Simultaneous optimization method of feature transformation and weighting for artificial neural networks using genetic algorithm : Application to Korean stock market

Kyoung-jae Kim[*] and Ingoo Han[*]

Abstract

In this paper, we propose a new hybrid model of artificial neural networks (ANNs) and genetic algorithm (GA) to optimal feature transformation and feature weighting. Previous research proposed several variants of hybrid ANNs and GA models including feature weighting, feature subset selection and network structure optimization. Among the vast majority of these studies, however, ANNs did not learn the patterns of data well, because they employed GA for simple use. In this study, we incorporate GA in a simultaneous manner to improve the learning and generalization ability of ANNs.

In this study, GA plays role to optimize feature weighting and feature transformation simultaneously. Globally optimized feature weighting overcome the well-known limitations of gradient descent algorithm and globally optimized feature transformation also reduce the dimensionality of the feature space and eliminate irrelevant factors in modeling ANNs. By this procedure, we can improve the performance and enhance the generalisability of ANNs.

Key words: simultaneous optimization method, feature transformation, feature weighting, artificial neural networks, genetic algorithm, stock market prediction

1. Introduction

It has been widely accepted that most financial variables are non-linear. Recently, artificial neural networks (ANNs) are applied to the problem of finance rapidly, such as stock market prediction, bankruptcy prediction, corporate bond rating, etc. Several studies on stock market prediction using artificial intelligence (AI) techniques were performed during past decades. Stock market prediction was typical problem of financial time-series prediction. Prior studies used various types of ANNs to predict accurate stock index and direction of change.

One of the earliest studies, Kimoto et *al.* (1990) used several learning algorithms and prediction methods for the Tokyo stock exchange prices index (TOPIX) prediction system. Their system used modular neural network to learn the relationships among various factors. Kamijo and Tanikawa (1990) used recurrent neural network and Ahmadi (1990) used backpropagation neural network with generalized delta rule to predict the stock market. Yoon and Swales (1991) also performed

[*] Graduate School of Management
Korea Advanced Institute of Science and Technology

prediction using qualitative and quantitative data. Some researchers investigated the issue of predicting stock index futures market. Trippi and DeSieno (1992) and Choi *et al.* (1995) predicted daily direction of change in S&P 500 index futures using ANNs. Duke and Long (1993) executed daily prediction of German government bond futures using feedforward backpropagation neural network.

Recent research tends to include novel factors and to hybridize several AI techniques. Hiemstra (1995) proposed fuzzy expert systems to predict stock market returns. He suggested that ANNs and fuzzy logic could capture the complexities of the functional mapping because they do not require the functional specification of the function to approximate. A more recent study of Kohara *et al.* (1997) incorporated prior knowledge to improve the performance of stock market prediction. Tsaih *et al.* (1998) integrated the rule-based technique and the ANNs to predict the direction of the S&P 500 stock index futures on a daily basis.

They, however, did not produce outstanding prediction accuracy partly because tremendous noise and non-stationary characteristics in stock market data. Training ANNs tend to be difficult with high noisy data then the network fall into a naive solution such as always predicting the most common output (Lawrence *et al.*, 1996).

For this reason, several studies proposed various kinds of hybrid models improve the learning ability of ANNs. Past research proposed several learning and search techniques to hybridize with ANNs such as MDA (Lee *et al.*, 1996), ID3 (Lee *et al.*, 1996), genetic algorithm (GA) (Harp and Samad, 1991; Schaffer *et al.*, 1992; Park *et al.*, 1994; Ornes and Sklansky, 1997; Yang *et al.*, 1998; Sexton *et al.*, 1998a). Lee *et al.* (1996) also used self-organizing feature map and Sexton *et al.* (1998b) used tabu search as a hybridizing method.

In this paper, we propose a new hybrid model of ANNs and GA for optimal feature transformation and feature weighting. Previous researches proposed several variants of ANNs and GA hybrid models such as feature weighting, feature subset selection and network structure optimization. The vast majority of these studies, however, ANNs did not learn the patterns of data well because they employed GA for simple use. GA approach, however, can potentially be used to optimize multiple factors of the learning process. In this study, we use GA to optimize multiple factors of learning process to improve the generalisability of ANNs. First, GA play a role to optimize feature weighting of ANNs. Feature weighting includes the notions of feature subset selection and the optimization of network structure. The majority of ANNs rely on a gradient descent algorithm to optimize the connection weights of network. Gradient descent algorithm, however, often did not produce generalized model because of well-known limitations. Second, this study adopts feature transformation based on GA. Because data preprocessing is an essential step for knowledge discovery and eliminate some irrelevant and redundant features, many researchers in society of data mining have a broad interest in feature transformation (Liu and Motoda, 1998). In many applications, the size of data is so large that learning of pattern may not work as well. Reducing and transforming the irrelevant or redundant features shortens the running time of a learning model and yields more generalized results (Dash and Liu, 1997). Feature transformation, in this study, is to transform continuous values into discrete ones in accordance with optimized threshold through genetic search. This approach effectively filters data, trains the classifier, and extracts the rules easier from the classifier. In addition, it reduces the dimensionality of the feature space then not only decreases the cost and times in the operation of the classifier but also enhances the generalisability of classifier.

The rest of the paper is organized into five sections. The next section reviews feature weighting and feature transformation methodologies. In the third section, we propose simultaneous optimization method for feature transformation and feature weighting in ANNs using GA and describe the benefits of proposed approach. In the fourth section, we describe the design of this research and execute experiments. In the fifth section, empirical results are summarized and discussed. In the following section, conclusions and research implications are presented with the assessment of our approach.

2. Feature weighting and feature transformation for ANNs

For a long time, there have been much research interests to predict future. Among them, several amount of research to predict future using data mining techniques including ANNs. ANNs have preeminent learning ability, however, often confront with inconsistent and unpredictable performance. As mentioned earlier, because data preprocessing is an essential step for knowledge discovery and eliminate some irrelevant and redundant features, many researchers in society of data mining have a broad interest in feature transformation and subset selection (Liu and Motoda, 1998). It is especially important to optimize network structure of ANNs. Feature subset selection and network structure optimization is partly reflected by the feature weighting in the process of modeling ANNs.

Feature weighting means, in this study, optimizing the connection weight vectors of ANNs. The vast majority of ANNs studies rely on a gradient descent algorithm to get the weight vector of the model. Sexton *et al.* (1998a) pointed out the fact that the gradient descent algorithm, however, applied to complex nonlinear optimization problems often resulted in inconsistent and unpredictable performance. Their indication stems from the fact that backpropagation is a local search algorithm and may tend to become fell into local minimum.

Several research have attempted to address this problem, Sexton *et al.* (1998a) stated as the use of the momentum, restarting training at many random points, restructuring the network architecture, and applying significant constraints to the permissible forms can fix it. They suggested that one of the more promising directions is using global search algorithms to search the weight vector of network instead of local search algorithm including backpropagation.

Some ANNs research advocated global search can improve performance. Sexton *et al.* (1998a) employed GA first to search the weight vector of ANNs. They compared backpropagation with GA and resulted each GA derived solution was superior to the corresponding backpropagation solution. Sexton *et al.* (1998b) also used tabu search to optimize the network, tabu search derived solutions were significantly superior to those of backpropagation solutions for all test data in the resulting comparison. In another paper, Sexton *et al.* (1999) again incorporated simulated annealing, one of the global search algorithms, to optimize the network. They compared with the solution derived by GA and simulated annealing and concluded solution with GA outperformed that with simulated annealing.

Although the effort of Sexton and his colleagues, Shin *et al*. (1998) concluded that ANNs are trained by gradient descent algorithm outperform GA in their application of bankruptcy prediction. In this paper, we result that GA solution cannot always guarantee the superior performance than ANNs are trained with gradient descent algorithm.

Feature transformation is the process of creating a new set of features (Liu and Motoda, 1998). It differs from feature subset selection in that the latter does not generate new features and it selects a subset of original features

(Blum and Langley, 1997; Dash and Liu, 1997). Feature transformation methods are classified as endogenous (unsupervised) versus exogenous (supervised), local versus global, parameterized versus non-parameterized, and hard versus fuzzy (Scott *et al*., 1997; Susmaga, 1997).

Endogenous (unsupervised) methods do not take into consideration of the value of the decision attribute while exogenous (supervised) do. Local methods discretize one attribute at once while the global ones discretize all attributes simultaneously. Parameterized methods specify the maximal number of intervals generated in advance while non-parameterized methods determine it automatically. Hard methods discretize the intervals at the cutting point exactly while fuzzy methods discretize it by overlapping bounds (Susmaga, 1997). The methods of endogenous feature transformation include discretization using a self-organizing map (Lawrence *et al*., 1996), percentile method (Scott *et al*., 1997; Buhlmann, 1998), clustering method (Scott *et al*., 1997; Kontkanen *et al*., 1997). Basak *et al*. (1998) also proposed neuro-fuzzy approach using feature evaluation index and Piramuthu *et al*. (1998) suggested decision-tree based approach as an endogenous method. These methods have the advantage of simplicity in transformation process. While they do not give consideration to the correlation among each independent and dependent variables. Prediction performance, however, is enhanced by the ability of discrimination from not only single variable but also the association among variables. For the reason of above limitation, endogenous methods do not provide an effective way of forming categories (Scott *et al*., 1997).

On the other hand, the methods of exogenous feature transformation include maximizing the statistical significance of Cramer's V between other dichotomized variable (Scott *et al*., 1997), entropy minimization heuristic in inductive learning and k-nearest neighbor method (Fayyad and Irani, 1993; Ting, 1997; Martens *et al*., 1998). Exogenous methods also include functional links found by genetic algorithm (GA) for ANNs and C4.5 (Haring *et al*., 1997; Vafaie and De Jong, 1998). These methods transform an independent variable to maximize its association with the values of dependent and other independent variables.

3. Feature transformation by GA for ANNs

Data analysis using statistical method or AI technique includes trend prediction and pattern classification. Trend prediction usually treats single or multiple time-series data with continuous type as input variables. It aims to capture temporal patterns between the time lag of historical data. The examples of trend prediction are the prediction of stock price, interest rate, and economic indices. They were traditionally analyzed by linear regression or time-series analysis such as autoregressive integrated moving average process (ARIMA). Pattern classification such as bond rating and credit evaluation, however, usually employs multiple cross-sectional data with discrete and continuous type as input variable. It aims to grasp the causality between the data simultaneously.

As mentioned above, it is very important to consider the temporal patterns between data on time lag when analyzing the time-series data using ANNs. A temporal pattern, however, is difficult to train because the multi-layer perceptron has the risk of learning the unnecessary random correlation and noise, because it has an outstanding ability of fitting. Weigned *et al.* (1991) used weight-elimination and Jhee and Lee (1993) used recurrent neural network to prevent the overfitting problem. In addition, time-series prediction requires a long computational time because it uses a large number of complex relationships.

Because many fund managers and investors in stock market generally accept and use criterion in Table 1 as the signal of future market trend. Therefore, they interpret technical indicator not by continuous measure but by qualitative term. Even if an attribute represents a continuous measure, the experts usually interpret the values in qualitative terms as bullish and bearish or low, medium and high. For 'stochastic %k', the value of about 75 is basically accepted by stock market analysts as strong signal, when the value exceeds about 75, the market is regarded as an overbought situation or bullish market. On the other hand, if it drop below 25, it is considered as oversold situation or the signal of bearish market. When the value of 'stochastic %k' is placed between about 25 and 75, it is regarded as the signal of neutral market (Edwards and Magee, 1997). Table 1 reviews interpretation threshold for technical indicator in stock market.

	Achelis (1995)	Gifford (1995)	Edwards and Magee (1997)	Murphy (1986)	Chang *et al.* (1996)	Choi (1995)
Stochastic %K	20/80	30/70	20~25 /75~80	30/70	25/75	20/80
Stochastic %D	20/80	30/70	20~25 /75~80	30/70	25/75	20/80
Stochastic slow %D	20/80	30/70	20~25 /75~80	30/70	25/75	20/80
Momentum	-6.5/+6.5				0	
ROC	-6.5/+6.5			100		
LW %R	20/80	20/80		20/80	10/90	20/80
AD OSC					0.5 or 0.2/0.8	
Disparity 5 days						100%
Disparity 10 days						100%
OSCP				0		0
CCI	-100/+100	-100/+100		-100/+100	0 or –100/+100	0 or –100/+100
RSI	30/70	30/70	20~30 /70~80	30/70	30/70	30/70

<Table 1> Interpretation threshold (Murphy, 1986; Achelis, 1995; Gifford, 1995; Chang *et al.*, 1996; Edwards and Magee, 1997; Choi, 1995)

The optimum interpretation threshold, however, vary depending on the security being analyzed and overall market condition (Achelis, 1995). Because each market has their specific threshold for interpreting, we do not have general guidelines for discretizing threshold. We may optimize the threshold for discretizing continuous measure into qualitative norm to capture domain specific knowledge. Althogh several research suggested various methods of transforming features, we propose discretization of continuous time-series data using GA as a method of feature transformation. This approach effectively filters data, trains the classifier. In addition, it reduces the dimensionality of the feature space then not only decreases the cost and time in the operation but also enhances the generalisability of classifier. It also reflects domain specific knowledge of each feature from optimized interpreting threshold. This method discretize search space into discrete categories. Because feature transformation using GA is classified as exogenous, global, parameterized, and hard method, it may find near-optimal threshold of discretization for maximum prediction performance.

4. Simultaneous optimization method using GA for ANNs

The overall framework of simultaneous optimization method is shown in Figure 1. The process of optimizing the feature weights and threshold for feature transformation consists of three stages.

For the first stage, we search optimal connection weights and thresholds for feature transformation. In search process of GA, the parameters for searching must be encoded on chromosomes as decision variables. GA is a search algorithm based on survival of the fittest among string structures to form a search algorithm (Goldberg, 1989). For solution of optimization problems, GA has been investigated recently and shown to be effective at exploring a complex space in an adaptive way, guided by the biological evolution mechanisms of reproduction, crossover, and mutation (Adeli and Hung, 1995).

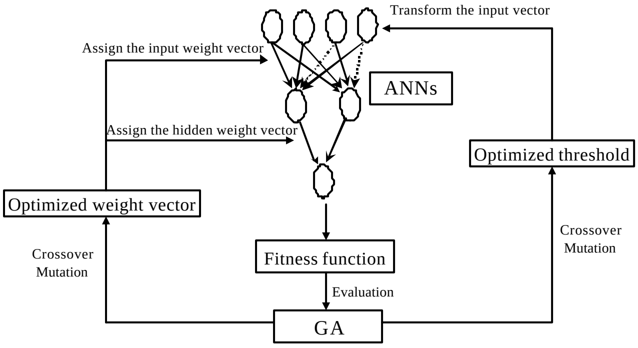

Figure 1. Simultaneous optimization method

In order to improve the performance of AI techniques, GA is usually employed (Harp and Samad, 1991; Schaffer, *et al.*, 1992; Park *et al.*, 1994). For ANNs, GA is usually employed to select neural network topology such as optimizing relevant feature subset, determining the optimal number of hidden layer and nodes, etc.

This study needs three vectors of parameters, first is connection weight vectors between input and hidden layer of network and second set is connection weight vectors between hidden and output layer. The third set is the threshold for feature transformation of each feature. Encoded connection weights and thresholds are optimized to maximize the fitness function. Fitness function is the average prediction accuracy rate of the test data set. The parameters are optimized using only the information about training data. Derived parameters from optimization process are applied to out-of-sample data. Because ANNs have eminent ability of learning the known data, the model may fall into overfitting with the training data. Therefore, the average prediction accuracy rate of test data is used as fitness function to avoid it. In this stage, GA operates the process of crossover and mutation on initial chromosomes and iterates until the stopping conditions are satisfied.

The second stage is the process of feedforward computation in ANNs. In this process, sigmoid function is used as activation function. This function is a popular aactivation function to model ANNs, because it can easily be differentiated. Linear function is used as combination function for feedforward computation with derived connection weight from first stage. In the third stage, derived connection weights and thresholds of feature transformation are applied to out-of-sample data. Table 2 summarizes the algorithm of simultaneous optimization for feature transformation and weighting.

Step 0	Initialize the populations (connection weights and thresholds for feature transformation). (Set to small random values between 0.0 and 1.0)
Step 1	While stopping condition is false, do Steps 2 – 9.
Step 2	Do Steps 3 – 8.
Step 3	Each input processing element receives input signal x_i and forwards this signal to all processing elements in hidden layer.
Step 4	Each processing element in hidden layer sums its weighted input signals and applies sigmoid activation function to compute its output signal of hidden processing element and forwards it to all processing elements in output layer.
Step 5	Each processing element in output layer sums its weighted signals from hidden layer and applies sigmoid activation function to compute its output signal of output processing element and computes the difference between output signal of output processing element and target value.
Step 6	Calculate fitness.
Step 7	Select individuals to become parents of the next generation.
Step 8	Create a second generation from the parent pool. (Perform crossover and mutation.)
Step 9	Test the stop condition.

<Table 2> The algorithm of simultaneous optimization for feature transformation and weighting

5. Research design and experiments

The research data used in this study are technical indicators and corresponding direction of change in daily Korea stock index (KOSPI). The total number of samples includes 2928 trading days from January, 1989 to December, 1998. The direction of daily change in stock index are categorized as “0” or “1”, “0” means next day’s index diminish than today's index, and "1" represent next day's index increase than today's index. We selected 12 technical indicators as feature subset by the review of domain experts and past research. Table 3 gives selected features and their formulas.

Feature	Names of feature	Formulas
X1	Stochastic %K	$\frac{C_t - L_n}{H_n - L_n} \times 100$
X2	Stochastic %D	$\frac{\sum_{i=0}^{n-1} \%K_{t-i}}{n}$
X3	Stochastic slow %D	$\frac{\sum_{i=0}^{n-1} \%D_{t-i}}{n}$
X4	Momentum	$C_t - C_{t-4}$
X5	ROC(rate of change)	$\frac{C_t}{C_{t-n}} \times 100$
X6	LW %R	$\frac{H_n - C_t}{H_n - L_n} \times 100$
X7	A/D Oscillator	$\frac{H_t - C_{t-1}}{H_t - L_t}$
X8	Disparity 5 days	$\frac{C_t}{MA_5} \times 100$
X9	Disparity 10 days	$\frac{C_t}{MA_{10}} \times 100$
X10	OSCP(price oscillator)	$\frac{MA_5 - MA_{10}}{MA_5}$
X11	CCI (commodity channel index)	$\frac{(M_t - SM_t)}{(0.015 \times D_t)}$
X12	RSI (relative strength index)	$100 - \frac{100}{1 + \frac{\sum_{i=0}^{n-1} Up_{t-i} / n}{\sum_{i=0}^{n-1} Dw_{t-i} / n}}$

<Table 3> Technical indicators (Achelis, 1995; Gifford, 1995; Chang *et al.*, 1996; Edwards and Magee, 1997; Choi, 1995)

Note) C: Closing price, L: Low price, H: High price, MA: Moving average of price, M_t : $\frac{(H_t + L_t + C_t)}{3}$,

SM_t : $\frac{\sum_{i=1}^{n} M_{t-i+1}}{n}$, D_t : $\frac{\sum_{i=1}^{n} |M_{t-i+1} - SM_t|}{n}$, Up: Upward price change, Dw: Downward price change

Table 4 presents summary statistics for each feature.

In order to compare the effectiveness of simultaneous optimization method with other competitive method, we organize three different sets of method according to feature weighting and transformation method. First method trains ANNs with gradient descent algorithm. The backpropagation algorithm and sigmoid function are used in this model. Second method trains ANNs with weights are optimized by GA instead of gradient descent algorithm. Unlike first method, GA searches among the several sets of weight vectors simultaneously. In the third model, GA simultaneously optimizes the weights of ANNs and the thresholds of feature transformation.

Name of feature	Max	Min	Mean	Standard Deviation
Stochastic %K	100.007	0.000	45.407	33.637
Stochastic %D	100.000	0.000	45.409	28.518
Stochastic slow %D	99.370	0.423	45.397	26.505
Momentum	102.900	-108.780	-0.458	21.317
ROC	119.337	81.992	99.994	3.449
LW %R	100.000	-0.107	54.593	33.637
AD OSC	3.730	-0.157	0.447	0.334
Disparity 5 days	110.003	90.077	99.974	1.866
Disparity 10 days	115.682	87.959	99.949	2.682
OSCP	5.975	-7.461	-0.052	1.330
CCI	226.273	-221.448	-5.945	80.731
RSI	100.000	0.000	47.598	29.531

<Table 4> Summary statistics

For the controlling parameters of GA search, the population size is set to 100 organisms and the crossover and mutation rates are changed to prevent falling into the local minimum. The range of crossover rate is set between 0.5 and 0.7 and mutation rate is ranged from 0.05 to 0.1 in this study. As a stopping condition, only 5000 trials are permitted. The ranges of search space of connection weights are set from –5.0 to 5.0. The range of thresholds for feature transformation is permitted between maximum and minimum value of each feature.

The data used in this study are split into three sets of data. The first set is training set. This set is used to develop the model and to determine the connection weights of networks. The test set is the second one, this set measures how well the model interpolates using the derived connection weights through the learning process of training set. The validation set is the third one, this set used to validate the generalisability of the model for unseen data. The process of extracting the test set from the training set is particularly important in the development of effective model (Klimasuaskas, 1994). In this study, the test sets are extracted by random sampling. The number of cases in each set is shown in Table 5.

Set	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	Total
Training	162	163	163	165	165	165	164	164	163	163	1637
Test	70	70	71	71	72	72	71	71	71	71	710
Validation	57	58	58	58	59	59	58	58	58	58	581

<Table 5> Number of cases in each data set

6. Experimental Results

Three sets of model are compared according to feature weighting and feature transformation method. Table 6 describes the average prediction accuracy of each model. The first set (such set called "Conventional method" in Table 6) assigns connection weights of ANNs using gradient descent method and transforms feature space by linear scaling to the range between 0.0 and 1.0. The first set is conventional method to model ANNs. The second set (such set called "Simple optimization method" in Table 6) assigns connection weights by optimization process of GA and transforms feature space using linear scaling to the range from 0.0 to 1.0. This model was proposed by the works of Sexton *et al.* (1998a) and Shin *et al.* (1998). In the third set (such set called "Simultaneous optimization method" in this study), GA simultaneously assigns connection weights and the thresholds of feature transformation.

In Table 6, simultaneous optimization method has higher prediction accuracy than the other two methods by 12% ~ 13% at out-of-sample data. The prediction accuracy of conventional method and simple optimization method is similar to each other. It is worth giving attention to the fact that there is a shade of difference of prediction accuracy between in-sample data (training sample and test sample) and out-of-sample data for simultaneous optimization method. There is, however, a wide difference of prediction accuracy between in-sample and out-of-sample data for other two methods.

Year	Conventional method		Simple optimization method		Simultaneous optimization method	
	In-sample	Out-of-sample	In-sample	Out-of-sample	In-sample	Out-of-sample
1989	59.05	48.28	57.33	49.12	56.04	63.16
1990	62.23	49.15	59.23	56.90	63.95	62.07
1991	58.97	53.45	53.42	50.00	60.26	63.79
1992	61.02	51.72	60.17	44.83	62.29	56.90
1993	54.01	44.07	54.43	44.07	59.07	62.71
1994	62.45	64.41	61.18	59.32	64.98	66.10
1995	63.83	44.83	63.83	53.45	65.96	67.24
1996	61.28	60.35	61.70	50.00	65.11	68.97
1997	46.15	50.00	50.43	50.00	62.82	63.79
1998	55.98	51.72	56.84	48.28	60.68	63.79
Total	58.50 %	51.81 %	57.86 %	50.60 %	62.12 %	63.86 %

<Table 6> Average prediction accuracy (%)

In Table 6, simultaneous optimization method has higher prediction accuracy than the other two methods by 12% ~ 13% at out-of-sample data. The prediction accuracy of conventional method and simple optimization method is similar to each other. It is worth giving attention to the fact that there is a shade of difference of prediction accuracy between in-sample data (training sample and test sample) and out-of-sample data for simultaneous optimization method. There is, however, a wide difference of prediction accuracy between in-sample and out-of-sample data for other two methods.

We use the McNemar tests to examine whether the performance of simultaneous optimization method is significantly higher than that of other two methods. The McNemar test is nonparametric test for two related samples. This test may be used with nominal data and is particularly useful with before-after measurement of the same subjects (Cooper and Emory, 1995). Table 7 shows the results of McNemar tests to compare the performance of three methods for out-of-sample data.

	Conventional method	Simple optimization method	Simultaneous optimization method
Conventional method		0.227 (0.634)	19.347* (0.000)
Simple optimization method			31.912* (0.000)

<Table 7> McNemar values (P values) for the pairwise comparison of performance among methods (* significant at the 1% level)

As shown in Table 7, the performance of simultaneous optimization method performs significantly better than that of other two methods at a 1% level. Also Table 7 shows that the other two methods, conventional method and simple optimization method, are not significantly outperforms with each other.

Table 8 shows the optimized thresholds for feature transformation. Each threshold is used as criteria for dicretizing the continuous data.

7. Conclusions and research implications

Previous study such as Sexton *et al.* (1998a, 1998b, 1999) and Shin *et al.* (1998) had tried to optimize the controlling parameters of ANNs. Their studies only focus on the optimization of connection weights for ANNs. GA approach, however, can potentially be used to optimize other specific factors of ANNs.

Name of feature	First threshold	Second threshold
Stochastic %K	32.1346	67.7126
Stochastic %D	34.7457	66.2435
Stochastic slow %D	28.8129	64.5794
Momentum	-12.4254	17.1770
ROC	98.4868	103.6160
LW %R	33.4142	81.9412
AD OSC	0.3333	0.7132
Disparity 5 days	98.9056	102.6326
Disparity 10 days	98.4584	102.2129
OSCP	-0.9347	0.7172
CCI	-38.7281	90.9651
RSI	33.3030	70.8495

<Table 8> Average thresholds for feature transformation

In this paper, we present the simultaneous optimization method for feature transformation and weighting of connections in ANNs using GA to predict the pattern of stock market trends. This method discretizes the original data according to optimal or near-optimal thresholds of feature transformation and assigns optimal or near-optimal connection weights simultaneously. We show that simultaneous optimization method effectively filters data, trains the classifier, optimizes connection weights. In addition, we conclude simultaneous optimization method reduces the dimensionality of the feature space then enhances the generalisability of classifier from the empirical results.

These results support following findings. First, the result of experiment with the simultaneous optimization method significantly outperforms that with simple optimization method in this study. It appears that simultaneous optimization method allows better to learn noisy patterns than simple optimization method. The implications of this result partly support that the more factors of the process of ANNs are optimized by GA simultaneously, the higher the prediction performance. The simultaneous optimization method can enhance the generalisability of models.

Second, although Sexton *et al.* (1998a) suggested simple optimization method for connection weights using GA outperform conventional back-propagation neural networks with gradient descent algorithm, this study does not find a evidence to support their conclusion. We show that the result of conventional method and simple optimization method does not have significant difference between each other. The reasons of these disappointing result are summarized as two factors. The one is generic limitation of GA. In other work of Sexton *et al.* (1999), performance with the connection weights of ANNs are optimized by simulated annealing, one of the global search algorithm, did not outperform that with back-propagation neural networks with gradient descent algorithm. This result is also supported by the work of Shin *et al.* (1998). They concluded the reason of disappointing results comes from the fact that GA is less competent in local search. As mentioned earlier, global search is more desirable for learning ANNs, however, some times local search is also needed. The other factor is maybe a "curse of dimensionality". GA is a global search algorithm, however, financial data including the data of stock market is too complex to search at once. Therefore, it is needed to reduce the dimensionality of data and irrelevant factors before searching. It is supported by the results of Table 6, simple optimization method and conventional learning method are not generalized well, however, simultaneous optimization method is generalized well for out-of-sample data. These results show global search algorithm may be an alternative method for the learning in ANNs. It does not generalize the fact that global search always provide the optimal connection weights.

Third, from the Table 1 and Table 8, we find optimized thresholds for feature transformation is approximate some of domain knowledge in stock market. As mentioned earlier, analysts in stock market do not understand the technical indicators as continuous form but interpret as qualitative norm such as low, medium, and high. The majority of thresholds for feature transformation in Table 8 is coincident with domain knowledge of stock market are shown in Table 1 except for some indicators including "Momentum", "CCI", and "OSCP". Therefore, feature transformation can not only reduces noise and irrelevant factors but also provides domain knowledge to learning process.

Simultaneous optimization method in this study has several implications. Features in modeling ANNs contribute the value of network outputs through not only sole but also synergistic way. The connection weight of specific connection in ANNs reflects the importance of specific connection. It is computed by taking into consideration of the value of dependent and the association of other independent features. The contribution of each feature to output value is can not be fully reflected by connection weights. Connection weights provide synergistic effect of the association among several features, however, may do not reflect the pure embedding knowledge of each feature. The pure embedding knowledge of each feature can be reflected by qualitative norm as mentioned above. It is closely akin to the process of human thinking.

This study has some limitations. First, the number of categories for feature transformation of each feature is limited to three categories. The number is varied with the nature of each feature. This study limits the number because the computational burden of unlimited categories is too heavy to be efficiently executed by personal computer. The second limitation is the objects for optimization are focused only two factors of the process of ANNs. Simultaneous optimization method in this study produces valid results, however, GA can potentially be used to optimize several factors of the process of ANNs including feature subset selection, network structure optimization, learning parameter optimization. We also believe that there is great potential for further research with simultaneous optimization method using GA for other AI techniques including case-based reasoning and decision tree.

References

Achelis, S. B., *Technical Analysis from A to Z*, Probus Publishing, 1995.

Adeli, H. and Hung, S., *Machine learning: Neural networks, genetic algorithms, and fuzzy systems*, John-Wiley & Sons, Inc, 1995.

Ahmadi, H., "Testability of the arbitrage pricing theory by neural networks", *Proceedings of the International Conference on Neural Networks*, 1990, pp. 385-393.

Basak, J., De, R. K. and Pal, S. K., "Unsupervised feature selection using a neuro-fuzzy approach", *Pattern Recognition Letters*, 1998, pp.997-1006.

Blum, A. L. and Langley, P., "Selection of relevant features and examples in machine learning", *Artificial Intelligence*, Vol. 97, No. 1-2, 1997, pp.245-271.

Buhlmann, P., "Extreme events from the return-volume process: a discretization approach for complexity reduction", *Applied Financial Economics*, Vol.8, 1998, pp.267-278.

Chang, J., Jung, Y., Yeon, K., Jun, J., Shin, D. and Kim, H., *Technical Indicators and Analysis Methods*, Seoul, Korea, Jinritamgu Publishing, 1996.

Choi, J., *Technical Indicator*, Seoul, Korea, Jinritamgu Publishing, 1995.

Choi, J. H., Lee, M. K. and Rhee, M. W., "Trading S&P 500 stock index futures using a neural network", *The Third Annual International Conference on Artificial Intelligence Applications on Wall Street*, 1995, pp. 63-72.

Cooper, D. R. and Emory, C. W., *Business research methods*, Fifth edition, Irwin, Inc., 1995, pp. 455-457.

Dash, M. and Liu, H., "Feature selection methods for classifications", *Intelligent Data Analysis: An International Journal*, Vol. 1, No. 3, 1997, http://www-east.elsevier.com/ida/browse/0103/ida00013/article.htm.

Duke, L. S., and Long, J. A., "Neural network futures trading - A feasibility study", *Adaptive Intelligent Systems*, Elsevier Science Publishers, 1993, pp. 121-132.

Edwards, R. D. and Magee, J., *Technical analysis of stock trends*, Chicago, Illinois, John Magee Inc., 1997.

Fayyad, U. M. and Irani, K. B., "Multi-interval discretization of continuous-valued attributes for classification learning", *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, 1993. pp. 1022-1027.

Gifford, E., *Investor's Guide to Technical Analysis: Predicting Price Action in the Markets*, London, Pitman Publishing, 1995.

Goldberg, D. E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, Inc., 1989.

Haring, S., Kok, J. N. and van Wezel, M. C., "Feature selection for neural networks through functional links found by evolutionary computation", *Advances in Intelligent Data Analysis*, 1997.

Harp, S. A. and Samad, T., "Optimizing neural networks with genetic algorithms", *Proceedings of the American Power Conference*, Chicago, pp.1138-1143.

Hiemstra, Y., "Modeling structured nonlinear knowledge to predict stock market returns", In Trippi, R. R. (ed.), *Chaos & Nonlinear Dynamics in the Financial Markets: Theory, Evidence and*

Applications, Irwin, 1995, pp. 163-175.

Jhee, W. C. and Lee, J. K., "Performance of Neural Networks in Managerial Forecasting", *International Journal of Intelligent Systems in Accounting, Finance and Management*, Vol. 2, pp. 55 ~ 71, 1993.

Kamijo, K. and Tanigawa, T., "Stock price pattern recognition: A recurrent neural network approach", *Proceedings of the International Joint Conference on Neural Networks,* 1990, pp. 215-221.

Kimoto, T., Asakawa, K., Yoda, M., and Takeoka, M., "Stock market prediction system with modular neural network", *Proceedings of the International Joint Conference on Neural Networks,* 1990, pp. 1-6.

Klimasauskas, C. C., "Neural network techniques", In Deboeck, G. J. (ed.), *Trading on the Edge*, New York: John Wiley, 1994, pp.3-26.

Kohara, K., Ishikawa, T., Fukuhara, Y. and Nakamura, Y., "Stock price prediction using prior knowledge and neural networks", *International Journal of Intelligent Systems in Accounting, Finance and Management,* Vol. 6, 1997, pp. 11-22.

Kontkanen, P., Myllymaki, P., Silander, T. and Tirri, H., "A Bayesian approach to discretization", *Proceedings of the European Symposium on Intelligent Techniques*, 1997, pp.265-268.

Lawrence, S., Tsoi, A. C. and Giles, C. L., "Noisy time series prediction using symbolic representation and recurrent neural network grammatical inference", *Technical Report UMIACS-TR-96-27 and CS-TR-3625*, Institute for Advanced computer Studies, University of Maryland, 1996.

Lee, K. C., Han, I. and Kwon, Y., "Hybrid neural network models for bankruptcy predictions", *Decision Support Systems*, Vol. 18, 1996, pp. 63-72.

Liu, H. and Motoda, H., "Feature transformation and subset selection", *IEEE Intelligent Systems and Their Applications*, Vol. 13, No. 2, 1998, pp. 26-28.

Martens, J., Wets, G., Vanthienen, J. and Mues, C., "An initial comparison of a fuzzy neural classifier and a decision tree based classifier", *Expert Systems with Applications*, Vol.15, 1998, pp.375-381.

Murphy, J. J., *Technical Analysis of the Futures Markets: A Comprehensive Guide to Trading Methods and Applications*, New York, New York Institute of Finance, A Prentice-Hall Company, 1986.

Piramuthu, S., Ragavan, H. and Shaw, M. J., "Using feature construction to improve the performance of neural networks", *Management Science*, Vol. 44, No. 3, 1998, pp. 416-430.

Ornes, C. and Sklanski, J., "A neural network that explains as well as predicts financial market behavior", *Proceedings of the IEEE/IAFE*, 1997, pp.43-49.

Park, D., Kandel, A. and Langholz, "Genetic-based new fuzzy reasoning models with application to fuzzy control", *IEEE Transactions on Systems, Man and Cybernetics*, Vol.24, No.1, 1994, pp.39-41.

Schaffer, J. D., Whitley, D. and Eshelman, L. J., "Combinations of genetic algorithms and neural networks: a survey of the state of the art", *Proceedings of the International Workshop on Combinations of Genetic Algorithms and Neural Networks*, Baltimore, 1992, pp.1-37.

Scott, P. D., williams, K. M. and Ho, K. M., "Forming categories in exploratory data analysis and data mining", *Advances in Intelligent Data Analysis*, 1997, Springer-Verlag, pp.235-246.

Sexton, R. S., Dorsey, R. E. and Johnson, J. D., "Toward global optimization of neural networks: A comparison of the genetic algorithm and backpropagation", *Decision Support Systems*, Vol. 22, 1998a, pp. 171-185.

Sexton, R. S., Alidaee, B., Dorsey, R. E. and Johnson, J.

D., "Global optimization for artificial neural networks: A tabu search application", *European Journal of Operational Research*, Vol. 106, 1998b, pp. 570-584.

Sexton, R. S., Dorsey, R. E. and Johnson, J. D., "Optimization of neural networks: A comparative analysis of the genetic algorithm and simulated annealing", *European Journal of Operational research*, Vol. 114, 1999, pp. 589-601.

Shin, K., Shin, T. and Han, I., "Neuro-genetic approach for bankruptcy prediction: A comparison to back-propagation algorithms", *Proceedings of the Korea Society of Management Information Systems '98 International Conference*, Seoul, Korea, 1998, pp. 585-597.

Susmaga, R., "Aanalyzing discretizations of continuous attributes given a monotonic discrimination function", *Intelligent Data Analysis: An International Journal*, Vol.1, No.3, 1997, http://www-east.elsevier.com/ida/browse/0103/ida00012/article.htm.

Ting, K.A., "Discretization in lazy learning algorithms", *Artificial Intelligence Review*, Vol.11, 1997, pp.157-174.

Trippi, R. R., and DeSieno, D., "Trading equity index futures with a neural network", *The Journal of Portfolio Management*, 1992, pp. 27-33.

Tsaih, R., Hsu, Y. and Lai, C. C., "Forecasting S&P 500 stock index futures with a hybrid AI system", *Decision Support Systems*, 1998, pp. 161-174.

Vafaie, H. and De Jong, K., "Feature space transformation using genetic algorithms", *IEEE Intelligent Systems and Their Applications*, Vol.13, No.2, 1998, pp.57-65.

Weigned, A. S., Rumelhart, D. E., and Huberman, B. A., "Generalization by weight elimination applied to currency exchange rate prediction", *Proceedings of the International Joint Conference on Neural Networks*, 1991, pp. 837-841.

Yang, J. and Honavar, V., "Feature subset selection using a genetic algorithm", *IEEE Intelligent Systems and Their Applications*, Vol.13, No.2, 1998, pp.44-49.

Yoon, Y. and Swales, G., "Predicting stock price performance: A neural network approach", *Proceedings of the 24th Annual Hawaii International Conference on System Sciences*, 1991, pp. 156-162.