Association for Information Systems
AIS Electronic Library (AISeL)

PACIS 1995 Proceedings

Pacific Asia Conference on Information Systems (PACIS)

December 1995

Enhancement of Degree of Generalization in Neural Clustering Systems and Statistical Unterpretation

Jong Choi
Sangmyung Women's University

Kyoung Bae
Sangmyung Women's University

Dae Lee
Sangmyung Women's University

Yoon Chung
Hankuk University of Foreign Studies

Enhancement of Degree of Generalization in Neural Clustering Systems and Statistical Interpretation

Jong U. Choi, Kyoung Y. Bae, & Dae W. Lee
Sangmyung Women's University

Yoon Chung
Hankuk University of Foreign Studies

ABSTRACT

Researchers in neural network community have claimed that the classification accuracy of neural network is higher or at least equal to conventional statistical methods. Based on the claims, many experimental research have been conducted and proved that the degree of generalization obtained in neural network's training scheme is better than other prediction/classification methods.

In a series of research conducted to develop an automated credit evalaution systems, various algorithms of neural networks have been experiemnted. In the experiments, back propagation algorithm and quickprop algorithms, with the increase of training data size, did not achieve classificaiton accuracy we expected. Therefore, learning vector quantization (LVQ) and generalized vector quantization (GLVQ) algorithms were employed to enhance the degree of generalization. Even though much better performance was obtained when GLVQ algorithm was employed, the prediction accuracy was still very low, around 50-60%.

The conclusion of this research is that the prediction accuracy of neural networks strongly depends on data characteristics and training environments. When the network algorithm is applied to noisy data set for training, that is, when there could not be found any meaningful statistical relation, higher accuracy might not be easily achieved. To the contrary, when there is a sound data set involved in the training, very high accuracy can be achieved.

1. Introduction

Neural-based training has been one of the hottest topics in AI research community. The technique has been successfully applied to various fields of classification: character recognition, signal processing, acoustic sound recognition, automobile licence plate recognition, business classification, and so on. For example, Kang et al. [1994a, 1994] applied neural network training to Korean character recognition in automobile licence plate and found that the accuracy rate of the system is much higher than that of an expert system. Even in business classification problem requiring experience of many years, the neural network systems have shown that the classification accuracies of the neural system are better than or at least equal to those of conventional systems or expert systems.

However, in a series of research on neural network application to credit evaluation system, it was found that the prediction(classification) accuracy strongly depends on the characteristics of the data included in training and testing. In other words, neural network approach to classification(prediction) is not a panacea to classification task, possibly applicable to every pattern classification problem. In fact, it is quite likely that neural-based training system, though many research results have claimed much better classification accuracy, can be worse than other classification techniques [Weigend and Gershenfeld, 1994]. In a series of experiments of Weigend and Gershenfeld [1994], conventional time series analysis technique showed better performance than neural-based systems in predicting future events with noise. To the contrary, in the cases of regular patterns, neural- based system showed much higher accuracy than others for predicting patterns.

In this research, the generalized learning vector quantization (GLVQ) algorithm [Pal, Bezdek, and Tsao, 1992] was applied to credit evaluation problem. GLVQ algorithm is a modified version of Kohonens' learning vector quantization (LVQ) algorithm [1991].

Originaly, this research was initiated to develop an automated credit evaluation system based on neural network training technique. The first experiment with a small set of training data was successful, showing almost 80 percent classification accuracy. However, as the size of training data and testing data increases, the system showed very low classification accuracy. To investigate the relationship between data characteristics and classification accuracy of the neural systems, the GLVQ algorithm was applied to two sets of credit evaluation data.

Although many research results conducted in neural network community have claimed classification accuracy much higher than or at least equal to statistical methods, we could not achieve such a high classification accuracy on the real world data set we have. The successful results of previous researches might have been obtained only because the network was trained with sound data set, having little number of noisy data or being large enough to control noisy data. As a conclusion, the neural system might not be the best solution, or a panacea to classification problem.

In this research, a series of research on neural network application to credit evaluation problem is introduced and statistical properties of the neural system are discussed. In the following section, credit evaluation system and neural systems built on back propagation algorithm are introduced to help understanding domain problem. In section three, a general concept of GLVQ algorithm is introduced, and in section four experimental results of LVQ algorithm and GLVQ algorithm are described in detail. Section five is reserved for conclusion and future research. In addition, as the task of future prediction, in cases of neural network's pattern recognition, are conducted mainly based on proper classification techniques of past data, the accuracy of classification is interchangeably used for the accuracy of prediction.

2. A Neural-Based Credit Evaluation System

Credit evaluation is one of the most important and difficult tasks usually assigned to experienced officers in credit card companies, mortgage companies, banks, consumer goods companies and other financial institutes. Traditionally, credit scoring has been the most widely used method in which applicant's credit is evaluated by picking up appropriate score corresponding to categories of evaluation value, than by summing up into total credit for thresholding. Recently, various method have been introduced to replace the credit scoring system and to provide more objective and convenient tools : statistical method [Majone, 1968; Apilado, Warner, and Dauten, 1974; Edelstein, 1975; Muchinsky, 1975; Beranek, Taylor, 1976; Borzar, 1978; Capon, 1982]. Induction trees(ID3, C3) [Carter and Catlett, 1987], expert system approach [Dungan, 1982; Dungan and Chandler, 1985; Kastner, Apte, Griesmer, Hong, Karnaugh, Mays, and Tozawa, 1986; Messier and Hansen, 1987; O'Leary, 1987] and the neural network researchers have shown that prediction accuracies of the neural network system, that is, the degrees of generalization, are better than or at least equal to those of the statistical methods [Kim, 1992; Odem and Sharda, 1990; Schumann and Lohrbach, 1992]. Therefore, many researchers have devoted their research efforts to enhance the degree of generalization to achieve higher level of prediction accuracy. In determining the degree of generalization, involved are many internal and external factors : the network architecture(number of hidden nodes, input nodes, hidden layers, initial weights, learning rate, momentum, etc.), training algorithm(back propagation, self-organization map, quickprop algorithm, activation function, etc.) and composition of training data set and test data set. Researchers experimented with various architectures by modifying factor values and learning algorithm.

There are some experimental reports on the relationship between training data set and the degree of generalization, with recommendation of the ways to achieve higher generalization capability. Whitley and Karunanithi [1991] proposed a partitional learning strategy in which the training space is divided into a set of subspace according to the data characteristics and then each subspace is trained using a separate network. In the data selection step, emphasized is decision boundaries and the central tendencies of decision regions. In the test of 'two-spiral' problems, they achieved almost 100 percent correctness ratio, using the border patterns. Higher accuracy, with partitioning training data set into separate subsections, may be achievable, because partitioning can increase homogeneity of the training set. Fu and Chen [1993] investigated the sensitivity of input vectors on generalization capability, and found that the norm of Jacobian matrix measures the sensitivity of the network performance with respect to its vector and that good generalization must imply insensitivity to changes in the input vectors.

This research was initiated to develop an integrated on-line credit evaluation system which would monitor system performance and enhance prediction accuracy through constant feedbacking customer's credit data. Especially, the neural network mechanism was adopted as a credit evaluating processor in this research. The neural network could predict the output values by nonlinear mapping through the hidden layer, even though it didn't know the direct relations between the input values and the output values.

E LTD. is one of leading companies in the Korea fashion business. A credit card system of this company is adopted to achieve 'Big Share' in fashion market. Recently, the number of card holders of this company has reached 180,000 and every month the number of overdue or delinquent credits reached 3,500 cases. Such delinquent customers inflict a serious loss to the company and thus the company had to devise a measure to solve the financial problem caused by continuously accumulated bad debts. One of the ideas popped up was to develop an automated credit evaluation system which continuously monitors evaluation system's performance and then can enhance the prediction accuracy through learning from customer's credit data.

S LTD is an another leading credit card company which holds a big share in Korea market. The customer's behaviour of E LTD and S LTD is almost the same, in terms of credit standing. Even though the two companies employ different application variables, accordingly different application forms, the contents were almost the same.

The neural network training system, as usual of the back propagation systems, consists of three layers : input layer, hidden layer and output layer. From the customer's records, the eight variables which were believed to have a strong relationship with customer's credit were derived as 'credit factors' : age, sex, marital status, occupation, organization, job position, residential condition, residential area. Selection of input variables for the system's training, that is, selection of critical factors significantly influencing on customer's creditability should be determined by consideration of customer's behaviour, social custom, and statistics. In this sense, the factors included in the current system reflect many features of Korean customers and social practices, and thus factors included in the current system might be much different from factors included in the system developed in other countries. For example, in the study of American loan application, occupation, length of employment, marital status, race and income level are important considerations[Capon, 1982], but work place dose not have a significant impact on credit evaluation. In contrast, work place might be the most important factor in determining an individual's credit status. Also, residential area might be very important factor, which was proved to be unimportant at all.

According to the number of overdue payment, 'credit status' was divided into two status such as 'good' and 'bad'. When the customer's payment is not overdue or the number of overdue payment is less than 3 months, he or she was classified into 'good' credit status. When the number of overdue payment exceeded 3 months, the customer was classified into 'bad' credit status.

Data Set	Back Propagation		Quickprop	
	# of Epoch	Degree of generalization	# of Epoch	Degree of generalization
20/20	22,644	19/40(47.5%)	26	23/40(42.5%)
30/30	42,212	22/40(55%)	49	24/40(60%)
40/40	238,802	26/40(65%)	168	24/40(60%)
50/50			267	27/40(67.5%)
60/60			338	23/40(57.5%)
70/70			467	23/40(57.5%)
80/80			488	28/40(70%)

Table 1. Classification Accuracy on E LTD. data (BackPropagation and Quick-Prop)

In the beginning stage of this research, back propagation algorithm was employed. Many experiments with the back propagation algorithm showed that the system with more than 40 training data would not which convergence state within a reasonable time and thus another efficient algorithm should be devised. Later, employed was the quickprop algorithm, an advanced form of back propagation, as a learning mechanism. 'Quickprop' algorithm suggested by Fahlman[1988] is well-known for speeding up convergence by jumping out directly the parabolic error space to the minimum point of the parabola. In this algorithm, the error defined as $\partial E/\partial w(t-1)$ is kept and then, for each weight, the weight change measured by the difference between current weight slope and previous weight slope is used for determining a parabola.

As predicted and assured by researchers [Fahlman, 1988], the quickprop algorithm effectively and quickly reduced total error, and thereby enabled the system to reach convergence in a reasonable time limit. As shown in Table-1, ordinary back propagation algorithm required more than 230,000 epochs to reach convergence state in which the degree of generalization was 65%. In contrast, the quickprop algorithm needed only 160 epochs to reach the convergence state in which the degree of generalization was measured around 60%, a slightly lower value than ordinary back propagation algorithm. When the number of training data set increases to 100 (50 bad creditors and another 50 good creditors), the back propagation system did not stop running. That is why the research, in back propagation learning system, could not extend testing the generalization capability beyond 80 cases. To the contrary, the quickprop algorithm with 100 training data easily reached the convergence state at the epoch of 267 and showed a little enhanced prediction capability, 67%. The same test results were obtained when the test of classification accuracy were conducted on S LTD data set. As shown in Table-2, the classification accuracy was less than 60%, which is too low for field application.

# of Epoch	Training Data	Test Data	Degree of Generalization
100	60/60	50/50	43%
195	100/600	50/50	46%

Table 2. ClassificationAccuracy on S LTD. data(Quick-Prop)

3. Generalized Learning Vector Quantization (GLVQ)

Vector quantization(VQ) is defined as a technique which "searches for small but representative set of prototypes, which we can then use to match sample patterns with nearest neighbor techniques," [Kong and Kosko, 1992, p.1]. It is a technique developed for solving data encoding problem leading to minimization of reconstruction error in data compression and decompression [Ritter, Martinnetz, and Schulten, 1992]. In searching for an efficient data decompression skill, one seeks to describe as faithfully as possible "the distribution of data points in a high-dimensional space, using only a space of lower dimension" [Ritter, Martinnetz, and Schulten, 1992, p.238]. That is, the most efficient projection of original data onto lower dimensional planes can yield the smallest project error. The technique has been modified and successfully applied and to various fields : image classification [Cannon, Dave and Bezdek, 1986], phoneme signal processing[Kong and Kosko, 1992], and travelling sales person(TSP) problem [Rose, Gurewitz, and Geoffrey, 1993].

Clustering through VQ is accomplished by partitioning the patterns $x \in R^n$ into k decision classes $\{D_j\} \in R^m$, the prototypes or reference vectors:

$$R^n = \bigcup D_j, \text{ and } D_i \cap D_j = \emptyset \text{ for } i \neq j.$$
$$x \in D_1 \quad \text{if } d(x, s_1) < d(x, s_2)$$
$$x \in D_2 \quad \text{if } d(x, s_1) > d(x, s_2),$$

where $d(x_i, s_j)$ is defined as the distance measure between the pattern x_i and prototype s_j. S_1 and s_2 are prototypes belonging to the decision classes D_1 and D_2, respectively.

The VQ system attempts to find appropriate decision class (D_1, D_2, ..., D_k) and centroids ($\overline{S_1}, \overline{S_2}, .. \overline{S_k}$), from the patterns ($X_1$, X_2, ..., X_p), the pattern belongs to. In the view of data compression, the patterns x_i will not completely vary, but rather will be correlated to next patterns. Thus, the essential problem of VQ technique is to find mapping functions from the patterns to hidden variables $r_1, r_2, ..., r_m$, for $M < P$, with minimal variances. The variables r_i provide a more economical description of the observed phenomenon. In the linear discriminant functions [Kong and Kosko, 1992; Kosko, 1991], the function behaves as a separating hyperplane in the pattern space R^n, that is, setting up K-dimensional hyperplane lying in the N-dimensional data space. The variables r_i can account for the total data variation. However, if the actual distribution of data points is deviated from the hyperplane, the description resulting from a projection on the principal axes of the distribution will be worse [Ritter, Martinetz, and Schulten, 1992; Cichocki and Unbehauen, 1993]. To overcome this problem, the linear principal axes or hyperplanes are replaced by curved surfaces, which may provide a better description of nonlinear data distributions. This can be interpreted geometrically as "a minimization of the mean-squared perpendicular distance $d(x, \bar{S_i})^2$ between the data points and the hyperplane" [Ritter, Martinetz, and Schulten, 1993, p.247].

<u>Learning vector quantization (LVQ)</u>, suggested by Kohonen, is considered as an approximation procedure for the computation of principal curves, surfaces, or higher-dimensional principal manifolds [Ritter, Martinetz, and Schulten, 1993]. The LVQ system tries to discover cluster substructure hidden in unlabeled N-dimensional data and extract M-dimensional features. The prototypes $S=\{S_1, S_2, ..., S_k\}$ are a array of unknown cluster centers $S_i \in R^m$ for $1 \leq i \leq k$. In LVQ, learning refers to finding values for the $\{S_{ij}\}$ [Pal, Bezdek, and Tsao, 1993]. When an input vector X_i is submitted to the system, the distance between the input vector and prototypes $d(X, S_i)$ is calculated and then the prototype with the shortest distance becomes a winner. The next step is to update the centroid of the prototype using update rules. The typical LVQ rule of finding the winner node and update is as following:

$$\| X_k - S_{i,t-i} \| = \min_{1 \leq i \leq k} \{ \| X_k - S_{j,t-1} \| \}$$

for finding.

$$S_{i,t} = S_{j,t-1} + \alpha(X_k - S_{j,t-1}) \text{ for updating.}$$

Although the LVQ algorithm has some nice theoretical foundation, it suffers from a serious problem : initialization problem. As the initial position of centroid $S_{i,0}$ have too strong influence on subsequential position updates, especially when they are outside the convex hull of the input data [Kang, Hwang and Yoo, 1994], it may not produce any meaningful clusters[Pal, Bezdek, and Tsao, 1993]. Also, as the winner node only update its position, the result of clustering might be biased by the gravitational force of winners.

To overcome these problems, Pal, Bezdek and Tsao [1993] suggested the GLVQ algorithm which updates either all the centroids of prototypes or none, for each new input vector. When there is a perfect match to the winner node, no node is updated.

The updates rule of GLVQ is

$$S_{i,t} = S_{i,t-i} + \alpha(X_k - S_{i,t-i}) \frac{D^2 - D + \| X_k - S_{i,t-i} \|^2}{D^2}$$

$$S_{i,t} = S_{i,t-i} + \alpha(X_k - S_{i,t-i}) \frac{\| X_k - S_{i,t-i} \|^2}{D^2}$$

$$(r \neq i)$$

where i is the best matching node, $D = \sum_{r=1}^{c} \| X_k - V_r \|^2$, $k=1, 2,.. n;\ r=1, 2, ...k$ and t is time.

4. Experiments with GLVQ

4.1 Comparison of LVQ and GLVQ Algorithms

A series of experiments were conducted on the data set of customer records obtained from S-insurance company. In the experiments it was found that LVQ algorithm was very sensitive to Alpha value, while GLVQ algorithm was not so sensitive. The correct clustering was obtained when the Alpha value and number of iteration were set 0.95 and 100, respectively. However, correct clustering was not obtained when the Alpha value was changed into 0.94, through the number of iterations was set more than 100. The other Alpha value less than 0.95 produced the same results: no clustering even with iterations more than 200, which implies that the network cannot converge into equilibrium state when the Alpha value exceeds a specific limit. In this case, the limit point of reasonable clustering is 0.95.

A reasonable clustering was obtained when the the Alpha values of the clustering were 0.95, and 0.80, with the iteration number of 100. Even with the Alpha value of 0.20, the network easily reached the convergence state with iteration of 200' at which the error size of the network (0.006) was reduced to much less than the predefined limit (0.01). When the Alpha value decreases to 0.15, the network slowed down to 295 iterations to get to the equilibrium state. Explicitly, there exists an important relationship between the number of iterations and Alpha value. To test the hypothesis, the Alpha value was set 0.10. The equilibrium state was obtained when the network cycled more than 500 iterations but less than 1000 iterations. The findings of the relationship between number of iterations and Alpha value might be summarized into the following equation.

$$f = nA$$

where n denotes the number of iterations and A denotes the Alpha value.

When experiments repetitively were conducted on other set of data, such as 40 samples (20 good, 20 bad credits), 80 samples (40 good, 40 bad credits), and 120 samples(60 good, 60 bad credits), the same results were obtained. From the test results, a generalized principle was derived that network behaviour of LVQ is very sensitive to the Alpha value, while the behaviour of GLVQ is not so sensitive to the Alpha value. Also, there exists an explicit relationship between the number of iterations and the size of Alpha value.

Interpretation of the test results is as following. The centroids of the clustering network in LVQ are at first located at a random initial places and subsequentially moves into the direction attracting input nodes through 1-nearest neighbor prototype principle. That is, the size of centroids movement is affected by a single input value. As a result, when the initial locations of clustering centroids are located far away from each other, it is very difficult to drag the centroids out of the convex hull into n-partitioned planes. Because of this, the initial Alpha value should be very large enough to move the clustering centroids into energy-minimum states, although the large number of network iterations has a slight impact on the clustering speed.

To the contrary, all the centroids of the GLVQ network is affected by input values, in proportion to the distance between the centroid and input value. Thus, the centroids moves into the direction of mass gravitational forces of all the input values. As a result, reasonable clustering can be obtained very quickly. The significant difference between LVQ algorithm and GLVQ algorithm come from the principle that a single input value does affect movement of all centroids in GLVQ, but a single nearest centroid in LVQ. In addition, the relationship identified in this research between the number of iterations and Alpha value can be explained in the equation:

$$A_t = A_0\left(1 - \frac{t}{T}\right)$$

where At is the Alpha value at the time t and A_0 is the initial Alpha value.

T denotes the maximum number of iterations of the network. In this equation, the initial Alpha value and the maximum number of iteration make a constant value. That is, when the initial Alpha value is large, the number of iteration should be small. To the contrary, when the Alpha value is small, the iteration number should be large enough to derive reasonable clustering.

4.2 Statistical Power of GLVQ Model

In the experiments with GLVQ, two sets of credit evaluation data were employed for performance comparison. In the first experiment, as shown in the figures below, the system showed very low classification accuracy. As the system was developed based on GLVQ algorithm, the system was not sensitive to the modification of learning parameter, alpha and number of iterations. As shown in Figure-1 and Figure-2 of E LTD case, the system partitioned 120 data consisting of 60 'bad' customer and another 60 'good' customer data into two clusters: cluster-1 and cluster-2. The data numbered 0 to 59 should be group-1, while the data numbered 60 to 119 should be in group-2. In other words, the data numbered 0 to 59 and the data numbered 60 to 119 should be not be in the same group to be cohesive. But, the clustering result is that the cluster-1 has 74 units of data including 33 data units from one group and other 41 data units from another group.

This means that clustering the credit data is not so meaningful for real world application, indicating that the data included in clustering does not have any meaningful relationship with each other in the same group. In other words, neural classification cannot impose any meaningful decision rules on clustered data. As shown in Table-4 the same thing was observed in S LTD case. Figure-1 and Figure-2 show that inconsistencies between natural clustering of

GLVQ algorithm and original credit clustering exist in case of S LTD data set. In Figure-1 and Figure-2, two different data set of the homogenous category belong to the same cluster while the other two data sets of the same category belong to the other cluster. For example, the shape of square and circle should not be in the same cluster. Irrespectively of the data characteristics, the credit data are randomly scattered. It implies that the natural clustering through GLVQ algorithm may not produce any meaningful classification result.

This might be attributable to the fact that the training data includes too much noisy data in it, mainly conflicting cases. For example, a department head of a business company usually earns better salary than other employees in the department and thus, the head is supposed to be much better in credit standing than others in the department. However, in the review of raw data it was found that the credit standing of employees is not correlated with income level of the employee. Even CEOs of business companies, though CEOs of large group companies are exceptional, were as bad as young undergraduates with less than one year's job experience.

5. Conclusion and Future Research

Neural network can be an efficient method for achieving prediction accuracy based on training of collected past data set. Generally, as claimed by neural network researchers, the network training algorithms is better than, or at least equal to conventional prediction methods such as regression methods, expert system approach, or machine learning mechanism.

However, we believe that the better performance of the network training systems can be achieved because the data set included in training and testing is noisy-low set. In other words, even though the prediction power of neural networks is excellent, it strongly depends on the data characteristics. In a series of experiments with noisy data set for developing automated credit evaluation systems, it was found that the non-linear projection of training data set into hyperplane cannot completely eliminate uncertainty included in white noise. The noise contained in the training data set can be explained as following. With the rapid increase of sales volume and credit market in Korea, many business companies have not imposed any restriction on credit card applicants. This is because, different from American companies with hundreds of years of experienced in financial market, Korean Companies pursue the goal of market penetration and market expansion through granting credit cards to any applicant without any scanning efforts. Therefore, the training data set obtained from credit companies does not have strong consistency, or statisitcal trends. That is, why any other prediction method cannot earn much better results than the results achieved by this research.

As a conclusion, very high accuracy cannot be achieved with noisy data.

Reference

Beranek, W. and W. Taylor, "Credit - Scoring Models and the Cut-off Point A Simplification", *Decision Sciences*, vol. 7, 1976.

Cannon, R.L., J.V. Dave, and J.E. Bezdek, "Efficient Implementation of the Fuzzy C-Means Clustering Algorithms," *IEEE Trans. on Pattern Analysis and Machine Intelligence* (Vol. 8, No.2), March, 1986.

Carter, C., and J. Catlett, "Assessing Credit Card Applications using Machine Learning", *IEEE EXPERT*, pp.71-79, Fall 1987.

Cichocki, A., and R. Unbehauen, *Neural Networks for Optimization and Signal Processing*, Wiley (West Sussez, England), 1993.

Schfield, and S. Ghosh, "Risk Assessment of Mortgage Applications with Neural Network Systems: An Update as the Test Portfolio Ages," *Proceedings of IJCNN 1990*, Vol.II, pp.II-479-482, 1990.

Fahlman, S. E., "Faster-Learning Variations on Back-Propagation: An Empirical Study," *in Touietzky D., T. Sejnowski, and G.Hinton (eds.) Proceedings of the '88 Connectionist Models Summer School*, CMU, 1988.

Fu, L., and T. Chen, "Sensitivity Analysis for Input Vector in Multilayer Feedforward Neural Networks," *Proceedings of IJCNN*, pp.215-218, 1993.

Gerasoulis, A., and Y. Tao, "On the Granularity and Clustering of Directed Acyclic Task Graphs," *IEEE Trans. on Parallel and Distributed Systems* (Vol. 4, No. 6), pp.686-701, June 1993.

Hampshire, J. B., and B.A. Pearlmutter, *Equivalence Proofs for MultiLayer Perceptron Classifiers and the Bayesian Discriminant Function*, Technical Report,

Department of Electrical and Computer Engineering, CMU, 1990.
Hancock, E. R., and J. Kittler, "A Bayesian Interpretation for the Hopfield Networks," *Proceedings of ICNN '93,* pp.241-346, 1993.
Hathaway, R.J., and J.C. Bezdek, "Switching Regression Models and Fuzzy Clustering, "*IEEE Trans. on Fuzzy Systems* (Vol.1, No.3), pp.195-204, August 1993.
Hathaway, R.J., and J.C. Bezdek, "Local Convergence of the Fuzzy C-means Algorithms", *Pattern Recognition* (Vol.19, No.6), pp.477-480, 1986.
Hornik, K., M. Stinchcombe, and H. White, "Multilayer Feedforward Networks are Universal Approximators," *Neural Networks*(Vol.2), pp.359-366, 1989.
Heywood, M. and P. Noakes, "Simple Addition to Back Propagation Learning for Dynamic Weight Pruning, Sparse Network Extraction and Faster Learning," *Proceedings of 1993 IJCNN,* pp.620-625, 1993.
Kang, B.H., J.H. Yoo, and S.J. Kang, "Speeding up Generalized Learning Vector Quantization," will be in the *Proceedings of ICONIP(International Conference on Neural Information Processing),* pp.450-453, Seoul, October, 1994.
Kang, B.H., D.S. Hwang, and J.H. Yoo, "Square Error Clustering Scheme and Clustering Networks," in the *Proceedings of Izuka-94 International Conference on Neural Networks, Fuzzy and Soft Computing,* Izuka, Japan, 1994.
Kavuri, S.N., and V.Venkatasubramanian, "Using Fuzzy Clustering with Ellipsoidal Units in Neural Networks for Robust Fault Classification,"*Computers Chemical Engineering* (Vol.17, No.8), pp.765-784, 1993.
Kim, J.W., *A comparative analysis of rule based, neural networkn and statistical classification for the bond rating problem,* Ph.d, Dissertation, Dept. of Information Systems, Virginia Commonwealth University Richmond, Virginia (April, 1992).
Kim, J.W., J.U. Choi, H.U. Choi, Y.Chung, and B.H. Kang, "Developing a Neural Net-Based Credit Evaluation System with Noisy Data," *Proceedings of the Third International Conference on Automation, Robotics, and Computer Vision,* Singapore, November 1994.
Kim, J.W., J.U. Choi, H.U. Choi, Y.Chung, and B.H. Kang, "A Neural Net-Based Credit Evaluation System with Noisy Data," *Proceedings of the International Conference on Neural Information Processing,* Seoul, Oct. 1994.
Kohonen, T., "The Self-Organizing Map," in Lau, C(eds), *IEEE Neural Networks,* IEEE (New York), pp.74-90, 1992.
Kong, S., and B. Kosko, "Differential Competitive Learning for Phoneme Recognition," in Kosk, B. (eds) *Neural Networks for Signal Processing,* Prentice-Hall (New Jersey), pp. 36, 1992.
Kononenko, I., "Feedforward Bayesian Neural Networks and Continuous Attributes," *Proceedings of ICNN '91,* pp.146-151, 1991.
Lam, W., and F. Bacchus, "Learning Bayesian Belief Networks: An Approach Based On the MDL Principle," *Proceedings of The Second Pacific RIM International Conference on AI,* pp.1237-1243, 1992.
Lansner, A., and O. Ekeberg, "A One-Layer Feedback Artificial Neural Network with a Basyesian Learning Rule," *International Journal of Neural Systems*(Vol.1, No.1), pp.77-87, 1989.
Lee, S., and S. Shimoji, "BAYESNET: Bayesian Classification Network Based on Biased Random Competition Using Gaussian Kernets," *Proceedings of ICNN '93,*pp.1354-1359,1993.
Leinweber, D., "Knowledge-Based Systems for Financial Applications," *IEEE EXPERT,* pp.18-31, Fall 1988.
MacKay, D.J.C., *A Practical Bayesian Framework for Back Propagation Networks,* Technical Report, Department of Computation and Neural Systems at CalTech, May 1991a.
MacKay, D.J.C., *Bayesian Interpolation,* Technical Report, Department of Computation and Neural Systems at CalTech, May 1991b.
Messier, and J. V. Hansen, "Expert Systems in Auditing : The State of the Art", *Auditing : A Journal of Practice & Theory* (Fall, 1987, Vol.7, No.1), pp.94-105.
Neal, R. M., *Bayesian Mixture Modeling by Monte Carlo Simulation,* Technical Report, CRG-TR-91-2, Department of Computer Science, University of Toronto, June 1991.
Neal, R.M., *Bayesian Training of Backpropagation Networks by the Hybrid Monte Carlo Method,* Technical Report, CRG-TR-91-1, Department of

Computer Science, University of Toronto, April, 1992.
Noel, C., "Credit Scoring Systems: A Critical Analysis", *Journal of Marketing*, (Vol. 46), Spring 1982, pp.82-91.
Odom, M.D., and R. Sharda, "A Neural Network Model for Bankruptcy Prediction," *Proceedigns of IJCNN 1990*, Vol.II, pp.II-163-168, 1990.
Pal, N.R., J.C. Bezdek, E.C, Tsao, "Generalized Clustering Networks and Kohonen's Self-Organizing Scheme," *IEEE Trans. on Neural Networks* (Vol.4, No.4), pp.549-557, July 1993.
Pao, Y., *Adaptive Pattern Recognition and Neural Networks*, Addison Wesley Publishing Company, 1989
Pfister, M. and R. Rojas, "Speeding - up Backpropagation- A Comparison of Orthogonal Technique," *Proceedings of 1993 IJCNN*, pp.517-523, 1993.
Ritter, H., T. Martinetz, and K. Schulten, *Neural Computation and Self-Organizing Maps*, Addison-Wesley (Reading, Massachusetts), 1992.
Rose, K., E. Gurewitz, and G.C. Fox, "Constrained Clustering as an Optimization Method," *IEEE Trans. on Pattern Analysis and Machine Intelligence* (Vol.15, No.8), pp.785-794, August 1993.
Schumann, M. and T. Lohrbach, "Comparing Artificial Neural Networks with Other Methods within the Field of Credit Scoring," *Proceedings of 2nd Pacific Rim International Conference on AI*, Vol.I, pp.72-78, 1992.
Simpson, P.K., "Fuzzy Min-Max Neural Networks-Part 2: Clustering," *IEEE Trans. on Fuzzy Systems* (Vol.1, No.1), pp.32-45, February 1993.
Smith, M., *Neural Networks for Statistical Modeling*, Nostrand Reinhold, New York, 1993.
Surkan, A. J., and J. C. Singleton, "Neural Networks for Bond Rating Improved by Multiple Hidden Layers," *Proceedings of IJCNN* 1990, Vol.II, pp.II-157-162, 1990.
Surkan, A. J., and X.Ying, "Bond Rating Formulas Derived Through Simplifying A Trained Neural Network," *Proceedings of IJCNN 1991*, Vol.II, pp.1566-1570, 1991.
Tam, K.Y., "Applying Conceptual Clustering to Knowledge-bases Construction, " *Decision Support Systems* (Vol.10), pp.173-198. 1993.
Tyagi, A., and M.A. Bayoumi, "Defect Clustering Viewed Through Generalized Poison Distribution," *IEEE Trans. on Semi Conductor Manufacturing* (Vol.5, No.3), pp.196-206, August 1992.
Wan, E. A., "Neural Networks Classification: A Bayesian Interpretation," *IEEE Transactions on Neural Networks* (Vol.1, No.4), pp.303-305, Dec. 1990.
Wann, M., T. Hediger, and N. N. Greenbaun, "The Influence of Training Sets on Generalization in Feed-Forward Neural Networks," *Proceedings of 1990 IJCNN*, pp.III-137-142, 1990.
Weigend, A. S., and N.A. Gershenfeld (eds.), *Time Series Prediction: Forecasting the Future and Understanding the Past Eds*, Addison Wesley, 1994.
Whitley, D., and N. Karunanithi, "Generalization in Feed Forward Neural Networks," *Proceedings of IJCNN*, July 8-12, Seattle, Washington, pp. II-77 - II-82, 1991.
Yu, Y. H., and R.F. Simmons, "Descending E psilon in Back-Propagation: A Technique for Better Generalization," *Proceedings of IJCNN 1990*, June 1990, San Diego, pp.II I-167-III-172, 1990.

Table 3.

Classification Accuracy on E company with GLVQ

*: Cluster-1(Group-1,Group-2)/Cluster-2(Group-1,Group-2)

Data Set	α_0	# of Epoch	Total Error	# of Data of Each Clusters
60/60	0.4	100	0.069523	78(29,49)/42(31,11)*
		300	0.015563	78(29,49)/42(31,11)
		500	0.007593	78(29,47)/42(31,11)
	0.8	100	0.174216	73(28,45)/47(32,15)
		300	0.040346	78(29,47)/42(31,11)
		500	0.020047	78(29,47)/42(31/11)
80/80	0.4	100	0.093169	106(45,61)/54(35,19)
		300	0.020133	106(45,61)/54(35,19)
		500	0.009760	106(45,61)/54(35,19)
	0.8	100	0.241917	106(45,61)/54(35,19)
		300	0.053071	106(45,61)/54(35,19)
		500	0.026022	106(45,61)/54(35,19)

Table 4.

Classification Accuracy on S company with GLVQ

*: Cluster-1(Group-1,Group-2)/Cluster-2(Group-1,Group-2)

Data Set	α_0	# of Epoch	Total Error	#of Data of Each Clusters
60/60	0.4	100	0.103775	58(14,44)/62(46,16)*
		300	0.031993	58(14,44)/62(46,16)
		500	0.018879	58(14,44)/62(46,16)
	0.8	100	0.279991	58(14,44)/62(46,16)
		300	0.065769	58(14,44)/62(46,16)
		500	0.038674	58(14,44)/62(46,16)
100/100	0.4	100	0.213850	92(54,38)/108(46,62)
		300	0.046501	92(54,38)/108(46,62)
		500	0.022322	92(54,38)/108(46,62)
	0.8	100	0.526733	91(53,38)/109(47,62)
		300	0.120223	92(54,38)/108(46,62)
		500	0.059159	92(54,38)/108(46,62)
200/200	0.4	100	0.353833	211(127,84)/189(73,116)
		300	0.086805	211(127,84)/189(73,116)
		500	0.046624	211(127,84)/189(73,116)
	0.8	100	0.887711	213(127,86)/187(73,114)
		300	0.209308	211(127,84)/189(73,116)
		500	0.109407	211(127,84)/189(73,116)
250/250	0.4	100	0.497821	314(123,191)/186(127,59)
		300	0.113342	313(123,190)/187(127,60)
		500	0.055464	313(123,190)/187(127,60)
	0.8	100	1.204873	317(123,194)/183(127,56)
		300	0.292172	313(123,190)/187(127,60)
		500	0.145729	313(123,190)/187(127,60)
300/300	0.4	100	0.631549	237(152,85)/363(148,215)
		300	0.137622	362(148,214)/238(152,86)
		500	0.072604	365(151,214)/235(149,86)
	0.8	100	1.330084	368(147,221)/232(153,79)
		300	0.354991	364(149,215)/236(151,85)
		500	0.180403	364(150,214)/236(150,86)

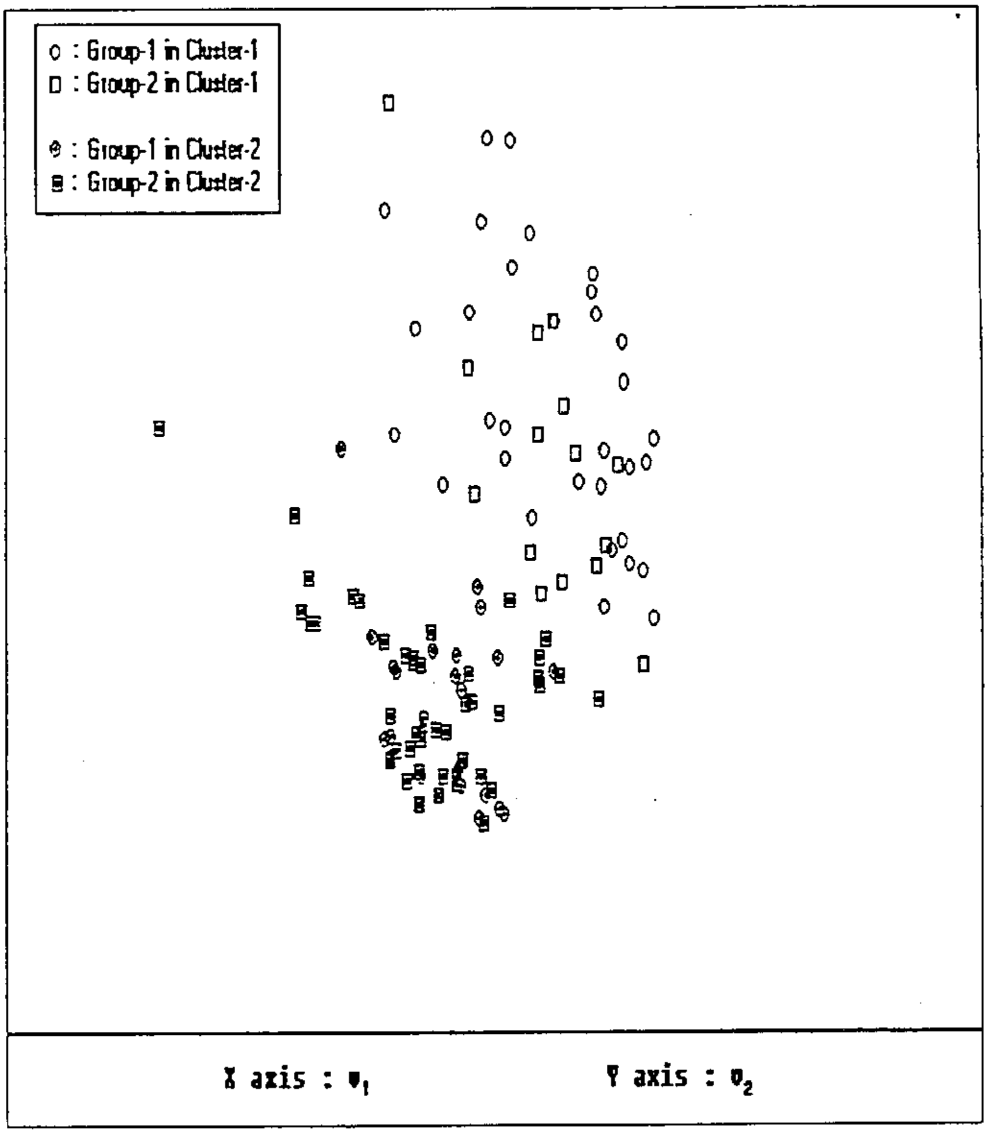

Figure 1.

Classification on E company with GLVQ

(Data Set = 60/60[Good/Bad], α_0 = 0.8, # of Epoch = 100)

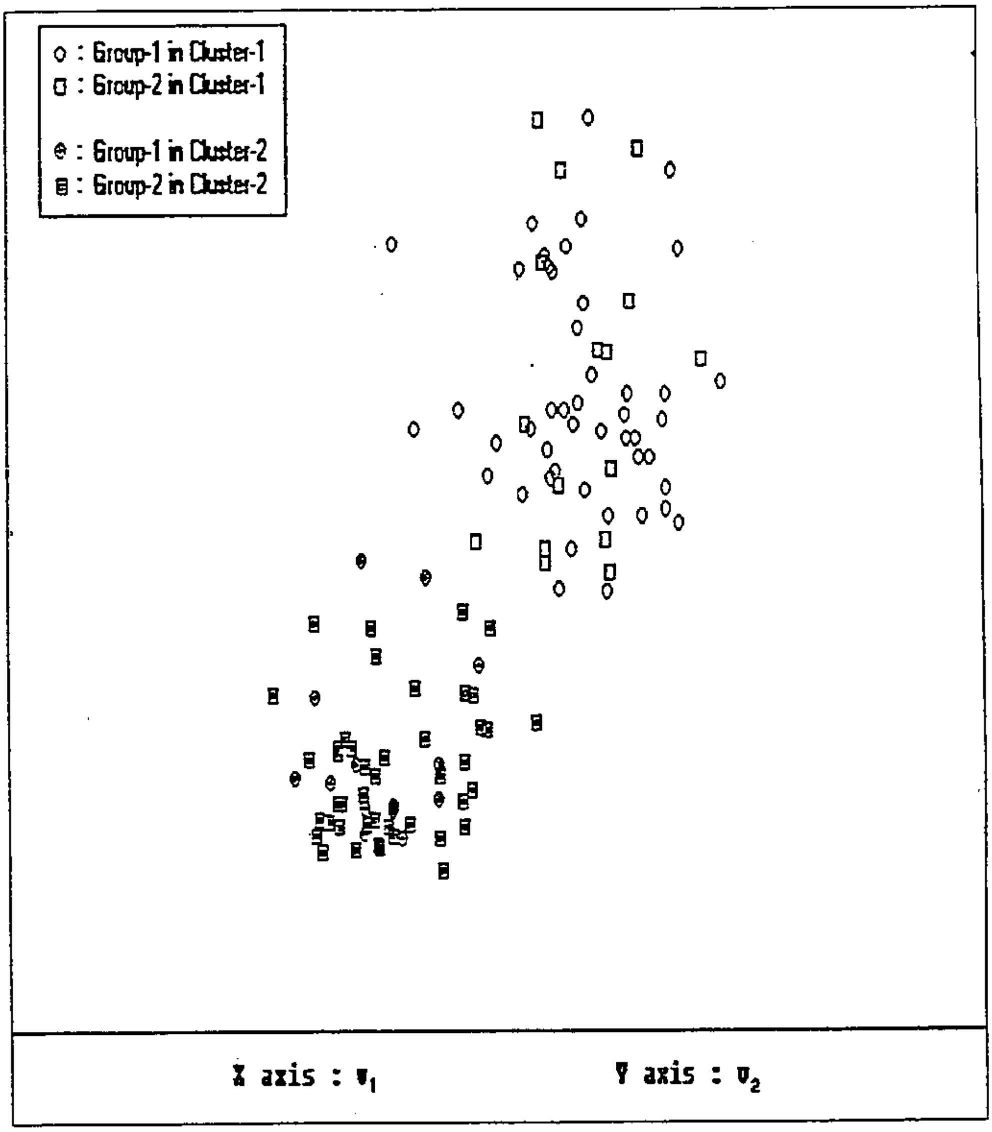

Figure 2.

Classification on S company with GLVQ

(Data Set = 60/60[Good/Bad], α_0 = 0.8, # of Epoch = 100)