A Proposed Clustering Method for Customer Segmentation in CRM Practices

Russell K.H. Ching[1), Ja-Shen Chen[2), Yi-Shen Lin[3)

[1) California State University, Sacramento Management Information Science Department (chingr@csus.edu)
[2) Yuan-Ze University, Department of Business Administration (jchen@saturn.yzu.edu.tw)
[3) Yuan-Ze University, Graduate Institute of Management (s897683@mail.yzu.edu.tw)

Abstract

Customer Relationship Management (CRM) has been a widely discussed issue in recent year. One of the important tasks for CRM practices is to classify customers into different clusters such that different marketing campaigns and/or promotion tactics can be applied for customers in different clusters. In this paper, a modified K-means algorithm was proposed to improve clustering performance and robustness comparing to original K-means algorithm. The idea as well as the procedure of the proposed algorithm was addressed. In addition, a number of test datasets were used to verify performance according to their computation results.

1. Introduction

Due to increasing competition and the trend toward e-business and global markets, many companies are beginning to invest heavily into information technology (IT) as a means of building and maintaining stronger relationships with their customers through real time customized services. Customer relationship management (CRM) is defined as an overall process that relies upon technology (Crosby and Johnson 2000) to manage the interactions between a business and its customers (Storey 2001b), and gain greater insights into its individual customer's needs (Ryals and Knox 2001). It helps businesses establish learning relationships with their customers (Peppers et al. 1999) and other external entities, and develop customer loyalty (Kohli et al. 2001; Sharp and Sharp 1997). By understanding the individual needs of its customers, businesses will be able to offer customized products and services that are congruent to their particular needs (Peppard 2000; Peppers et al. 1999; Ryals and Knox 2001; Yim and Kannan 1999). However, success lies in how well the business can categorize its customers into groups or segments with the same needs, and build its relationships based upon these segments. Many of today's business leaders recognize that customer relationships will be the future's major source of a sustainable competitive advantage. Thus, CRM will be a critical issue in the near future (Battista and Verhun 2000).

One of the most critical functions of CRM lies in its ability to distinguish customer segments (Storey 2001a). Treating customers to their preferred levels of service forms the basis for segmentation (Cholewka 2001). Often, a predefined similarity measure is used to divide customers into homogeneous groups or clusters. By identifying such clusters, a business can better serve their needs and in turn, gain their loyalty. A technique frequently used is clustering analysis. Although clustering techniques provide an adequate means for identifying and separating customer segments, most to date have not yet been perfected. Chen and Sakaguchi (2000) point out that most customer segmentation techniques frequently perform below expectations, and leave many opportunities for improving their accuracy. Thus, CRM performance lags in this area.

As a means to improve the segmentation performance of CRM and other related marketing practices, this paper presents the most popular segmentation technique, K-Means (MacQueen 1967), and develops a modified version of K-Means algorithm. The paper is organized as following: Section 2 provides a brief review of clustering methods. The details of the proposed algorithm are presented in section 3. Section 4 covers the experimental results comparing performance of the proposed method with K-Means and the conclusions are presented in section 5.

2. Literature Review

Clustering techniques generally can be categorized into hierarchical, density-based, grid-based, model-based methods, and partitioning approaches (Han and Kamber 2001). Further, a number of clustering algorithms have been developed recently. For example, Fred (2001) proposed a majority voting method to find consistent clusters. Dash et al. (2001) developed a clustering method called BRIDGE to integrate K-Means and DBSCAN for dealing with clusters of arbitrary shape. Even though many clustering methods have been brought up, K-Means is still one of the most widely used algorithms (Murthy and Chowdhury 1996; Al-Daoud and Roberts 1996), especially in marketing research area (Green and Krieger 1995; Hruschka and Natter 1999). Since the objective of clustering is to group data with maximal intra-class similarity and minimal interclass similarity (Al-Daoud and Roberts 1996; Murthy and Chowdhury 1996; Han and Kamber 2001), K-Means can be applied to market segmentation so as to identify and group customers with similar behavior patterns.

Despite of the popular use of K-Means algorithm, many scholars (ex, Selim and Ismail 1984, Bandyopadhyay et al. 2001) has found that K-Means algorithm might get stuck into sub-optimal values. Huang (1998) suggested the reason why K-Means algorithm produces locally optimal solutions is related to the selection of initial points. Similarly, Milligan (1980) and Huth (1996) also expressed their concerns on initialization problem. The study of initialization problem to improve K-Means performance is a quite popular issue. For example, Linde et al. (1980) developed a binary splitting technique to generate initial condition by splitting centroids; Tou and Gonzales (1974) developed Simple Cluster-Seeking (SCS) method to make sure that all initial points is broadly located. Katsavounidies et al. (1994) also proposed a KKZ method which claimed to produce better results than the Binary Splitting technique. Al-Daoud and Roberts (1996) proposed two initialization methods, which split space to generate initial conditions. Furthermore, Pena et al. (1999) also took Square Error (SE) as the measurement of similarity to compare performance of some initialization methods.

In addition to studies on initialization problem, some other clustering algorithms have also been developed and explored in the past decade. For example, Krishna and Murty (1999) developed a novel hybrid genetic algorithm for a better clustering result. Balakrishnan et al (1996) suggested to combine both approaches after compared the performance of FSCL neural net with K-Means. Hruschka and Natter (1999) also explored feed-forward neural nets and K-Means for cluster-based market segmentation.

3. Model Description

The clustering method proposed in this study is intended to adopt hierarchical concept into the K-Means procedure. This proposed method attempts to cluster data with higher similarity within clusters and can be simply divided into three steps: (1) initialization, (2) iteration, and (3) merging process. The details of each process will be illustrated as follows:

3.1 Initialization

The first step of the proposed clustering method is to randomly pick *m* objects as initial seeds where *m* should be larger than *k,* the predefined number of cluster. Then assign rest of objects by object order to the nearest initial seed and form m initial clusters. This step is similar to the first step of K-Means algorithm. The only difference between these two is that, instead of forming k clusters, we assign all objects to m initial clusters. After the first process, all objects are individually assigned to its nearest cluster, and the centroid of each cluster is calculated as the average of the objects within that cluster. Thus, m initial clusters and centroids are generated in this stage.

3.2 Iteration

The second step is to reassign each object (by object order) to the nearest cluster with minimum distance between the object and the centroid of the cluster. If any new assignment is occurred, then recalculate the centroid of the cluster. The process of reassigning all of objects is called iteration, and the process of iteration should be repeated until no object is moved between clusters. While this process is finished, *m* clusters are produced (i.e. M-Means).

3.3 Merging Process

m clusters are formed in previous step. Because *m* is larger than *k*, we decrease the number of clusters by 1 in each iteration until the number reaches *k*. In this process, we develop a function, Predict_{SE}, and use it to help us to decide which pair of the clusters should be merged.

Predict_{SE} is a measurement function to predict how SE value changes after clusters merge. Predict_{SE} value of two clusters is defined as

$$\text{Predict}_{SE}(i,j) = \text{Dis}(i,j)^2 * N(i) * N(j) \,/\, [N(i)+N(j)] \tag{1}$$

Here Dis(i,j) denotes the Euclidean distance between the centroid of cluster i and that of cluster j. N(i), and N(j) denote the object number within cluster i and j, respectively. For example, N(i) equals to 8 if cluster i includes 8 objects.

If there are two clusters (i and j) and the Euclidean distance between the centroids of both clusters is Dis(i,j), thus, the centroid of the new cluster, c, is located between i and j while cluster i and cluster j are combined, and it could be calculated as

$$C_c = (\Sigma X^i + \Sigma X^j) / [N(i)+N(j)] \quad (2)$$

Here C_c denotes the centroid of the new cluster c, and 'ΣX^i' is the sum of all objects within cluster i. Because the centroid of cluster i - C_i -is the mean of all objects within cluster i, C_i equals (ΣX^i) / N(i). Thus,

$$\Sigma X^i = C_i * N(i) \quad (3)$$

As 'ΣX^i' in equation (2) is replaced with equation (3), the centroid of the new cluster - C_c - could be transferred as equation (4).

$$C_c = [C_i * N(i) + C_j * N(j)] / [N(i)+N(j)] \quad (4)$$

Thus, the distance between the centroids of c and i (denoted as Dis(i,c)) could be calculated as equation (5).

$$\begin{aligned} Dis(i,c) &= \| C_c - C_i \| \\ &= \| \{ [N(i)*C_i + N(j)*C_j] / [N(i)+N(j)] \} - C_i \| \\ &= \| [N(i)*C_i + N(j)*C_j - N(i)*C_i - N(j)*C_i] / [N(i)+N(j)] \| \\ &= \| N(j)*C_j - N(j)*C_i \| / [N(i)+N(j)] \\ &= \| C_j - C_i \|*N(j) / [N(i)+N(j)] \\ &= Dis(i,j)*N(j) / [N(i)+N(j)] \quad (5) \end{aligned}$$

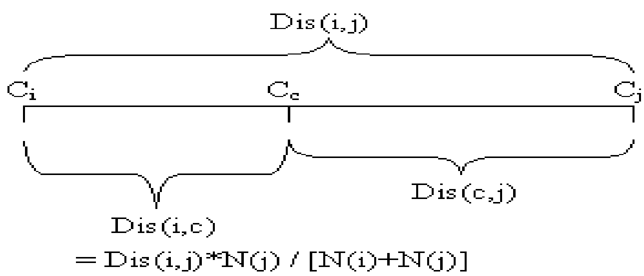

Figure 1. distance between the centroid of the merged cluster and that of the new cluster

Once we know the number of objects and the centroids of two selected clusters, we can calculate the centroid of the new cluster, which is produced by merging these two clusters. Similarly, the Euclidean distance between the centroids of clusters c and j can be calculated by equation (6).

$$Dis(c,j) = Dis(i,j)*N(i) / [N(i)+N(j)] \quad (6)$$

If we know N(i), N(j), C_i, and C_j, we can compute Dis(i,c) and Dis(c,j). Also, SE of a cluster is the sum of squared Euclidean distance between objects and centroid. If we replace each object by the centroid where the object belongs to, SE value of the cluster generated from clusters' combination can be predicted as

$$Predict_{SE} = Dis(i,c)^2*N(i) + Dis(c,j)^2*N(j) \quad (7)$$

Here we replace respectively 'Dis(i,c) ' and 'Dis(c,j) ' of equation (7) by equation (5) and equation (6), then equation (7) can be transferred to equation (8) and then equation (8) can be used to predict the SE value of the cluster c.

$$= Dis(i,j)^2*N(j)^2*N(i) / [N(i)+N(j)]^2 + Dis(i,j)^2*N(i)^2*N(j) / [N(i)+N(j)]^2$$

$$= [Dis(i,j)^2*N(j)^2*N(i) + Dis(i,j)^2*N(i)^2*N(j)] / [N(i)+N(j)]^2$$

$$= \{ Dis(i,j)^2*N(j)*N(i)*[N(i)+N(j)] \} / [N(i)+N(j)]^2$$

$$= Dis(i,j)^2*N(j)*N(i) / [N(i)+N(j)] \quad (8)$$

Since $Predict_{SE}$ function is used to predict SE value of the cluster, which is produced by combining two selected clusters, it helps us to choose which two clusters should be merged. As mentioned above, SE value is the measurement of clustering algorithm. A clustering result with the smallest SE value is what we are trying to achieve. Thus, the pair of clusters- that has the smallest $Predict_{SE}$ value among all will be chosen in this process.

The proposed clustering method is composed of the aforementioned three steps (e.g. initialization, iteration, and merging). The step of initialization is performed first, and iteration procedure follows next and last comes the process of merging. In fact, the processes of iteration and merging will be repeated back and forth until k clusters are generated. To be more specific, after the fist m clusters are produced, the merging process will decrease the cluster number by 1, which means that m clusters will be reduced to m-1 once merging process performs. Then the m-1 clusters and centroids will be regarded as another input for the process of iteration, and next continues the merging process after iteration. These two processes- iteration and merging- will be repeated until m is reduced to k. The complete details of the proposed clustering algorithm are provided as *Algorithm 1* and Figure 2.

Algorithm 1. *Proposed Clustering Method*

1. Randomly pick m objects as initial seeds.
2. Assign the rest of the objects (by object order) to the closest cluster, and update the centroids after each assignment.
3. Re-Assign all of the objects (by object order) to the closest cluster, and recalculate related centroids once object is moved from one cluster to another.
4. Repeat step 3 until no object could be moved.
5. Stop, if m was reduced to k; or else go to step 6.
6. Calculate $Predict_{SE}$ value of each pair of clusters, and merge clusters, which produce the smallest $Predict_{SE}$ value among all.
7. Go to step 3.

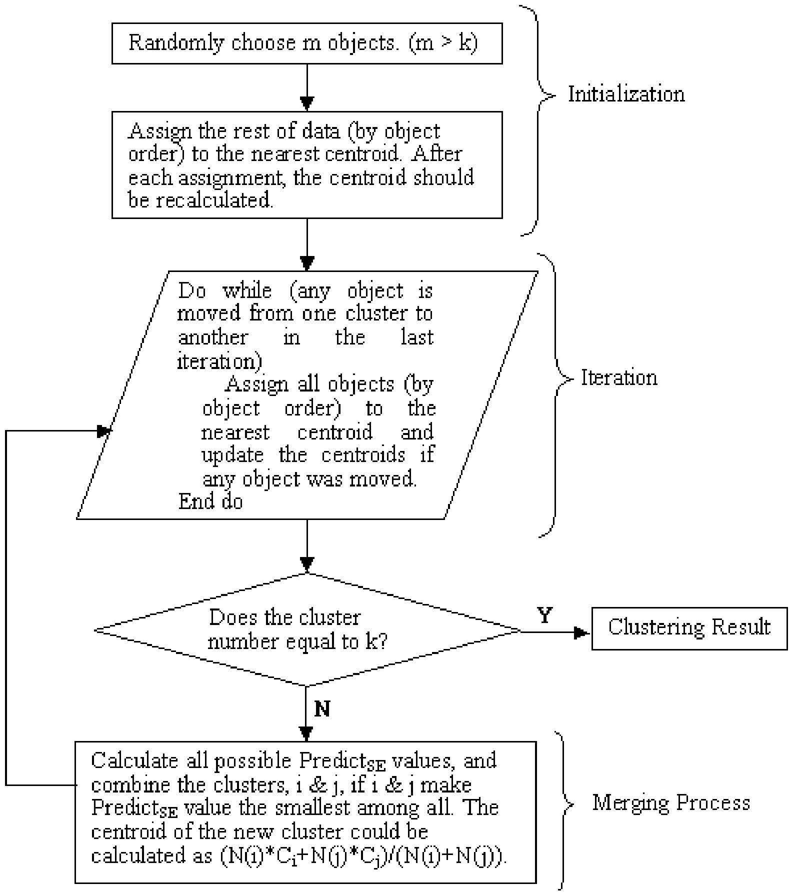

Figure 2 Flowchart of the Proposed Algorithm

4. Experimental Study

To explore how the algorithm works, the comparison of the proposed method and K-Means algorithm is given in this section. Here we take SE value to be a measurement as scholars did (e.g. Krishna & Murty 1999, Hruschka & Natter 1999, and Bandyopadhyay el at. 2001). Also, the robustness of an algorithm is a critical issue. For example, Chung and Gary (1999) said that robustness was one of the critical issues concerning to the quality of an algorithm. Furthermore, Pena et al. (1999) utilized standard deviation (Sd) as a measurement to check whether clustering results generated by the same algorithm are similar or greatly different. So, we take Sd value as another measurement to see how robust they are.

First, we choose 5 well-known datasets, which are Rusipini, Ionosphere, Lung-Cancer, Wine, and Time Series data. Ruspini data is provided by Dr. Jose M. Pena, and others are generated from UCI KDD Archive. All of the attributes for each data set are numeric values. As Table 1 shows, Ruspini data includes 75 objects with 2 dimensions, and the exact number of clusters is 4. Lung-Cancer data includes 32 objects with 56 dimensions, in between, 5 objects have missing value. Therefore, we exclude these 5 objects, and the rest of the data contains 27 objects. Information of all datasets is listed in Table 1.

Table 1 Data Description

Name	Ruspini	Ionosphere	Lung-Cancer	Wine	Synthetic Control Chart Time Series
No. of object	75	351	27	178	600
No. of dimensions	2	34	56	13	60
No. of clusters	4	2	3	3	6

Second, we let m equal 4 k while using proposed clustering algorithm. For example, we perform the proposed algorithm with m=16 (4×4) for Ruspini data. Accordingly, m equals 8, 12, 12, and 24 while dealing with Ionosphere, Lung-Cancer, Wine, and Time Series data sets, respectively.

Third, because K-Means is sensitive to object order (Huang 1998) and initial seeds (Pena et al. 1999; Dash 2001), thus, we randomly re-order each data set 500 times and also apply each method 50 times for each order of each data. Take figure 3 as an example, for each object order, these two algorithms are performed 50 times individually, and the average of 50 SE values (as SE_i in figure 3) is regarded as an expectant SE value with respect to this object order. Thus, 500 Sd and averaged SE values can be generated for each single data, and these two indexes will help us to understand the robustness of the algorithms and the intra-similarity of the produced groups.

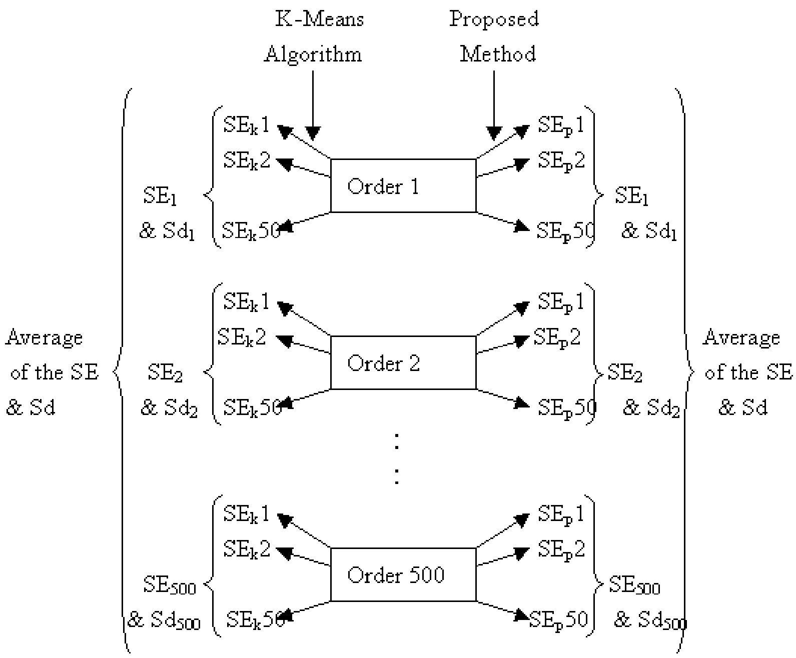

Figure 3 Experimental Design

The results of the experimental study are demonstrated in Table 2. As Table 2 shows, the SE and Sd values generated by the proposed algorithm with respect to any data are lower than those generated by K-Means. For example, the average SE value generated by K-Means is 1013037.9375 comparing to 944783.7500 generated by the proposed algorithm for Synthetic Control Chart Time Series. Similarly, the Sd value for Synthetic Control Chart Time Series data generated by the proposed algorithm is 1852.7645, which is much lower than that of K-Means, 61217.6523. Moreover, for Ionosphere data, the Sd value generated by K-means algorithm is quite small (e.g. 266.7418), but the one generated by the proposed algorithm is even decreased in a credible way (e.g. 0.0133).

Table 2 Experimental Result 1

		Ruspini	Ionosphere	Lung-Cancer	Wine	Synthetic Control Chart Time Series
Averaged Square Error (SE)	K-Means Algorithm	33184.7617	2525.1680	437.0696	2426721.2500	1013037.9375
	Proposed Clustering Method	12882.4277	2419.3850	403.2434	2370687.2500	944783.7500
Averaged Standard Deviation (Sd)	K-Means Algorithm	18191.7734	266.7418	18.8533	104945.4531	61217.6523
	Proposed Clustering Method	10.0332	0.0133	5.1980	0.9322	1852.7645

In related marketing research, customer data often includes thousands customers. Since this algorithm is proposed to improve segmentation performance of CRM and/or related marketing practice, we here choose a well-known large dataset, letter recognition dataset, for a further comparison. Letter recognition data includes 20,000 objects, and 16 numeric features. The dataset has no missing value and is classified into 26 clusters.

For letter recognition data, we randomly re-order objects for 20 times to diminish the influence from object order. For each order, we perform the proposed algorithm and the K-Means method to calculate SE value individually, and then we can calculate and compare Sd value of 20 SE values. The result of the experimental study is provided in Table 3.

Table 3 Experimental Result 2

	K-Means Algorithm	Proposed Method
Averaged SE Value	619797.2	611331.2
Standard Deviation	4319.442	167.8429

In table 3, the proposed algorithm outperforms the K-Means method. The average SE value generated by K-Means is 619797.2, and that generated by proposed method is 611331.2. Thus, the similarity within clusters of the result generated by the proposed algorithm is higher than that generated by K-Means. Similarly, because the Sd value generated by the proposed algorithm (e.g. 167.8429) is much lower than that of K-Means method (e.g. 4319.442), the proposed algorithm is more robust than K-Means method.

According to the experimental results, the proposed method outperforms K-Means algorithm, even when it is applied to a large data set. Therefore, we believe that the proposed algorithm can generate a better clustering result than K-Means does, because the proposed clustering method is more robust, and is not greatly sensitive to object order as K-Means does.

4. Conclusion

The clustering algorithm proposed in this study combines K-Means method and the concept of the merging process of hierarchical clustering approaches. Iteration process (extracted from K-Means) and merging process (a concept of hierarchical approaches) are performed back and forth, until m is reduced to k. In this experimental study, the results reveal that this clustering method could produce a better result with lower SE value. Also, this proposed algorithm is more robust than K-Means method, because Sd of the proposed method is much lower than that of K-Means. As SE and Sd value decrease, the algorithm increases the similarity within segments and is less affected by the order of customer data Therefore, the clustering algorithm could be a good alternative for related business practices, especially for CRM and marketing segmentation.

Although we conclude that proposed clustering method could produce a better clustering result than K-Means method does, there are still a few concerns for further research. First, since the parameter- m- depends on a user's setting, we don't know whether the best rule of setting this parameter exists. If m is too large, then part of the computation time will become time-consuming. On the other hand, the algorithm might not obviously perform better than K-Means if m is too small. As a result, parameter setting is a dilemma for users. Next, Although we let m equal 4×k and the experimental results are good, we can't mathematically prove this setting method is the best one. Also, the best m might depend on the data characters (ex, number of objects, number of dimensions, and data structure, etc.), and a general setting method of m still needs to be addressed.

Second, there are many clustering methods focusing on the increase in similarity. However, "which algorithm is the best one" is still an issue and a further comparative study still needs to be done. Finally, the first m points of the proposed algorithm is generated randomly in the first step, but some initialization methods such as KKZ (Katsavounidis et al. 1994) or Kaufman Approach (Kaufman and Rousseeuw 1990) could still be applied to this sub-process. Therefore, we wonder whether the clustering result would be improved by such applications as we mentioned above. The answer depends on broadly comparisons and further explorations.

References

[1] Al-Daoud, M. B.,and S. A. Roberts (1996), "New Methods For the Initialisation of Clusters," *Pattern Recognition Letters*, Vol.17, pp.451-455.

[2] Balakrishnan, P. V. (Sundar), M. C. Cooper, V. S. Jacob, and P. A. Lewis(1996), "Comparative Performance *of* the FSCL Neural Net and K-Means Algorithm for Market Segmentation," *European Journal of Operational Research*, 93 (2), pp.346-357.

[3] Bandyopadhyay, S., U. Maulik, and M. K. Pakhira (2001), "Clustering Using Simulated Annealing With Probabilistic Redistribution," *International Journal of Pattern Recognition and Artificial Intelligence*, 15 (2),

pp.269-285.
[4] Battista, P., and D. Verhun (2000), "Customer Relationship Management," *CMA Management*, 74 (4), pp.34-37.
[5] Chen, Lei-da, T. Sakaguchi, and M. N. Frolick (2000), "Data Mining Methods, Applications, and Tools," *Information Systems Management,* 17 (1), pp.65-70.
[6] Cholewka, K. (2001), "Tiered CRM: Serving Pip-squeaks to VIPs," *Sales And Marketing Management*, 153 (4), pp.25-26.
[7] Chung, H. M., and P. Gary (1999), "Special Section: Data Mining," *Journal of Management Information Systems*, 16 (1), pp.11-16.
[8] Crosby, L. A., and S. L. Johnson (2000), "Customer Relationship Management," *Marketing Management*, 9 (3), pp.4-5.
[9] Dash, M., H. Liu, X. Xu (2001), " '1+1<2': Merging Distance and Density Based Clustering," *Proceedings of Seventh International Conference on Database Systems for Advanced Applications*, p.p. 32-39.
[10] Garcia-Escudero, L. A., and A. Gordaliza (1999), "Robustness Properties of K Means and Trimmed K Means," *Journal of the American Statistical Association*, 94 (447), pp.956-969.
[11] Green, P. E., and A. M. Krieger (1995), "Alternative Approaches to Cluster-Based Market Segmentation," *Journal of the Market Research Society*, Vol.3, pp.221-239.
[12] Han, J., and M. Kamber (2001), Data Mining: Concepts and Techniques, *Morgan Kaufmann Publishers*, San Francisco.
[13] Hruschka, H., and M. Natter (1999), "Comparing Performance of Feedforward Neural Nets and K-Means for Cluster-Based Market Segmentation," *European Journal of Operational Research*, 114 (2), pp.346-353.
[14] Huang, Z.hexue (1998), "Extensions to the K-Means Algorithm for Clustering Large Data Sets with Categorical *Values*," *Data Mining and Knowledge Discovery*, 2 (3), pp.283-304.
[15] Huth, R. (1996), "An intercomparison of computer-assested circulation classification methods," *Int. J. Climatol.*, Vol.16, pp.893-922.
[16] Katsavounidis, I., C. Kuo, and Z. Zhang (1994), "A New Initialization Technique for Generalized Lioyd Iteration," *IEEE Signal Process Lett.*, 1 (10), pp.144-146.
[17] Kaufman, L., and P. J. Rousseeuw (1990), Finding Groups in Data: An Introduction to Cluster Analysis, *Wiley*, Canada.
[18] Kohli, R., and F. Piontek, et al. (2001), "Managing Customer Relationships Through E-business Decision Sopport Applications: A Case of Hospital-Physician Collaboration," *Decision Support Systems,* 32 (2), pp.171-187.
[19] Krishna, K., and M. N. Murty (1999), "Genetic K-Means Algorithm," *IEEE Trans. Systems, Man, and Cybernetics-Part B: Cybernetics*, 29 (3), pp.433-439.
[20] Linde, Y., A. Buzo, and R. Gary (1980), "An Algorithm for Vector Quantizer Design," *IEEE Trans. Comm.*, 28 (1), pp.84-95.
[21] MacQueen, J. (1967), Some Methods for Classification and Analysis of Multivariate Observations, *Proc. 5th Berkeley Symp. Math. Stat. and Prob.*, 1: pp.281-297.
[22] Milligan, G. W. (1980), "An Examination of the Effect of Six Types of Error Perturbation on Fifteen Clustering *Algorithm*," *Psychometrika*, Vol.45, pp.325-342.
[23] Murthy, C. A., and N. Chowdhury (1996), "In Research of Optimal Clusters Using Genetic Algorithms," *Pattern Recognition Letters*, Vol.17, pp.825-832.
[24] Pena, J. M., J. A. Lozano, and P. Larranaga (1999), "An Empirical Comparison of Four Initialization Methods for the K-Means Algorithm," *Pattern Recognition Letters*, Vol.20, pp.1027-1040.
[25] Peppard, J. (2000), "Customer Relationship Management (CRM) in Financial Services," *European Management Journal*, 18 (3), pp.312-327.
[26] Peppers, D., and M. Rogers (1999), "When Extreme Isn't Enough," *Sales and Marketing Management,* 151 (2), pp.26-27.
[27] Ryals, L., and S. Knox (2001), "Cross-Functional Issues in the Implementation of Relationship Marketing Through Customer Relationship Management," *European Management Journal,* 19 (5), pp.534-542.
[28] Selim, S. Z., and M. A. Ismail (1984), "K-means Type Algorithms: A Generalized Convergence Theorem And Characterization of Local Optimality," *IEEE Trans. Patt. Anal. Mach. Intell.*, 6 (1), pp.81-87.
[29] Sharp, B., and A. Sharp (1997), "Loyalty Programs and Their Impact on Repeat-Purchase Loyalty Patterns," *International Journal of Research in Marketing,* 14 (5), pp.473-486.
[30] Storey, C. K. (2001a), "Relationship Connection," *Credit Union Management*, 24 (2), pp.50-53.
[31] Storey, C. K. (2001b), "Don't Get Stuck," *Credit Union Management*, 24 (7), pp.38-41.
[32] Tou, J., and R. Gonzales (1974), Pattern Recognition Principles, *Addison-Wesley*, Reading, MA.
[33] Yim, C. K., and P. K. Kannan (1999), "Customer Behavioral Loyalty: A Segmentation Model and Analysis,"

Journal of Business Research, 44 (2), pp.75-92.