experimental analysis and results. Discussion and conclusion are given in Sections 6 and 7, respectively.

MATERIAL AND METHODS

Clustering Analysis Techniques

Clustering is used to assign a set of objects to groups (called clusters) and the objects in the same cluster are more similar than to the objects in other clusters. Based on the similarities between the objects, cluster analysis is the classification and the organisation of the objects into groups (Gan, Ma, & Wu, 2007). Clustering partition methods can be fuzzy or hard. Hard clustering partition methods are based on the classical set theory, which requires an object to belong to only one cluster. Fuzzy clustering partition methods allow objects to belong to many clusters simultaneously, with different degrees of membership (Touzi, 2010).

The Gustafson – Kessel Algorithm

The GK algorithm is a powerful fuzzy clustering technique, with a large number of applications, such as image processing and classification systems. The importance of this algorithm lies in its ability to estimate the cluster covariance matrix to adapt the distance metric to the shape of the cluster (Gustafson & Kessel, 1979).

The GK algorithm needs a set of N samples in the n dimensional space and the number of clusters as the input parameters. A fuzzy partition of the data set X can be represented by an $(N * k)$ matrix $\boldsymbol{U} = [u_{ij}]$, where u_{ij} denotes the degree of membership, with which the i^{th} object belongs to the j^{th} cluster, where $(1 \leq i \leq N)$ and $(1 \leq j \leq k)$. $\boldsymbol{U}$ is the fuzzy partition matrix, and it must satisfy the following constraints:

- $0 \leq u_{ij} \leq 1 \quad \text{for } i \in \{1,\ldots,N\},\ j \in \{1,\ldots,k\}.$

$0 < \sum_{i=1}^{N} u_{ij} < N \quad \text{for } j \in \{1,\ldots,k\}$ (1)

- $\sum_{j=1}^{k} u_{ij} = 1 \quad \text{for } i \in \{1,2,\ldots,N\}.$

The final constraint expresses that the sum of the memberships of an object over the all set of clusters must be equal to 1. The number of clusters is at least two. The objective function of the *GK* algorithm is defined as follows:

$$J_m = \sum_{i=1}^{N}\sum_{j=1}^{k} {u_{ij}}^m d_{ij}^2 \tag{2}$$

where

$$d_{ij}^2 = (x_{i-}c_j)\ A_j\ (x_i - c_j)^T \tag{3}$$

$C=(c_1, c_2, \ldots.., c_k)$ represents the cluster centre in R^n, m represents the fuzziness exponential, where $1 \leq m < \infty$, $d_{ij} = d(c_j, x_i)$, is the distance between the centre c_j and the data point x_i, and u_{ij} is the degree of membership of point x_i in the j^{th} cluster. A_j is the Mahalanobis distance matrix for the j^{th} cluster; if A_j is the identity matrix, then the square Euclidean distance measure is obtained. A_j is defined as follows:

$$A_j = V_j[det(F_j)]^{\frac{1}{n}} F_j^{-1} \tag{4}$$

Here n is the number of attributes or features, V_j is the volume of the j^{th} cluster and F_j is the cluster covariance matrix from the following formula:

$$F_j = \frac{\sum_{j=1}^{k} (u_{ij})^m (x_i - c_j)(x_i - c_j)^T}{\sum_{j=1}^{k} (u_{ij})^m} \tag{5}$$

In the GK algorithm, the cluster shape changes depending on the data, and can be in an ellipsoidal form or a hyper ellipsoidal form. For this reason, the GK algorithm employs the covariance matrix.

The steps of the GK algorithm are as follows:

1. Given a dataset $X=\{x_1, x_2, \ldots., x_N\}$.
2. Select the k which represent the number of clusters, ($2\leq k\leq N$), and select the termination condition $\epsilon > 0$.
3. Choose the initial centre c_j from the dataset.
4. Compute the cluster covariance matrix using the formula in equation (5).
5. Compute the Mahalanobis distance using equation (3).
6. Compute the partition matrix (u_{ij}), as follows:

$$u_{ij} = \frac{1}{\sum_{r=1}^{k}(\frac{d_{ij}}{drj})^{\frac{2}{m-1}}} \tag{6}$$

where, i=1, 2, …., N, and j=1, 2, …., k

7. Update the C-means matrix (c_j), as follows:

$$c_j = \frac{\sum_{i=1}^{N}\left(u_{ij}\right)^{m} x_i}{\sum_{i=1}^{N}\left(u_{ij}\right)^{m}} \quad for\ 1 \leq j \leq k \tag{7}$$

8. Repeat the above steps until the centre matrix for two sequential iterations is stable, in the following sense:

$$\|C^{I+1} - C^{I}\| < \epsilon \tag{8}$$

where (I) represents the number of iterations.

Self-Organising Map

Networks with supervised training techniques are networks with a target output for every input pattern. The networks learn how to produce the outputs that are required; in unsupervised training, however, the networks learn to form classifications of the training data without external supervision (Yin, 2008). When input patterns and features are shared, the network is able to identify those features across the input patterns. An unsupervised system is based on competitive learning, in which the neurons that are the output compete among themselves to be activated. Only one neuron is activated at any one time, and this activated neuron is called the winning neuron (Kohonen et al., 2009). This competition can be implemented if the neurons have parallel inhibition connections (negative criticism ways) among them; therefore, the neurons learn to organize themselves. This neural network is called a self-organising map (SOM) and projects high-dimensional data onto a low-dimensional grid (Kohonen, 1982).

Kohonen Network

A Kohonen network is a classification method that forms the basis of self-organising maps (SOM) (Kohonen, 1982). The Kohonen network method, which was proposed by Kohonen, has a single computational layer arranged in rows and columns, and is a feed-forward structure. Every neuron is connected to all the nodes in the input layer or source (Kohonen, 1998). The Kohonen algorithm is an unsupervised classified network and it deals with inputs that are unable to be led without overlapping in the classes. It is robust (that is, it is able to resist noise); therefore, the Kohonen algorithm has interesting properties.

MODIFIED KOHONEN ALGORITHM

The GK algorithm chooses points as initial clustering centres randomly, and different points may lead to different solutions. In order to overcome the sensitivity of the initial point choice, we modified the Kohonen Network method to obtain better initial centres. The steps of the algorithm are as follows:

1. Compute the similar neighbourhood of each point *x* of data set *X*; it is denoted by *simneighbor (x, r)*, take *x* as the centre and *r* is the threshold value of degree similarity. The objects with large degree of *(r)* are in similar neighbourhood of point *x*.

$$Simneighbor(x,r)=\left\{\begin{matrix} x_i | r \leq sim_{x_{i\in}X}(x_{i,}x) \leq 1{,}0 \leq r \leq 1, \\ X = \{x_1, \dots, x_n\} \end{matrix}\right\} \quad (9)$$

where $sim_{x_{i\in}X}(x_{i,}x)$ is the similarity degree between x_i and x can be denoted by the formula

$$sim_{x_{i\in}X}(x_{i,}x) = \lambda d_m(x_{i,}x) + (1 - \lambda)cos(x_{i,}x) \quad (10)$$

and $d_{m_{x_i,x\epsilon X}}(x_{i,}x) = \frac{d_{x_i,x\epsilon X}(x_{i,}x) - min_{x_i,x\epsilon X}\{d(x_{i,}x)\}}{max_{x_i,x\epsilon X}\{d(x_{i,}x)\} - min_{x_i,x\epsilon X}\{d(x_{i,}x)\}}$ is normalised Euclidean distance (Zhang, 2013). The symbol $cos\ (x_{i,}x)$ is the cosine of the intersection angle of two points and can be computed by the formula:

$\frac{\sum_{i=1}^{p}(x_i x_{j)}}{\sqrt{\sum_{i=1}^{p} x_i^2}\sqrt{\sum_{i=1}^{p} x_j^2}}$ and its value ranging from -1 to 1.

2. Compute the similarity density of each point and it is denoted by Density (x_i) where x_i belong to X. The formula of density is as

$$Density_{x_i \in X}(x_i) = \frac{\sum_{i=1}^{|p_{neighbor(x_i)}|} sim(x_i, p_{neighbor(x_i)})}{|p_{neighbor(x_i)}|} \quad ,$$

$$X = \{x_i, \ldots, x_n\} \tag{11}$$

The symbol $p_{neighbor(x_i)}$ denoted the points that satisfy the threshold r in similar neighbourhood of x_i.
The $|p_{neighbor(x_i)}|$ is the number of points in the neighbourhood of x_i.

3. Choose the points that have high means of max density points as the weights of kohonen algorithm. And input the number of nodes (k, which equals the number of weights) and let I=1 represent the time or number of iterations.

4. Compute the distance to nodes by determining the Euclidean distance d_j between the input data point and each weight:

$$d_j = \sum_{i=1}^{N-1}(x_i - x_j), \text{ for } j = 1,.., K \,;\, i = 1,\ldots,N \tag{12}$$

where x_j points that have high density in dataset X.

5. Select the winning node j^* that produces the minimum d_j and update the weights at iteration I for node j^* and its neighbours:

$$x_j(\mathrm{I}+1) = x_j(\mathrm{I}) + \eta(\mathrm{I})(x_i - x_j) \tag{13}$$

where $\eta(\mathrm{I})$ is the learning rate parameter, with the initial learning rate parameter being set (usually to a figure between 0.2 and 0.5). The learning rate is initialised at 0.5, and will decrease at each iteration by the following expression:

$$\eta(\mathrm{I}+1) = 0.5\eta(I) \tag{14}$$

The nodes in the neighbourhood of j^* become more similar to the input vector x_i, after these updates.
After optimising the modified Kohonen Network method of selection the initial cluster centres, the Gustafson–Kessel Algorithm begins with these centres clustering analysis, as follows:

6. Input the centres retrieved from the modified Kohonen Network method as the initial centres of the clusters.

7. Compute the cluster covariance matrices using equation (5).

8. Compute the distances using equation (3).

9. Compute the partition matrix using equation (6).

10. Update the C-means matrix (c_j) from the following expression:

$$c_j = \frac{\sum_{i=1}^{N} (u_{ij})^m x_i}{\sum_{i=1}^{N} (u_{ij})^m}$$

11. Repeat the above steps (6 to 10) until the centres matrix for two sequential iterations (*I,I*+1) are stable in the following sense:

$$\|C^{I+1} - C^I\| < \epsilon$$

At this step, the partition matrix gated by fuzzy clustering is used to classify the creditworthiness of credit clients.

MEASURES OF CLUSTER VALIDITY

The clustering algorithm always seeks to find the best fit for a fixed number of clusters and the shapes of the parameterised cluster. Cluster validity refers to whether a given fuzzy partition fits the data at all. The number of clusters is application-specific and is usually identified by a user. Cluster validity criteria are applied to determine the optimal number of clusters and a good clustering algorithm (Wang & Zhang, 2007). Although there are many cluster validity measures that can be used for this purpose, none is perfect. There are mainly two types of validity measures:

- External measures: using the class label for cluster analysis.
- Internal measures: use the vectors for analysis.

Internal Measures

Several internal indices are used simultaneously and the most important ones are described below:

Partition Coefficient (PC). This measures the amount of "overlapping" between clusters. Bezdek (Bezdek, 1981; Pal & Bezdek, 1995) defines it as follows:

$$PC(k) = \frac{1}{N} \sum_{j=1}^{k} \sum_{i=1}^{N} \tag{12}$$

Here, u_{ij} is the membership degree of the i^{th} data point to the j^{th} cluster. The best algorithm for partitioning the data is the one that produces the highest value of PC.

Classifications Entropy (CE) (Pal & Bezdek, 1995). This measures only the fuzziness of the clusters partition, so it has similarity to the Partitions Coefficient.

$$CE(k) = -\frac{1}{N} \sum_{j=1}^{k} \sum_{i=1}^{N} u_{ij} \log(u_{ij}) \tag{13}$$

The best clustering algorithm is the one with the lowest value for this index.

Partitions Index (SC) (Bensaid, 1996). This is the ratio between the sum of the separation and the compactness of the clusters. It is the sum of the cluster validity measures for each individual divided by the fuzzy cardinality for each cluster.

$$SC(k) = \sum_{j=1}^{k} \frac{\sum_{i=1}^{N} (u_{ij})^m ||x_i - c_j||^2}{\sum_{i=1}^{N} u_{ij} \sum_{d=1}^{k} ||c_d - c_j||^2} \tag{14}$$

When comparing different partitions with an equal number of clusters, SC is useful. A better partition can be obtained by a lower value of SC.

Separation Index (S) (Bensaid, 1996). In contrast to the partition index (SC), the separation index uses a minimum-distance separation for partition validity, and a lower value of S indicates a better partition.

$$S(k) = \sum_{j=1}^{k} \frac{\sum_{h=1}^{N} (u_{ij})^2 ||x_h - c_j||^2}{N min_{i \neq j, i=1,..,k} ||c_i - c_j||^2} \tag{15}$$

Xie and Beni's Index (XB) (Xie & Beni, 1991). This aims to measure the proportion between the total variation within clusters and the separation of clusters, and it is defined as follows:

$$XB(k) = \sum_{j=1}^{k} \frac{\sum_{i=1}^{N} (u_{ij})^m ||x_i - c_j||^2}{N min_{ij} ||x_i - c_j||^2} \tag{16}$$

This index focuses on separation and compactness properties. The clusters are more separated if Index (*XB*) has a smaller value.

Dunn's Index (DI) (Xie & Beni, 1991). This index aims to recognise dense and well-separated clusters. It is defined as the proportion between the minimal intra-cluster distance and the maximal inter-cluster distance. For each cluster partition, this index can be identified as follows:

$$DI(k) = min_{j \in k} \left\{ min_{j \in k i \neq j} \left\{ \frac{min_{x \in k_i, y \in k_j} d(x,y)}{max_{x,y \in k} d(x,y)} \right\} \right\} \tag{17}$$

A high Dunn's index indictes a desirable algorithm for producing clusters.

Davies-Bouldin index (DB) (Davies & Bouldin, 1979). This index can be identified as follows:

$$DB = \frac{1}{n} \sum_{i=1}^{n} max_{i \neq j} \left(\frac{d_i + d_j}{d(c_i, c_j)} \right) \tag{18}$$

Here n is the number of clusters, c_j and c_i are the centres of cluster, while d_j *and* d_i are the average distances of all elements in clusters j and i respectively, and $d(c_i,c_j)$ is the distance between the centres c_i and c_j. The best algorithm is the clustering algorithm that produces a collection of clusters with the smallest DB index.

External Measures

These methods give an indication of the quality of the resulting partitioning and thus they can only be considered as a tool at the disposal of the experts in order to evaluate the clustering results. Fuzzy Rand Index is a well-known measure of similarity between two partitions of a data set (Hullermeier, 2012). Given a fuzzy partition P = {P1, P2 ,…, P_k} of X, each element $x \in X$ can be characterised by its membership vector.

$$P(x) = (P_1(x), P_2(x), \ldots, P_k(x)) \in [0; 1]^k \tag{19}$$

where $P_i(x)$ is the degree of membership of x in the *i-th* cluster P_i. A similarity measure on the associated membership vectors (19) can be formed as:

$$E_P(x, x') = 1 - ||P(x) - P(x')|| \tag{20}$$

Where, ||.|| is a proper metric on $[0; 1]^k$ if two fuzzy partitions P and Q are given. To generalise the concept of concordance, a pair (x; x') is defined and the degree of concordance is:

$$conc(x, x') = 1 - ||E_P(x, x') - E_Q(x, x')|| \in [0\ 1] \tag{21}$$

the degree of discordance is:

$$disc(x, x') = ||E_P(x, x') - E_Q(x, x')|| \tag{22}$$

the distance measure on fuzzy partitions is then defined by the normalised sum of degrees of discordance:

$$d(P, Q) = \frac{\sum_{(x,x')\in X} ||EP(x,x') - EQ(x,x')||}{N(N-1)/2} \tag{23}$$

Likewise,

$$RE(P,Q) = 1 - d(P,Q) \tag{24}$$

Corresponding to the normalised degree of concordance, and therefore, it is a direct generalisation of the original Rand index. Rand index is a similarity measure which assumes values between 0 and 1. If near 1 means that the *i-th* cluster in P and the *i-th* cluster in Q are identical, thus P=Q.

EXPERIMENTAL ANALYSIS AND RESULTS

Data Sample

We run experiments on three real-life data sets: Australian credit data, German credit data and Taiwan credit data (UCI machine learning databases). The Australian credit data are composed of 690 entries, of which 307 match creditworthy clients and 383 match bad clients. Each entry is described by 14 features or attributes, 6 of which are continuous, while the remainders are categorical. The German credit data consist of 1000 entries, 70% of which correspond to clients who are a good credit risk and 30% of which relate to clients to whom credit should be refused or who are bad credit risk. Each client is described by 20 features, including personal information such as age, sex and marital status, existing accounts, credit history records, loan amount and purpose and employment status. Seven features are continuous, and the rest are categorical. Taiwan data are composed of 30000 entries, with 23 features and of which 23364 match creditworthy clients and 6636 match bad clients. Table 1 shows the characteristics of the datasets.

Table 1
Characteristics of the data sets used in the experiment

Dataset	Number of features	Number of data	Number of groups
Australian	14	690	2
Germany	20	1000	2
Taiwan	23	30000	2

Implementation

The work was implemented using MATLAB version R2010a by creating a programme to perform the GK algorithm, GK+MK) algorithm and modified kohonen algorithm. Tables 2, 3 and 4 show the results for the objective function, number of iterations, as well as internal indexes and external indexes.

Table 2
The validity measures of Australian credit data

Algorithm validity measures	*Gustafson-Kessel(GK)*	Gustafson-Kessel with Kohonen Network method(GK+K)	Gustafson-Kessel with modified Kohonen Network method (GK+MK)
PC	0.8277	0.8487	0.8959
CE	0.3897	0.3269	0.3247
SC	2.9343	1.8472	1.7238
S	1.7605	1.5121	1.2952
XB	1.5017	0.9084	0.3133
DI	1.4938e-005	1.4938e-005	1.2282
DB	1.4972	1.4972	0.8140
***J* (objective Function)**	94.2459	90.3452	80.3230
No. iteration	31	25	20

Table 3
The validity measures of German credit data

Algorithm	Gustafson-Kessel (GK)	Gustafson-Kessel with Kohonen Network method (GK+K)	Gustafson-Kessel with modified Kohonen method (GK+MK)
validity measures			
PC	0.5698	0.8007	0.8854
CE	0.6084	0.6374	0.5394
SC	0.5694	0.4421	0.3869
S	0.2952	0.2480	0.2039
XB	0.1721	0.1678	0.1216
DI	0.2741	0.2489	0.1234
DB	0.7487	0.7387	0.6387
***J*(objective Function)**	417.3321	316.3481	250.3240
No. iteration	19	15	12

Table 4
The validity measures of Taiwan credit data

Algorithm	Gustafson-Kessel (GK)	Gustafson-Kessel with Kohonen Network method (GK+K)	Gustafson-Kessel modified with Kohonen Network method (GK+MK)
validity measures			
PC	0.5000	0.5035	0.7432
CE	0.6931	0.6894	0.5437
SC	6.1690	5.8680	1.8760
S	8.1324	1.1483	0.5630
XB	1.5423	1.5212	0.5481
DI	0.0043	0.0045	0.0033
DB	0.8709	0.8709	0.7540
***J*(objective Function)**	155	153	140
No. iteration	10	6	5

Table 5
The fuzzy rand validity measure of three credit data

Algorithm	Gustafson-Kessel (GK)	Gustafson-Kessel with Kohonen Network method (GK+K)	Gustafson-Kessel modified with Kohonen Network method (GK+MK)
data			
Australian	0.8600	0.8803	0.9410
Germany	0.7888	0.8406	0.9032
Taiwan	0.8476	0.8530	0.8942

DISCUSSION

As shown in the summarised results, the modified Kohonen method with Gustafson-Kessel algorithm (GK+MK) has the best performance for the three data sets, since it has a smaller distance function (objective function) and a lower number of iterations for the three data sets.

Table (2) shows the internal index values for the Australian data for GK, GK+K and GK+MK. The value of the first index (the partitions index, PC) for the GK+MK algorithm is near to 1, which is greater than its value for the GK and GKK. The value of the second index (the Classification Entropy, CE) for the GK+MK algorithm is less than the value for the GK and GKK. It can be seen that the values of the other indexes (SC, S, XB) for the GK+MK algorithm are lower than the values for the GK algorithm, but DI and DB for two algorithms are equal, which mean the algorithms well-separated clusters. The objective function GK+MK algorithm is less than the value for the GK and GK+K Algorithms. The number of iterations is 31 for the GK Algorithm, which is greater than the iterations of *GKK* algorithm 25.

In Table 3, the index values of the German data are showed. The partitions index value (PC) for the new GK+MK algorithm is greater than the value of GK and GKK algorithm, which means the GK+MK algorithm is the best. The Classification Entropy, CE in GK+MK algorithm is less than the value in GK and GKK algorithm, which means the fuzziness of the clusters partition for GK+MK algorithm is less than its coordinate for the GK and GKK algorithm. The other indexes (SC, S, and XB) for the GK+MK algorithm are lower than the values for the GK and GKK algorithm.

DI and DB for the two algorithms are unequal, which means the GK+MK algorithms are well-separated clusters than the GK and GKK algorithm. In addition, the objective function and number of iterations for GK+MK algorithm are less than the value of the two algorithms.

Table 4 shows the internal index values for the Taiwan data for the algorithms. The values of the two indexes (the partitions index, PC and Dunn's index DI) for the new GK+MK algorithm are high, while the values of the other indexes (SC, S*,* XB , DB) are low. The results indicate that the new GK+MK algorithm is the best.

Table 5 shows the fuzzy rand validity measure of three credit data. The results show that the values of GK+MK are greater than values of the two other method fuzzy partitions are robust.

CONCLUSION

In this paper, a new modified Kohonen method to centres selection of fuzzy clustering was proposed. Developing and improving the GK algorithm to identify the centres of clusters, the three algorithms were applied to three datasets. A comparative study among the algorithms was carried out.

The cluster internal validity indexes confirmed that the performance of the proposed algorithm (GK+MK) is better than that of the GK and GKK algorithms. A fuzzy validity index is applied in this paper for evaluating the fitness of clustering to data sets.

ACKNOWLEDGEMENT

We are thankful to God who has given us the power, good sense and confidence to complete this research analysis successfully, and I am grateful to my research supervisor and to the Department of Mathematics for support, valuable suggestions and guidance. I also would like to express gratitude to the editor and to others who have made suggestions and given help with comments to improve the paper.

REFERENCES

Bensaid, A. (1996). Validity–guided (Re) Clustering with Application to Image Segmentation. *IEEE Transactions on fuzzy Systems, 4*(2)*,* 112-123.

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

Davies, D. L., & Bouldin, D. W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1*(2), 224-227.

Gan, G., Ma, C., & Wu, J. (2007). *Data Clustering: Theory, Algorithm and Application* (Vol. 20). Society for Industrial and Applied Mathematics (SIAM).

Gustafson, D. E., & Kessel, W. (1979). Fuzzy clustering with a Fuzzy Covariance Matrix. In *Proceeding of 55th IEEE Conference on Decision and Control* (pp. 761–766). Las Vegas, USA.

Hullermeier, E., Rifqi, M., Henzgen, S., & Senge. (2012). Comparing Fuzzy Partitions: A Generalization of the Rand Index and Related Measures. *IEEE Transactions on Fuzzy Systems, 20*(3), 546-555.

Kohonen, T. (1982). Self-Organized Formation of Topologically Correct Feature Maps. *Biological Cybernetics, 43*(1), 59–69.

Kohonen, T. (1998). *Self Organization and associative memory* (2nd Ed.). Springer Series in Information Sciences. Berlin: Springer.

Kohonen, T., Nieminen, I., & Honkela, T. (2009). On the quantization error in SOM vs. VQ: a critical and systematic study. In *International Workshop on Self-Organizing Maps* (pp. 133-144). Springer Berlin Heidelberg.

Lahsasna, A., Ainon, R. N., & Wah, T. Y. (2010). Credit scoring models using soft computing methods: A survey. *The International Arab Journal of Information Technology, 7*(2), 115-123.

Pal, N. R. & Bezdek, J. C. (1995). On cluster validity for the fuzzy c-means model, IEEE Trans. *Fuzzy System, 3*(3), 370–379.

Thomas, L. C. (2000). A survey of credit and behavioral scoring: Forecasting financial risks of lending to customers. *International Journal of Forecasting, 16*(2), 149-172.

Touzi, A. G. (2010). An Alternative Extension of the FCM Algorithm for Clustering Fuzzy Databases. In *Advances in Databases Knowledge and Data Applications (DBKDA), 2010 Second International Conference on* (pp. 135-142). IEEE.

Wang, W. N., & Zhang, Y. J. (2007). On fuzzy cluster validity indices. *Fuzzy sets and systems, 158*(19), 2095–2117.

West, D. (2000). Neural network credit scoring models. *Computers and Operations Research, 27*(11–12), 1131–1152.

Wu, X. (2008, June 29). *Credit Scoring Model Validation*. (Master Disertation). Faculty of Science, Korteweg-de Vries Institute for Mathematics, Universiti Van Amsterdam.

Xie, X. L. & Beni, G. (1991). A validity measure for fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 13*(4), 841-846**.**

Yang, Y. (2007). Adaptive credit scoring with kernel learning methods. *European Journal of Operational Research, 183*(3), 1521–1536.

Yin, H. (2008). The self-organizing maps: background, theories, extensions and applications. In *Computational intelligence: A compendium* (pp. 715-762). Springer Berlin Heidelberg.

Zhang, Y., & Cheng, E. (2013). *An Optimized Method for Selection of the Initial Centres of K-Means Clustering* (pp. 149-156). Springer-Verlag Berlin Heidelberg.