

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities

**WEB OF SCIENCE™**Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us? Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com

Forecasting, Diagnosis and Decision Making with Neural Networks and Self-Organizing Maps

Kazuhiro Kohara, Katsuyoshi Aoki and Mamoru Isomae
Chiba Institute of Technology
Japan

1. Introduction

Intelligent techniques such as back-propagation neural networks (BPNN) (Rumelhart et al., 1986), self-organizing maps (SOM) (Kohonen, 1995), decision trees (Quinlan, 1993) and Bayesian networks (Jensen, 2001) have been extensively investigated, and various attempts have been made to apply them to identification, prediction and control (e.g., (Bishop, 1995); (Kil & Shin, 1996); (Pham & Liu, 1995)). This chapter describes three topics: (1) forecasting with BPNN and *selective learning*, (2) diagnosis with SOM and *ensemble learning*, and (3) decision making with SOM and Analytic Hierarchy Process (AHP) (Saaty, 1980).

The first section describes stock market prediction with BPNN and selective learning techniques for improving the ability of BPNN to predict large changes. *Selective-presentation* approach, in which the training data corresponding to large changes in the prediction-target time series are presented more often, *selective-learning-rate* approach, in which the learning rate for training data corresponding to small changes is reduced, and combining two approaches are described. The prediction of daily stock prices is used as a noisy real-world problem. The results of several experiments on stock-price prediction showed that the performances of these two approaches were similar and both better than the usual presentation approach, and combining them further improved the performance.

The second section shows an experimental study on medical diagnosis with SOM and ensemble learning. We apply SOM to medical diagnosis such as breast cancer, heart disease and hypothyroid diagnosis, comparing with decision trees and Bayesian networks. We used the UCI data sets as medical diagnosis problem. The accuracy for breast cancer and hypothyroid diagnosis was comparatively high and the accuracy for heart disease diagnosis was comparatively low. Therefore, we apply ensemble learning such as *bagging* and *boosting* of SOM to heart disease diagnosis. The accuracy with ensemble learning of SOM was much improved.

The third section describes purchase decision making with SOM and AHP. We proposed a purchase decision making method using SOM and AHP. First, we divide many products into several clusters using SOM. Secondly, we select some alternatives using the product maps. Finally, we make a final choice from the alternatives using AHP. As an example of real-world applications, we apply our method to a buying personal computer (PC) problem.

We considered one hundred and twenty kinds of notebook PCs. We evaluated our method through experiments conducted by 117 people and confirmed its effectiveness.

2. Forecasting with Neural Networks and Selective Learning

Prediction using back-propagation neural networks has been extensively investigated (e.g., (Weigend et al., 1990) ; (Vemuri & Rogers, 1994) ; (Mandic & Chambers, 2001)), and various attempts have been made to apply neural networks to financial market prediction (e.g., (Azoff, 1994) ; (Refenes & Azema-Barac, 1994) ; (White, 1988) ; (Baba & Kozaki, 1992) ; (Freisleben, 1992) ; (Tang et al., 1991) ; (Kohara, 2002)), electricity load forecasting (e.g., (Park et al., 1991)) and other areas. In the usual approach, all training data are equally presented to a neural network (i.e., presented in each cycle) and the learning rates are equal for all the training data independently of the size of the changes in the prediction-target time series. Also, network learning is usually stopped at the point of minimal mean squared error between the network's outputs and the desired outputs.

Generally, the ability to predict large changes is more important than the ability to predict small changes, as we mentioned in the previous paper (Kohara, 1995). When all training data are presented equally with an equal learning rate, the BPNN will learn the small and large changes equally well, so it cannot learn the large changes more effectively. We have investigated selective learning techniques for improving the ability of neural networks to predict large changes. We previously proposed the *selective-presentation* (Kohara, 1995) and *selective-learning-rate* (Kohara, 1996) approaches and applied them into stock market prediction. In the *selective-presentation* approach, the training data corresponding to large changes in the prediction-target time series are presented more often. In the *selective-learning-rate* approach, the learning rate for training data corresponding to small changes is reduced. The previous paper (Kohara, 1995) also investigated another stopping criterion for financial predictions. Network learning is stopped at the point having the maximum profit through experimental stock-trading.

We also previously proposed combining the *selective-presentation* and *selective-learning-rate* approaches (Kohara, 2008). By combining these two approaches, we can easily achieved fine-tuned and step-by-step selective learning of neural networks according to the degree of change. Daily stock prices were predicted as a noisy real-world problem.

2.1 Selective-Presentation and Selective-Learning-Rate Approaches

To allow neural networks to learn about large changes in prediction-target time series more effectively, we separate the training data into large-change data (L-data) and small-change data (S-data). L-data (S-data) have next-day changes that are larger (smaller) than a preset value. In the *selective-presentation* approach, the L-data are presented to neural networks more often than S-data. For example, all training data are presented every fifth learning cycle, while the L-data are presented every cycle. In the *selective-learning-rate* approach, all training data are presented in every cycle; however, the learning rate of the back-propagation training algorithm for S-data is reduced compared with that for L-data. These two approaches are outlined as follows.

Selective-Presentation Approach

1. Separate the training data into L-data and S-data.
2. Train back-propagation networks with more presentations of L-data than of S-data.

3. Stop network learning at the point satisfying a certain stopping criterion (e.g., stop at the point having the maximum profit).

Selective-Learning-Rate Approach

1. Separate the training data into L-data and S-data.

2. Train back-propagation networks with a lower learning rate for the S-data than for the L-data.

3. Stop network learning at the point satisfying a certain stopping criterion (e.g., stop at the point having the maximum profit).

We combined these two approaches to achieve fine-tuned and step-by-step learning of neural networks according to the degree of change. The outline is as follows.

Combining Selective-Presentation and Selective-Learning-Rate Approaches

1. Separate the training data into L-data and S-data.

2. Separate L-data into two subsets: L1-data and L2-data, where changes in L2-data are larger than those in L1-data.

3. Separate S-data into two subsets: S1-data and S2-data, where changes in S2-data are larger than those in S1-data.

4. Train back-propagation networks with more presentations of L1- and L2-data than of S1- and S2-data, and with a lower learning rate for L1- and S1-data than for L2 and S2-data.

5. Stop network learning at the point satisfying a certain stopping criterion (e.g., stop at the point having the maximum profit).

2.2 Evaluation through Experimental Stock-Price Prediction

We considered the following types of knowledge for predicting Tokyo stock prices. These types of knowledge involve numerical economic indicators (Kohara, 1995).

1. If interest rates decrease, stock prices tend to increase, and vice versa.

2. If the dollar-to-yen exchange rate decreases, stock prices tend to decrease, and vice versa.

3. If the price of crude oil increases, stock prices tend to decrease, and vice versa.

We used the following five indicators as inputs to the neural network.

- TOPIX: the chief Tokyo stock exchange price index
- EXCHANGE: the dollar-to-yen exchange rate (yen/dollar)
- INTEREST: an interest rate (3-month CD, new issue, offered rates) (%)
- OIL: the price of crude oil (dollars/barrel)
- NY: New York Dow-Jones average of the closing prices of 30 industrial stocks (dollars)

TOPIX was the prediction target. EXCHANGE, INTEREST and OIL were chosen based on the knowledge of numerical economic indicators. The Dow-Jones average was used because Tokyo stock market prices are often influenced by New York exchange prices. We assume that tomorrow's change in TOPIX is determined by today's changes in the five indicators according to the knowledge. Therefore, the daily changes in these five indicators (e.g. $\Delta \text{TOPIX}(t) = \text{TOPIX}(t) - \text{TOPIX}(t-1)$) were input into neural networks, and the next-day's change in TOPIX was presented to the neural network as the desired output (Fig. 1). The back-propagation algorithm was used to train the network. All the data of the daily changes were scaled to the interval [0.1, 0.9]. A 5-5-1 multi-layered neural network was used (five neurons in the input layer, five in the hidden layer, and one in the output layer).

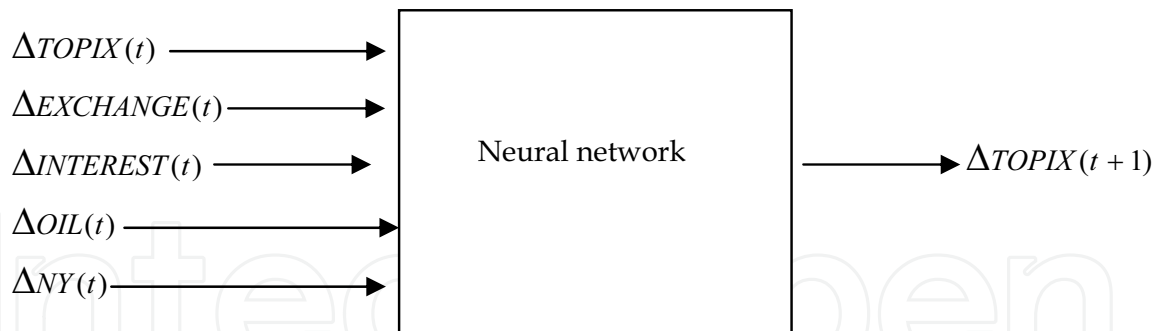


Fig. 1. Neural prediction model

2.3 Evaluation Experiments

We used data from a total of 409 days (from August 1, 1989 to March 31, 1991): 300 days for training, 30 days for validation (making decisions on stopping the network learning), and 79 days for making predictions. In Experiment 1, all training data were presented in each cycle with an equal learning rate ($\varepsilon = 0.7$). In Experiment 2, L-data were presented five times as often as S-data. Here, the large-change threshold was 14.78 points (about US\$ 1.40), which was the median of absolute value of TOPIX daily changes in the training data. In Experiment 3, the learning rate for the S-data was reduced up to 20% (i.e., $\varepsilon = 0.7$ for the L-data and $\varepsilon = 0.14$ for the S-data). In each experiment, network learning was stopped at the point having the maximum profit (the learning was stopped at the point having the maximum profit for the validation data during 8000 learning cycles). The prediction error and profit were monitored after every hundred learning cycles.

When a large change in TOPIX was predicted, we tried to calculate "Profit" as follows: when the predicted direction was the same as the actual direction, the daily change in TOPIX was earned, and when it was different, the daily change in TOPIX was lost. This calculation of profit corresponds to the following experimental TOPIX trading system. A buy (sell) order is issued when the predicted next-day's up (down) in TOPIX is larger than a preset value which corresponds to a large change. When a buy (sell) order is issued, the system buys (sells) TOPIX shares at the current price and subsequently sells (buys) them back at the next-day price. Transaction costs on the trades were ignored in calculating the profit. The more accurately a large change is predicted, the larger the profit is.

In each experiment, the momentum parameter α was 0.7. All the weights and biases in the neural network were initialized randomly between -0.3 and 0.3. In each experiment the neural network was run four times for the same training data with different initial weights and the average was taken.

The experimental results are shown in Table 1. Multiple regression analysis (MR) was also used in the experiments. The "prediction error on large-change test data" is the mean absolute value of the prediction error for the test L-data. Applying our *selective-presentation* approach (Experiment 2) reduced the prediction error for test L-data and improved profits: the prediction-error on L-data was reduced by 7% (1- (21.3/22.9)) and the network's ability to make profits through experimental TOPIX-trading was improved by 30% (550/422) compared with the results obtained with the usual presentation approach (Experiment 1). The prediction error and profits in Experiment 3 (*selective-learning-rate* approach) were comparable to those in Experiment 2 (*selective-presentation* approach). Combining *selective-presentation* with *selective-learning-rate* approaches further reduced the prediction error for

test L-data and improved profits: the prediction-error was reduced by 10% and the network’s ability to make profits was improved by 38% (Kohara, 2008).

	MR	Experiment 1	Experiment 2	Experiment 3
Presentation method	equal	equal	<i>selective</i>	equal
Learning rate		equal	equal	<i>selective</i>
Prediction error for large-change data (relative value)	24.3 (1.06)	22.9 (1)	21.3 (0.93)	21.3 (0.93)
Profit on test data (relative value)	265 (0.62)	422 (1)	550 (1.30)	563 (1.33)

Table 1. Experimental results on daily stock price prediction

2.4 Summary of Section 2

We described selective learning techniques for forecasting. In the first approach, training data corresponding to large changes in the prediction-target time series are presented more often, in the second approach, the learning rate for training data corresponding to small changes is reduced, and in the third approach, these two techniques are combined. The results of several experiments on stock-price prediction showed that the performances of these two approaches were similar and both better than the usual presentation approach, and combining them further improved the performance. Next, we will apply these techniques today’s stock market and other real-world forecasting problems. We also plan to develop a forecasting method that integrates statistical analysis with neural networks.

3. Diagnosis with Self-Organizing Maps and Ensemble Learning

We applied decision trees, Bayesian networks and SOM to medical diagnosis such as breast cancer, heart disease and hypothyroid diagnosis. The UCI data sets (Merz et al., 1997) were used as medical diagnosis problem. The accuracy of breast cancer and hypothyroid diagnosis was comparatively high and the accuracy of heart disease diagnosis was comparatively low. Therefore, we applied ensemble learning such as *bagging* (Breiman, 1994) and *boosting* (Schapire, 1990) of SOM to heart disease diagnosis. The accuracy with ensemble learning of SOM was much improved.

3.1 Medical Diagnosis with Decision Trees, Self-Organizing Maps, and Bayesian Networks

Viscovery SOMine 4.0 was used as SOM software and *See5 release 1.19* was used as decision tree software with default parameter values. We used *BayoNet 3.0.7* as Bayesian network software and constructed Bayesian networks semi-automatically. All problems were evaluated with 10-fold cross-validation. Experimental results on accuracy for medical diagnosis are shown in Table 2. Accuracy for breast cancer diagnosis (breast-w) was comparatively high and accuracy for heart disease diagnosis (heart-c) was comparatively low. In these problems, the accuracy with SOM was comparatively high and the accuracy with decision trees was comparatively low.

	Decision trees	SOM	Bayesian networks
Software	See5 release 1.19	Viscovery SOMine 4.0	BayoNet 3.0.7
Breast-w	95.3%	<u>99.1%</u>	96.0%
Heart-c	72.7%	<u>89.4%</u>	80.6%

Table 2. Experimental results on accuracy for medical diagnosis (1)

Therefore, we apply SOM and decision trees to hypothyroid data. All problems were evaluated with 10-fold cross-validation. Experimental results on accuracy are shown in Table 3. The accuracy for hypothyroid diagnosis was comparatively high. In this problem, the accuracy with decision trees was better than that with SOM.

	Decision trees	SOM
Software	See5 release 1.19	Viscovery SOMine 4.0
Hypothyroid	<u>99.5%</u>	96.8%

Table 3. Experimental results on accuracy for medical diagnosis (2)

3.2 Related Work

Tsoumakas et al. dealt with the combination of classification models that have been derived from running different (heterogeneous) learning algorithm on the same data set (Tsoumakas et al., 2004). They used WEKA implementations of the 10 base-level classification algorithms: decision tree (C4.5), support vector machine (SVM), naïve Bayes, k nearest neighbor, radial basis function and so on. Experimental results on accuracy for the same medical data as we used are shown in Table 4. Both worst and best results stated in the reference are shown.

Zhang & Su extended decision trees to represent a joint distribution and conditional independence, called conditional independence trees (CITrees) and reported that the CITree algorithm outperforms C4.5 and naïve Bayes significantly in classification accuracy (Zhang & Su, 2004). Their experimental results on accuracy for the same medical data as we used are also shown in Table 4.

Garcia-Pedrajas et al. presented a new approach to crossover operator in genetic evolution of neural networks and reported that their approach was compared to a classical crossover with an excellent performance (Garcia-Pedrajas et al., 2006). Their experimental results on accuracy for the same medical data as we used are also shown in Table 4.

Radivojac et al. investigated the problem of supervised feature selection within the filtering framework (Radivojac et al., 2004). In their approach, applicable to the two-class problems, the feature strength is inversely proportional to the p-value of the null hypothesis. Their experimental results for accuracy of heart disease data performed using naïve Bayes and SVM are shown in Table 4.

Robnik-Sikonja improved random forests (Breiman, 2001) and the experimental results on accuracy for breast cancer data are shown in Table 4 (Robnik-Sikonja, 2004). As mentioned above, the accuracy for breast cancer was high up to 99.5% (Garcia-Pedrajas et al., 2006) and the accuracy for hypothyroid was also high up to 99.7% (Tsoumakas et al., 2004). On the other hand, the accuracy for heart disease was comparatively low. The accuracy was 89.4% at most which was attained by SOM.

Reference	Tsoumakas et al., 2004	Zhang & Su, 2004	Garcia-Pedrajas et al., 2006
Classifier	Voting	DT, NB	NN, GA
Breast-w	97.0% to 98.0%	94.3% to 97.1%	93.8% to <u>99.5%</u>
Heart-c	81.8% to 88.4%	78.1% to 84.4%	86.7% to 89.1%
Hypothyroid	97.2% to <u>99.7%</u>	93.1% to 93.2%	94.4% to 94.9%

Reference	Radivojac et al., 2004	Robnik-Sikonja, 2004	This chapter
Classifier	NB, SVM	RF	SOM
Breast-w	Not available	96.6% to 96.7%	99.1%
Heart-c	83.5% to 83.9%	Not available	<u>89.4%</u>
Hypothyroid	Not available	Not available	96.8%

Voting: voting of heterogeneous classifiers including DT, NB, SVM etc.,

DT: decision trees, NB: naïve Bayes, SVM: support vector machine,

NN: neural networks, GA: genetic algorithm, RF: random forest.

Table 4. Experimental results on accuracy for medical diagnosis (3)

Quinlan investigated ensemble learning such as bagging and boosting of C4.5 (Quinlan, 1996). Experimental results on accuracy for the same medical data as we used are shown in Table 5. Bagging and boosting improved average of accuracy by about 1%. Therefore, we have decided to apply ensemble learning of SOM to heart disease data.

	C4.5	Bagged C4.5	Boosted C4.5
Breast-w	94.7%	95.8%	95.9%
Heart-c	77.0%	78.5%	78.6%
Hypothyroid	99.5%	99.6%	99.6%
Average	90.4%	91.3%	91.4%

Table 5. Experimental results on accuracy for medical diagnosis (4)

3.3 Ensemble Learning of Self-Organizing Maps

The steps of bagging and boosting for 10-fold cross-validation we used are as follows.

Bagging

Step 1: Divide all data into ten sets.

Step 2: Select one set for test data and use the remains for training set.

Step 3: Resample training subset from the training set mentioned in Step 2.

Step 4: Train SOM using the training subset resampled in Step 3.

Step 5: Repeat Step 3 to Step 4 nine times more and train the other nine SOM.

Step 6: Input test data to ten SOM and obtain results with mean summed outputs by ten SOM.

Step 7: Repeat Step 2 to Step 6 nine times more and obtain average accuracy for test data.

Boosting

Step 1: Divide all data into ten sets.

Step 2: Select one set for test data and use the remains for training set.

Step 3: Train SOM using training set, input test data to the SOM and obtain results with outputs by the SOM.

Step 4: Duplicate the misclassified data in Step 3 to the training set used in Step 3 and obtain modified training set.

Step 5: Repeat Step 3 to Step 4 nine times more to the other nine SOM.

Step 6: Input test data to ten SOM and obtain results with mean summed outputs by ten SOM.

Step 7: Repeat Step 2 to Step 6 nine times more and obtain average accuracy for test data.

We resampled 60% data of the training set in the Step 3 of bagging. When there were not misclassified data in the Step 5 of boosting, we didn't repeat Step 3 to Step 4 and the number of SOM in the Step 6 of boosting was not always ten.

Experimental results on accuracy for heart disease data with ensemble learning of SOM are shown in Table 6. Bagging and boosting improved accuracy by 1.7% and 5.0%, respectively. Especially, boosted SOM was very effective in heart disease data and its accuracy reached 94.4% which was better than that described in references mentioned above by about 5 to 10%.

	SOM	Bagged SOM	Boosted SOM
Heart-c	89.4%	91.1%	94.4%

Table 6. Experimental results on accuracy for medical diagnosis (5)

3.4 Summary of Section 3

We applied SOM to medical diagnosis such as breast cancer, heart disease and hypothyroid diagnosis, comparing with decision trees and Bayesian networks. We found that their accuracy of breast cancer and hypothyroid diagnosis was comparatively high and their accuracy of heart disease diagnosis was comparatively low. Then, we applied ensemble learning such as bagging and boosting of SOM to heart disease diagnosis. We found that their accuracy was much improved. Next, we will apply SOM and ensemble learning to the other medical diagnosis of the UCI data sets such as lung cancer and diabetes. We also plan to develop a diagnosis method that integrates decision trees and Bayesian networks with SOM.

4. Decision Making with Self-Organizing Maps and Analytic Hierarchy Process

According to Kotler (Kotler, 2002), marketing scholars have developed a *stages model* of the buying decision process. The consumer passes through five stages: problem recognition, information search, evaluation of alternatives, purchase decision, and postpurchase behavior. Five successive sets are involved in consumer decision making. The first set is the *total set* of brands available to the consumer. The individual consumer will come to know only a subset of these brands (*awareness set*). Some brands will meet initial buying criteria (*consideration set*). As the person gathers more information, only a few will remain as strong contenders (*choice set*). The brands in the choice set might all be acceptable. The person makes a final choice from this set.

Several intelligent decision support systems (DSS) have been proposed to aid in a variety of problems (see Table 7). We proposed a purchase decision support method using SOM and

AHP (Kohara & Isomae, 2006). First, we divide many products (*total set*) into several clusters using SOM. Secondly, we select some alternatives (*choice set*) using the product maps. Finally, we make a final choice from the alternatives using AHP. As an example of real-world applications, we apply our method to a buying personal computer (PC) problem. We considered one hundred and twenty kinds of notebook PCs. We evaluated our method through experiments conducted by 117 people and confirmed its effectiveness.

Application	Approach	Reference
Bankruptcy prediction	CBR and AHP	Park & Han, 2002
Weather prediction	Fuzzy reasoning	Riordan & Hansen, 2002
Economic prediction	Neural networks	Kohara, 2002
Electricity business planning	Knowledge-based system	Ha et al., 2003
Economic market model	Multi-agent system	Walle & Moldovan, 2003
Clinical decision support	AHP	Suka et al., 2003
Purchase decision support	SOM and AHP	Kohara & Isomae, 2006

Table 7. Examples of intelligent decision support systems

4.1 Self-Organizing Maps of Personal Computers

The SOM algorithm is based on unsupervised, competitive learning. It provides a topology preserving mapping from the high dimensional space to map units. Map units, or neurons, usually form a two-dimensional lattice and thus the mapping is a mapping from high dimensional space onto a plane. The property of topology preserving means that the mapping preserves the relative distance between the points. Points that are near each other in the input space are mapped to nearby map units in the SOM. The SOM can thus serve as a cluster analyzing tool of high-dimensional data. We considered one hundred and twenty kinds of notebook PCs which were sold in Japan on June 2004. We clustered these PCs using the following features: CPU speed (GHz), main memory capacity (MB), HDD storage capacity (GB), weight (kg), price (yen), battery life (hours), and so on.

We used the above features in two ways: continuous and classified data input. For classified data of CPU speed, we divide into three classes: under 1, 1 to 2, and over 2 GHz. For classified data of main memory capacity, we divide into two classes: 256 and 512 MB. For classified data of HDD storage capacity, we divide into three classes: under 40, 40 to 60, and over 60 GB. For classified data of weight, we divide into five classes: under 1, 1 to 2, 2 to 3, 3 to 4, and over 4 Kg. For classified data of price, we divide into six classes: under 100, 100 to 150, 150 to 200, 200 to 250, 250 to 300, and over 300 thousand yen. For classified data of battery life, we divide into six classes: under 1, 1 to 2, 2 to 3, 3 to 4, 4 to 5, and over 5 hours.

Discovery SOMine 4.0 was used as SOM software. Fig. 2 and Fig. 3 show self-organizing map and an example of component map of PCs with classified data inputs, respectively. Fig. 4 and Fig. 5 show self-organizing map and an example of component map of PCs with continuous data inputs, respectively. There were five clusters in Fig. 2. When inspecting component maps, the feature of each cluster is clear. For example, when inspecting "under 1 GHz (1-GHz)" component map (see Fig. 3), we understand that one of the features of Cluster 5 is that CPU speed is under 1 GHz. In Fig. 3, originally red color (here, black) neurons correspond to under 1 GHz class and originally blue color (here, dark grey) neurons correspond to the other class.

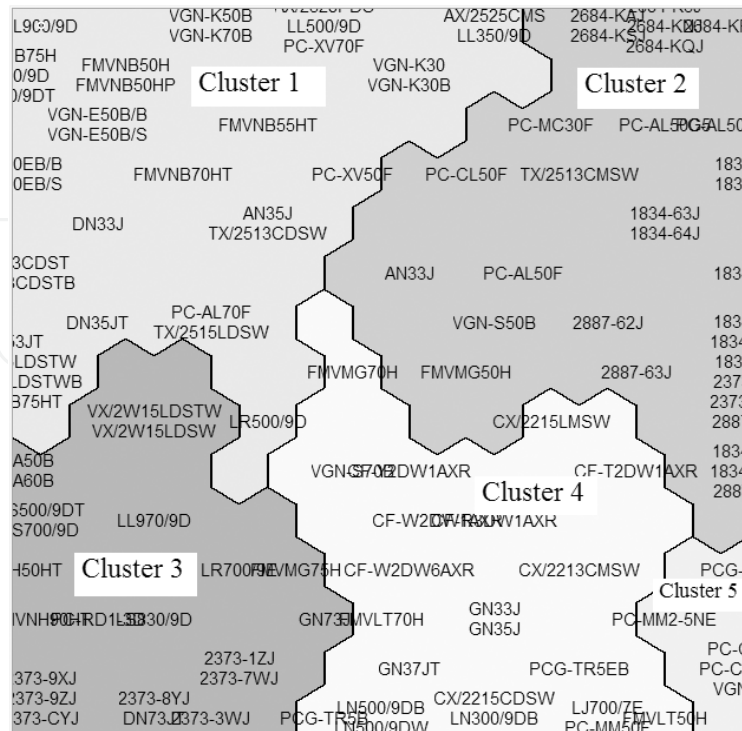


Fig. 2. Self-organizing map of PCs with classified data inputs

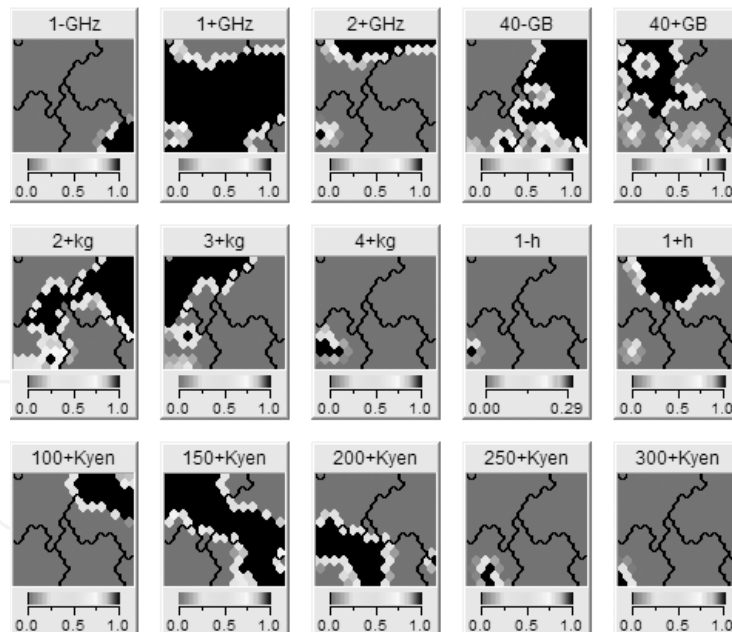


Fig. 3. Component maps of PCs with classified data inputs

There were four clusters in Fig. 4. In CPU (GHz) component map of Fig. 5, originally red (here, black) neurons correspond to 2.6 and more GHz and originally blue (here, dark grey) neurons correspond to 0.9 GHz CPU speed. Originally green and yellow (here, light grey) neurons correspond to intermediate values of CPU speed. When inspecting CPU component map of Fig. 5, the feature of each cluster is not clear. So, classified data input is better than

continuous data input for clustering PCs. From now, we used classified data input only. We inspected every component map and understand that features of Cluster 1 to Cluster 5 are as in Table 8.

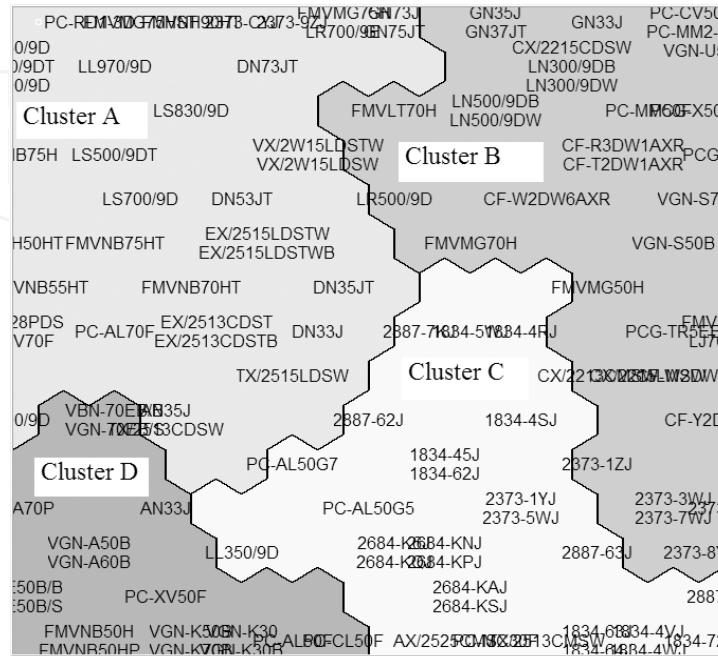


Fig. 4. Self-organizing map of PCs with continuous data inputs

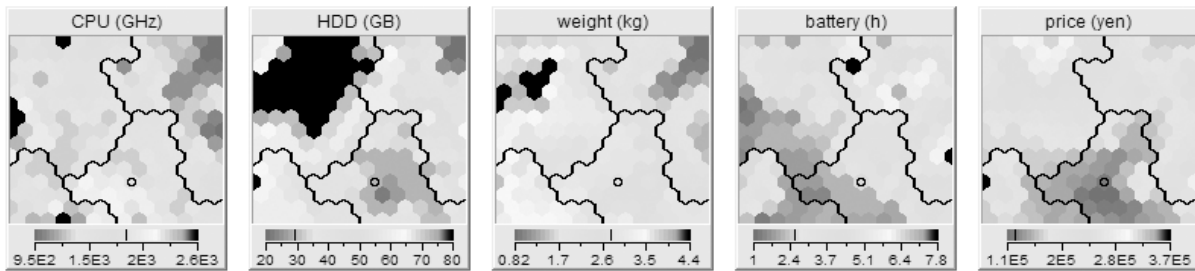


Fig. 5. Component maps of PCs with continuous data inputs

	Features	Main feature
Cluster 1	1 to 2 GHz (CPU), 40 to 60 GB (HDD), 3 to 4 Kg (weight), 150 to 200 thousand yen (price)	High performance
Cluster 2	under 40 GB (HDD), 256 MB (main memory), 100 to 150 thousand yen (price)	Low performance and low price
Cluster 3	over 60 GB (HDD), 512 MB (main memory), over 200 thousand yen (price)	Highest performance and high price
Cluster 4	1 to 2 Kg (weight), over 4 hours (battery life)	High mobility
Cluster 5	under 1 GHz (CPU), under 1 Kg (weight), 150 to 200 thousand yen (price)	Small size

Table 8. Features of clusters with classified data inputs

4.2 Decision Making with Analytic Hierarchy Process

AHP is a multi-criteria decision method that uses hierarchical structures to represent a problem and develop priorities for alternatives based on the user. Saaty (Saaty, 1980) has shown that weighting activities in multi-criteria decision-making can be effectively dealt with via hierarchical structuring and pairwise comparisons. Pairwise comparisons are based on forming judgment between two particular elements rather than attempting to prioritize an entire list of elements.

Fig. 6 shows the AHP model for a buying PC problem. Towards the goal on the first level (i.e. buying personal computer problem), four criteria in the second level and five alternatives on the third level were defined. Here, we used the following four criteria: low price, high mobility, high performance, and preference of design. Here, high mobility means small weight and long battery life. High performance means high CPU speed, large main memory capacity and large HDD storage capacity. We can select some alternatives using the PC maps in various ways. Here, we selected five alternatives using *neighborhood view* function of *Viscovery SOMine* software (this function displays all nodes that are topological similar to a reference node) and conducted AHP (see Tables 9 to 14). In this case, price is most important (see Table 9) and PC 2 is selected as final choice (see Table 14).

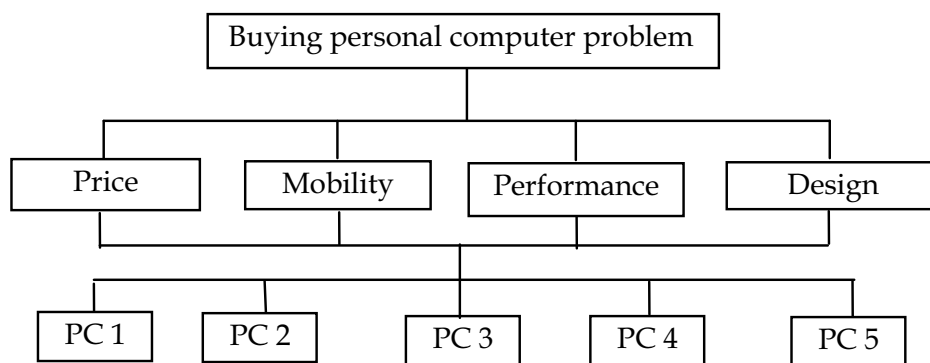


Fig. 6. AHP model for a buying PC problem

	Price	Mobility	Performance	Design	Weights
Price	1	7	3	5	0.565
Mobility	1/7	1	1/5	1/3	0.055
Performance	1/3	5	1	3	0.262
Design	1/5	3	1/3	1	0.118

Consistency index = 0.039

Table 9. Pairwise comparison matrix among four criteria

	PC 1	PC 2	PC 3	PC 4	PC 5	Weights
PC 1	1	1/5	1/4	1/4	1	0.063
PC 2	5	1	3	3	5	0.456
PC 3	4	1/3	1	1	4	0.209
PC 4	4	1/3	1	1	4	0.209
PC 5	1	1/5	1/4	1/4	1	0.063

Consistency index = 0.031

Table 10. Pairwise comparison for price

	PC 1	PC 2	PC 3	PC 4	PC 5	Weights
PC 1	1	5	5	1	1/3	0.221
PC 2	1/5	1	1	1/5	1/5	0.056
PC 3	1/5	1	1	1/5	1/5	0.056
PC 4	1	5	5	1	1/3	0.221
PC 5	3	5	5	3	1	0.448

Consistency index =0.049

Table 11. Pairwise comparison for mobility

	PC 1	PC 2	PC 3	PC 4	PC 5	Weights
PC 1	1	2	2	1/2	1/2	0.186
PC 2	1/2	1	1	1/2	1/2	0.121
PC 3	1/2	1	1	1/2	1/2	0.121
PC 4	2	2	2	1	1/2	0.246
PC 5	2	2	2	2	1	0.326

Consistency index = 0.034

Table 12. Pairwise comparison for performance

	PC 1	PC 2	PC 3	PC 4	PC 5	Weights
PC 1	1	3	3	1/3	5	0.247
PC 2	1/3	1	1	1/5	3	0.104
PC 3	1/3	1	1	1/5	3	0.104
PC 4	3	5	5	1	6	0.497
PC 5	1/5	1/3	1/3	1/6	1	0.048

Consistency index =0.040

Table 13. Pairwise comparison for design

Alternatives	Results (rank)
PC 1	0.126 (5)
PC 2	0.305 (1)
PC 3	0.165 (3)
PC 4	0.253 (2)
PC 5	0.151 (4)

Table 14. Final results of AHP for a buying PC problem

4.3 Evaluation Experiments

We evaluated our decision support method by experimental purchase decision making with SOM and AHP (Kohara & Isomae, 2006). The number of people conducting evaluation experiments is 117. They were undergraduate students of the Department of Electrical, Electronics and Computer Engineering, Chiba Institute of Technology. The above buying PC problem was used. After the experiments, we asked subjects some questions. 89.7% subjects answered that self-organizing map of PCs is *extremely useful* or *useful* for selecting some alternatives. 85.4% subjects answered that AHP is *extremely useful* or *useful* for making a final choice from the alternatives and 91.5% subjects consent with the result.

4.4 Summary of Section 4

We described a purchase decision support method using SOM and AHP. First, we divide many products into several clusters using SOM. Secondly, we select some alternatives using the product maps. Finally, we make a final choice from the alternatives using AHP. As an example of real-world applications, we apply our method to a buying personal computer problem. We evaluated our method through experiments conducted by 117 people and confirmed its effectiveness. We will apply our method into another buying product problem.

5. Conclusion

This chapter has described forecasting with BPNN and *selective learning*, diagnosis with SOM and *ensemble learning*, and decision making with SOM and AHP. First, we described stock market prediction with BPNN and selective learning techniques for improving the ability of BPNN to predict large changes. Secondly, we showed an experimental study on medical diagnosis with SOM and ensemble learning. Thirdly, we described purchase decision making with SOM and AHP. These techniques can be applied to the other forecasting, diagnosis and decision making problems.

6. References

- Azoff, E. (1994). *Neural Network Time Series Forecasting of Financial Markets*, John Wiley and Sons, West Sussex
- Baba, N. & Kozaki, M. (1992). An intelligent forecasting system of stock price using neural networks, *Proceedings of International Conference on Neural Networks*, pp. I-371-I-377, Singapore
- Bishop, C. (1995). *Neural Networks for Pattern Recognition*, Oxford University Press
- Breiman L. (1994). Bagging predictors. Technical Report 421, Statistics Department, University of California, Berkley
- Breiman, L. (2001). Random forests. *Machine Learning*, 45, (2001) 5-23
- Freisleben, B. (1992). Stock market prediction with backpropagation networks. *Lecture Notes in Computer Science*, Vol. 604, 451-460, Springer-Verlag, Heidelberg
- Garcia-Pedrajas, N.; Ortiz-Boyer, D. & Hervás-Martínez, C. (2006). An alternative approach for neural network evolution with a genetic algorithm: Crossover by combinatorial optimization. *Neural Networks*, Vol. 19, No. 4, (2006) 514-528
- Ha, L.; Forgionne, G. & Wang, F. (2003). Facilitating Electronic Business Planning with Decision Making Support Systems, *Lecture Notes in Artificial Intelligence*, Vol. 2774, 45-51, Springer-Verlag, Heidelberg
- Jensen, F. (2001). *Bayesian Networks and Decision Graphs*, Springer
- Kil, D. & Shin, F. (1996). *Pattern Recognition and Prediction with Applications to Signal Characterization*, American Institute of Physics Press
- Kohara, K. (1995). Selective Presentation Learning for Forecasting by Neural Networks, *Proceedings of International Workshop on Applications of Neural Networks to Telecommunications*, pp. 316-323, Stockholm

- Kohara, K. (1996). Selectively Intensive Learning to Improve Large-Change Prediction by Neural Networks, Proceedings of International Conference on Engineering Applications of Neural Networks, pp. 463-466, London
- Kohara, K. (2002). Neural networks for economic forecasting problems. In: Expert Systems - The Technology of Knowledge Management and Decision Making for the 21st Century-, Leondes, C. T. (Ed.) Academic Press. San Diego, CA
- Kohara, K. & Isomae, M. (2006). Purchase Decision Support with Self-Organizing Maps and Analytic Hierarchy Process, Proceedings of International Conference on Knowledge Engineering and Decision Support, pp. 151-157, Lisbon
- Kohara, K. (2008). Combining Selective-Presentation and Selective-Learning-Rate Approaches for Neural Network Forecasting of Stock Markets, Proceedings of International Workshop on Artificial Neural Networks and Intelligent Information Processing, pp. 3-9, Madeira
- Kohonen, T. (1995). Self-Organizing Maps, Springer
- Kotler, P. (2002). Marketing Management, eleventh edition, Prentice Hall
- Mandic, D. & Chambers, J. (2001). Recurrent Neural Networks for Prediction, John Wiley & Sons
- Merz, C.; Murphy, P. & Aha, D. (1997). UCI repository of machine learning databases. Dept of ICS, University of California, Irvine
- Park, C-S. & Han, I. (2002). A CBR with feature weights derived by analytic hierarchy process for bankruptcy prediction. Expert Systems with Applications, Vol. 14, No. 3, (2002) 255-264
- Park, D.; El-Sharkawi, M.; Marks II, R.; Atlas, L. & Damborg, M. (1991). Electric load forecasting using an artificial neural network. IEEE Transactions on Power Systems, Vol. 6, No. 2, (1991) 442-449
- Pham, D. & Liu, X. (1995). Neural Networks for Identification, Prediction and Control, Springer
- Quinlan, J. (1993). C4.5: Programs for Machine Learning, Morgan Kaufmann
- Quinlan, J. (1996). Bagging, boosting and C4.5, Proceedings of the 13th National Conference on Artificial Intelligence and the 8th Innovative Applications of Artificial Intelligence Conference, pp. 725-730
- Refenes, A. & Azema-Barac, M. (1994). Neural network applications in financial asset management. Neural Computing & Applications, Vol. 2, No. 1, (1994) 13-39
- Riordan, D. & Hansen, B. K. (2002). A fuzzy case-based system for weather prediction. Engineering Intelligent Systems, Vol. 3, (2002) 139-145
- Rumelhart, D.; Hinton, G. & Williams, R. (1986). Learning internal representations by error propagation. In: Parallel Distributed Processing 1, Rumelhart, D.; McClelland, J. & the PDP Research Group (Ed.), MIT Press, Cambridge, MA
- Saaty, T. (1980). The Analytic Hierarchy Process, McGraw-Hill
- Schapire, R. (1990). The strength of weak learnability. Machine Learning, 5, (1990) 197-227
- Suka, M.; Ichimura, T. & Yoshida, K. (2003). Clinical Decision Support System Applied the Analytic Hierarchy Process. Lecture Notes in Artificial Intelligence, Vol.2774, 417-423, Springer-Verlag, Heidelberg
- Tang, Z., Almeida, C. & Fishwick, P. (1991). Time series forecasting using neural networks vs. Box-Jenkins methodology. Simulation, Vol. 57, No. 5, (1991) 303-31

- Tsoumakas, G.; Katakis, I. & Vlahavas, I. (2004). Effective Voting of Heterogeneous Classifiers. *Lecture Notes in Artificial Intelligence*, Vol. 3201, 465-476, Springer-Verlag, Heidelberg
- Radivojac, P.; Obradovic, Z.; Dunker, A. & Vucetic, S. (2004). Feature Selection Filters Based on the Permutation Test. *Lecture Notes in Artificial Intelligence*, Vol. 3201, 334-346, Springer-Verlag, Heidelberg
- Robnik-Sikonja, M. (2004). Improving Random Forests. *Lecture Notes in Artificial Intelligence*, Vol. 3201, 359-370, Springer-Verlag, Heidelberg
- Vemuri, V. & Rogers, R. (Ed.) (1994). *Artificial Neural Networks: Forecasting Time Series*, IEEE Press, Los Alamitos, CA
- Walle, B. V. & Moldovan, M. (2003). An Information Market for Multi-agent Decision Making: Observation from a Human Experiment. *Lecture Notes in Artificial Intelligence*, Vol. 2774, 66-72, Springer-Verlag, Heidelberg
- Weigend, A., Huberman, B. & Rumelhart, D. (1990). Predicting the future: a connectionist approach. *International Journal of Neural Systems*, Vol. 1, No. 3, (1990) 193-209
- White, H. (1988). Economic prediction using neural networks: the case of IBM daily stock return, *Proceedings of International Conference on Neural Networks*, pp. II-451-II-458, San Diego, CA
- Zhang, H. & Su, J. (2004). Conditional Independence Trees. In: Boulicaut J-F, Esposito F, Giannotti F, Pedreschi D (eds). *Lecture Notes in Artificial Intelligence*, Vol. 3201, 513-524, Springer-Verlag, Heidelberg

IntechOpen



Automation Control - Theory and Practice

Edited by A D Rodi

ISBN 978-953-307-039-1

Hard cover, 350 pages

Publisher InTech

Published online 01, December, 2009

Published in print edition December, 2009

The present edited book is a collection of 18 chapters written by internationally recognized experts and well-known professionals of the field. Chapters contribute to diverse facets of automation and control. The volume is organized in four parts according to the main subjects, regarding the recent advances in this field of engineering. The first thematic part of the book is devoted to automation. This includes solving of assembly line balancing problem and design of software architecture for cognitive assembling in production systems. The second part of the book concerns different aspects of modelling and control. This includes a study on modelling pollutant emission of diesel engine, development of a PLC program obtained from DEVS model, control networks for digital home, automatic control of temperature and flow in heat exchanger, and non-linear analysis and design of phase locked loops. The third part addresses issues of parameter estimation and filter design, including methods for parameters estimation, control and design of the wave digital filters. The fourth part presents new results in the intelligent control. This includes building a neural PDF strategy for hydroelectric saturation simulator, intelligent network system for process control, neural generalized predictive control for industrial processes, intelligent system for forecasting, diagnosis and decision making based on neural networks and self-organizing maps, development of a smart semantic middleware for the Internet, development of appropriate AI methods in fault-tolerant control, building expert system in rotary railcar dumpers, expert system for plant asset management, and building of a image retrieval system in heterogeneous database. The content of this thematic book admirably reflects the complementary aspects of theory and practice which have taken place in the last years. Certainly, the content of this book will serve as a valuable overview of theoretical and practical methods in control and automation to those who deal with engineering and research in this field of activities.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Kazuhiro Kohara, Katsuyoshi Aoki and Mamoru Isomae (2009). Forecasting, Diagnosis and Decision Making with Neural Networks and Self-Organizing Maps, Automation Control - Theory and Practice, A D Rodi (Ed.), ISBN: 978-953-307-039-1, InTech, Available from: <http://www.intechopen.com/books/automation-control-theory-and-practice/forecasting-diagnosis-and-decision-making-with-neural-networks-and-self-organizing-maps>

INTECH
open science | open minds

InTech Europe

InTech China

www.intechopen.com

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen

IntechOpen

© 2009 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen