Available online at www.sciencedirect.com
SCIENCE DIRECT®

ELSEVIER

Neurocomputing 61 (2004) 361–379

NEUROCOMPUTING

www.elsevier.com/locate/neucom

Prediction of colon cancer using an evolutionary neural network

Kyung-Joong Kim*, Sung-Bae Cho

Department of Computer Science, Yonsei University, 134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, South Korea

Received 24 March 2003; received in revised form 8 November 2003; accepted 19 November 2003

Abstract

Colon cancer is second only to lung cancer as a cause of cancer-related mortality in Western countries. Colon cancer is a genetic disease, propagated by the acquisition of somatic alterations that influence gene expression. DNA microarray technology provides a format for the simultaneous measurement of the expression level of thousands of genes in a single hybridization assay. The most exciting result of microarray technology has been the demonstration that patterns of gene expression can distinguish between tumors of different anatomical origin. Standard statistical methodologies in classification and prediction do not work well or even at all when N (a number of samples) $< p$ (genes). Modification of conventional statistical methodologies or devise of new methodologies is needed for the analysis of colon cancer. Recently, designing artificial neural networks by evolutionary algorithms has emerged as a preferred alternative to the common practice of selecting the apparent best network. In this paper, we propose an evolutionary neural network that classifies gene expression profiles into normal or colon cancer cell. Experimental results on colon microarray data show that the proposed method is superior to other classifiers.

Keywords: DNA Microarray; Evolutionary neural network; Colon cancer; Feature selection; Information gain

1. Introduction

Recently, the techniques based on oligonucleotide or cDNA arrays allow the expression level of thousands of genes to be monitored in parallel. Critically important

* Corresponding author. Tel.: +82221234803; fax: +8223652579.

0925-2312/$ - see front matter © 2003 Published by Elsevier B.V.
doi:10.1016/j.neucom.2003.11.008

thing for cancer diagnosis and treatment is precise prediction of tumors. One of the remarkable advances for molecular biology and for cancer research is DNA microarray technology. DNA microarray datasets have a high dimensionality corresponding to the large number of genes monitored and there are often comparatively few samples. In this paper, we address the problem of prediction of cancer using a small subset of genes from broad patterns of gene expression data.

In cancer research, microarray technology allows the better understanding of the regulation of activity of cells and tumors in various states [32]. Prediction, classification, and clustering techniques are used for analysis and interpretation of the microarray data. Colon cancer is the second most common cause of cancer mortality in Western countries [7]. Gene expression in 40 tumor and 22 normal colon tissue samples was analyzed with an Affymetrix oligonucleotide array complementary to more than 6500 human genes. We chose to work only with the 2000 genes of the greatest minimal expression over the samples [1].

Evolutionary artificial neural networks (EANNs) combine the learning of neural networks and evolution of evolutionary algorithms [8]. A lot of works have been made on EANNs. For the game of checkers, the evolutionary algorithm can discover a neural network that can be used to play at a near-expert level without injecting expert knowledge about how to play the game [12]. Evolutionary algorithm can be used for various tasks, such as connection weight training, architecture design, learning rule adaptation, input feature selection, connection weight initialization and rule extraction from ANNs [38]. We propose an evolutionary neural network for classifying (predicting) human tumor samples based on microarray gene expressions. This procedure involves the dimension reduction with information gain and the classification with EANN. The proposed method is applied to colon cancer microarray data sets containing various human tumor samples. We have compared the evolutionary neural network to the well-known classification methods.

The rest of the paper is organized as follows. In Section 2, we describe the microarray technology and related works on the prediction of cancer, which include oligonucleotide microarray technology, the relevant works with evolutionary neural networks and the results on colon cancer data set of previous studies. In Section 3, we present the evolutionary neural network in details. In Section 4 we examine the performance of the proposed method.

2. Bioinformatics with DNA microarray

Uncovering broad patterns of genetic activity, providing new understanding of gene functions and generating unexpected insight into biological mechanism are the impact of microarray-based studies [19]. With the development and application of DNA microarrays, the expression of almost all human genes can now be systematically examined in human malignancies [18]. DNA sequences are initially transcribed into mRNA sequences. These mRNA sequences are translated into the amino acid sequences of the proteins that perform various functions. Measuring mRNA levels can provide a detailed molecular view of the genes. Measuring gene expression levels under different

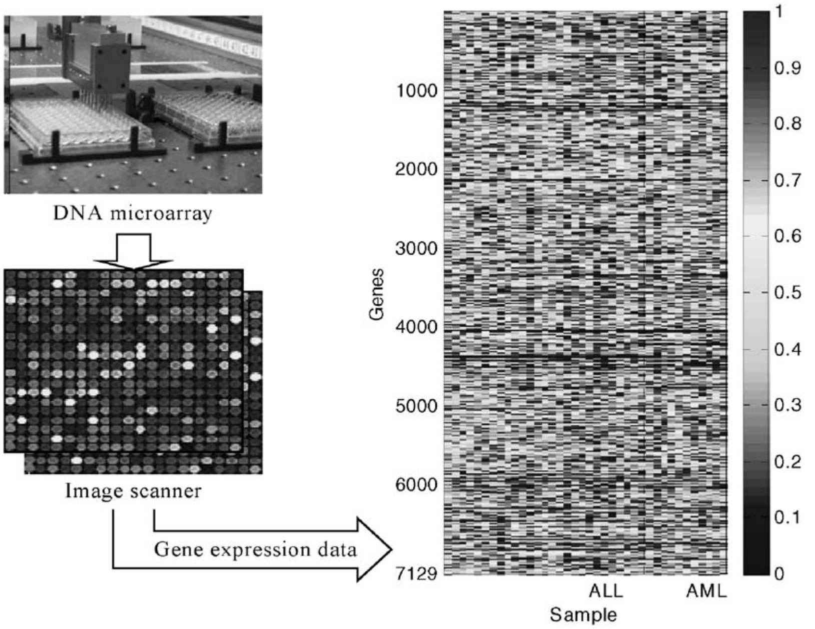


Fig. 1. General process of acquiring the gene expression data from DNA microarray. (this is an example of Leukemia cancer and there are two types of cancers including ALL and AML. A sample comes from patient.)

conditions is important for expanding our knowledge of gene function. Gene expression data can help in better understanding of cancer.

2.1. Oligonucleotide DNA microarray

A main goal of the analysis of gene expression data is the identification of sets of genes that can serve as classification. Understanding cellular responses to drug treatment is another important goal of gene expression profiling. The complexity of microarray data calls for data analysis tools that will effectively aid in biological data mining. DNA microarrays are composed of thousands of individual DNA sequences printed in a high-density array on a glass microscope slide using a robotic arrayer as shown in Fig. 1. After hybridization of two samples, the slides are imaged using scanner that makes fluorescene measurements for each dye.

In this study, Alon’s colon cancer data that are monitored using Affymetrix oligonucleotide array are used [1]. High-density oligonucleotide chip arrays are made using spatially patterned, light-directed combinatorial chemical synthesis, and contain up to hundreds of thousands of different oligonucleotides on a small glass surface [22]. As the chemical cycle is repeated, each spot on the array contains a short synthetic oligonucleotide, typically 20–25 bases. The oligonucleotides are designed based on the knowledge of the DNA target sequences, to ensure high-affinity and specificity

Table 1
Relevant works on colon cancer classification

Authors	Method		Accuracy (%)
	Feature	Classifier	
Furey et al. [13]	Signal to noise ratio	SVM	90.3
Li et al. [26]	Genetic algorithm	KNN	94.1
Ben-Dor et al. [5]	All genes, TNoM score	Nearest neighbor	80.6
		SVM with quadratic kernel	74.2
		AdaBoost	72.6
Nguyen et al. [30]	Principal component analysis	Logistic discriminant	87.1
		Quadratic discriminant	87.1
	Partial least square	Logistic discriminant	93.5
		Quadratic discriminant	91.9

of each oligonucleotide to a particular gene. This allows cross-hybridization with the other similar sequenced gene and local background to be estimated and subtracted. Oligonucleotide DNA microarray might eventually eliminate the use of cDNA arrays [4].

2.2. Related works

Derisi et al. [10] published that the expression patterns of many previously uncharacterized genes provided clues to their possible functions [10]. Eisen et al. [11] presented that clustering gene expression data grouped together efficiently genes of known similar function [11]. Shamir [34] described some of the main algorithmic approaches to clustering gene expression data [34]. Getz et al. [14] presented two-way clustering approach to gene microarray data analysis. There are many researchers to attempt to predict colon cancer using various machine learning methods and they show that prediction rate of colon cancer can be approximately 80–90% (Table 1).

Sarkar et al. [33] presented a novel and simple method that exhaustively scanned microarray data for unambiguous gene expression patterns [33]. Tclass is a corresponding program of a method that incorporates feature selection into Fisher's linear discriminant analysis for gene expression based on tumor classification [36]. Li et al. investigated two Bayesian classification algorithms incorporating feature selection and these algorithms were applied to the classification of gene expression data derived from cDNA microarrays [25]. Li et al. studied to decide which and how many genes should be selected [24]. Guyon et al. proposed a new method of gene selection using support vector machine based on recursive feature elimination (RFE) [17]. Xiong et al. reported that using two or three genes, one could achieve more than 90% accuracy of classification in colon cancer, breast cancer, and leukemia [37].

There are some related works on EANNs that combine the advantages of the global search performed by evolutionary algorithms and local search of the learning algorithms (like BP) of ANN. Yao [39] proposed EANNs approach, EPNet based on Fogel's evolutionary programming (EP) as evolutionary algorithm. EPNet emphasizes the evolution of ANN behaviors by EP and uses a number of techniques, such as partial training after each architectural mutation and node splitting, to maintain the behavioral link between parent and its offspring effectively. EPNet also encourages parsimony of evolved ANNs by attempting different mutations sequentially. That is, node or connection deletion is always attempted before addition. EPNet has shown good performance in error rate and size of ANN.

Cho proposed a new approach of constructing multiple neural networks that used genetic algorithms with speciation to generate a population of accurate and diverse ANNs. Speciation in genetic algorithm creates different species, each embodying a sub-solution, which means to create diverse solutions not the best one [2]. Experiments with the breast cancer data from UCI benchmark datasets show that the method can produce more speciated ANNs and improve the performance by combining only representative individuals [3]. Several combination methods are applied to combine speciated neural networks [23].

3. Evolutionary neural network for cancer classification

A traditional artificial neural network based on backpropagation algorithm has some limitations. At first, the architecture of the neural network is fixed and a designer needs much knowledge to determine it. Also, error function of the learning algorithm must have a derivative. Finally, it frequently gets stuck in local optima because it is based on gradient-based search without stochastic property. Evolutionary algorithm is a kind of search method based on biological facts and uses a population of multiple individuals. The combination of evolutionary algorithm and neural network can overcome these shortcomings.

Design of a near optimal ANN architecture can be formulated as a search problem in the architecture space where each point represents architecture. One major issue in evolving pure architectures is to decide how much information about architecture should be encoded into a chromosome (genotype). There are two representative encoding schemes for neural network including direct and indirect methods. In indirect encoding, rules for generating neural network structure are represented as a chromosome for the evolution [21]. If the phenotype has many overlapped components, indirect encoding is more useful than direct one because it can reduce the length of a chromosome by simple rule representation. In our work, recurrent link is not allowed and only feed-forward link is acceptable. Usually, recurrent link is used for memorizing information but in our problem, it is not useful to adopt the link.

3.1. Feature selection

There are two approaches to reduce the dimensionality of data. In filtering approach, there is no concern about which classifier is used and only characteristics of the features are measured for selection. The method is very fast and easily implemented. Meanwhile, wrapper approach is a method that uses a specific classifier for the selection procedure and the performance of the classifier-feature combination is measured for selection. In this paper, we adopt the filtering approach because it is computationally inexpensive. Usually, evolutionary computation is computationally expensive and wrapper approach is not appropriate. Details of comparison between two approaches can be found in [28].

The number of genes is too large to manipulate in learning algorithm and all features are not useful for classification. Only relevant features are useful for classification to produce better performance. Feature ranking method is used to classify genes. Information gain is representative feature ranking and selection method used in C4.5 [31] that utilizes information gain to find the most important feature for each time. Definition of information gain is restricted to genes that take on a discrete set of values. This restriction can easily be removed by dynamically defining new discrete valued genes that partition the continuous gene value into a discrete set of intervals (threshold c is used for the separation). How to select the optimal threshold c is described in [27]. In the formula below, k is the total number of classes; n is the total number of expression values; n_l is the number of values in the left partition; and n_r is the number of values in the right partition; l_i is the number of values that belong to class i in the left partition; r_i is the number of values that belong to class i in the right partition. Information gain of a gene is defined as follows:

$$\mathrm{IG} = \sum_{i=1}^{k} \left(\frac{l_i}{n} \log \frac{l_i}{n_l} + \frac{r_i}{n} \log \frac{r_i}{n_r} \right) - \sum_{i=1}^{k} \left(\frac{l_i + r_i}{n} \right) \log \left(\frac{l_i + r_i}{n} \right).$$

3.2. EANN

The simultaneous evolution of both architectures and weights can be summarized as follows: (1) Evaluate each individual based on its error and/or other performance criteria such as its complexity. (2) Select individuals for reproduction and genetic operation. (3) Apply genetic operators, such as crossover and mutation, to the ANN's architectures and weights, and obtain the next generation. Fig. 2 shows the overview of evolving neural network. Each ANN is generated with random initial weights and full connection. Then, each ANN is trained partially with training data to help the evolution search the optimal architecture of ANN and is tested with validation data to compute the fitness. The fitness of ANN is recognition rate of validation data. Once the fitness is calculated, selection is conducted that chooses the best 50% individuals to apply genetic operators. The genetic operators, crossover and mutation, are applied to those selected individuals. Then a population of the next generation is created. The process is repeated until stop criterion is satisfied. The ANNs in the last generation are trained fully.

Feature selection is used to reduce the dimensionality for EANN because one feature is corresponding to one input node and if the number of features is very large, the size of network is required to be large. Large network size is not useful for the generalization and the dimensionality reduction is needed for the EANN procedure. Data separation procedure divides the data into three distinct sample sets such as training, validation, and test sets. Training data are used for partial training and full training. Validation data are used for fitness calculation and full training. The fitness of each

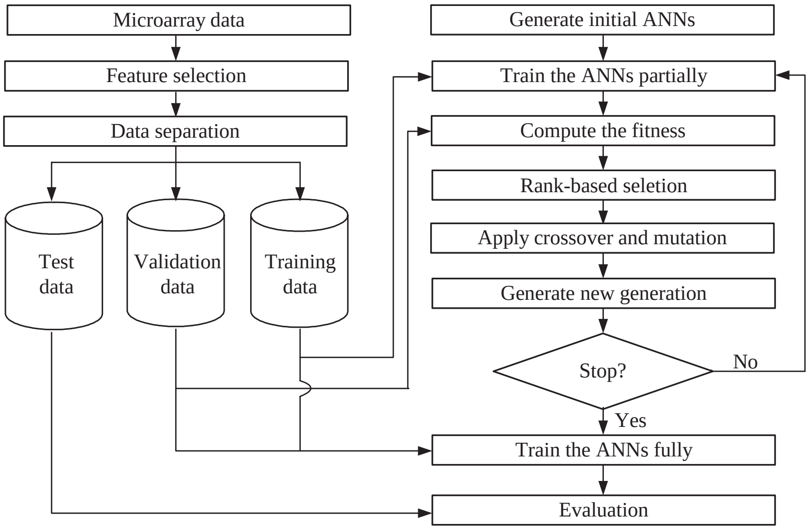


Fig. 2. The procedure for evolving neural network.

individual in EANN is solely determined by the inverse of an error value. The selection mechanism used in EANN is rank based. M is the population size. Let M sorted individuals be numbered as $0, 1, \ldots, M-1$, with the zeroth being the fittest. Then the $(M-j)$th individual is selected with probability

$$p(M-j) = \frac{j}{\sum_{k=1}^{M} k}.$$

Each sample (gene expression data for one person) is used to train or validate each individual (EANN) of the population (a set of individuals). Population is a collection of individuals and the size is fixed at the initial stage. Each individual represents one evolutionary neural network. In Fig. 2, iteration is repeated until stop criterion is satisfied. It stops when an individual shows better performance than pre-defined accuracy (100%) or iteration number exceeds pre-defined maximum number of generations.

3.2.1. Representation

To evolve an ANN, it needs to be expressed in proper form. There are some methods to encode an ANN like binary representation, tree, linked list, and matrix. We have used a matrix to encode an ANN since it is straightforward to implement and easy to apply genetic operators [35]. When N is the total number of nodes in an ANN including input, hidden, and output nodes, the matrix is $N \times N$, and its entries consist of connection links and corresponding weights. In the matrix, upper right triangle (see Fig. 3) has connection link information that is 1 when there is a connection link and 0 when there is no connection link. Lower left triangle describes the weight values corresponding to the connection link information. There will be no connections among input nodes. Architectural crossover and mutation can be implemented easily under

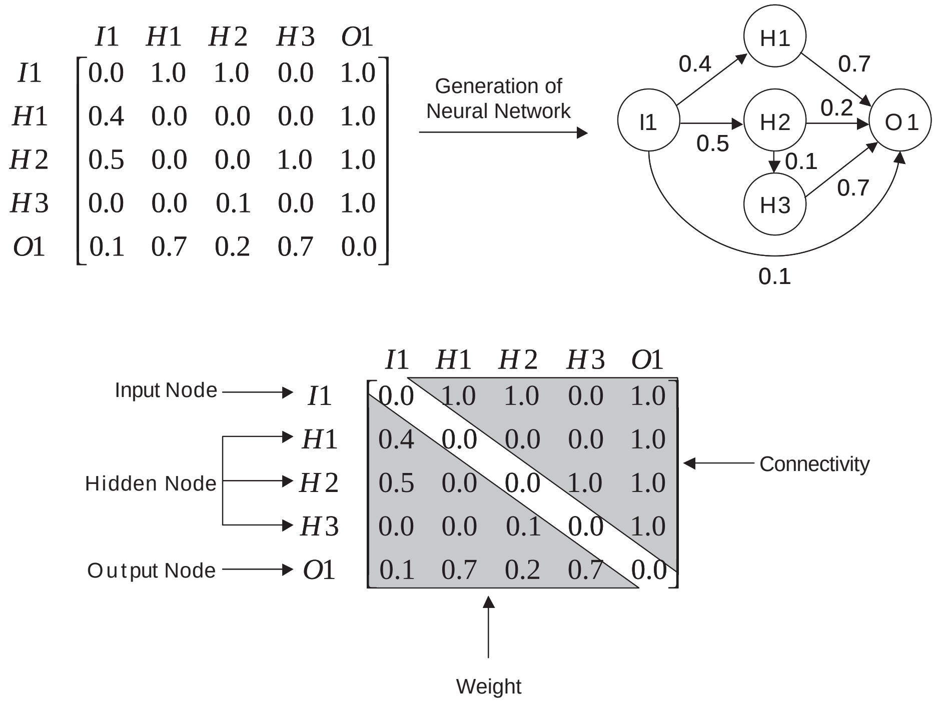


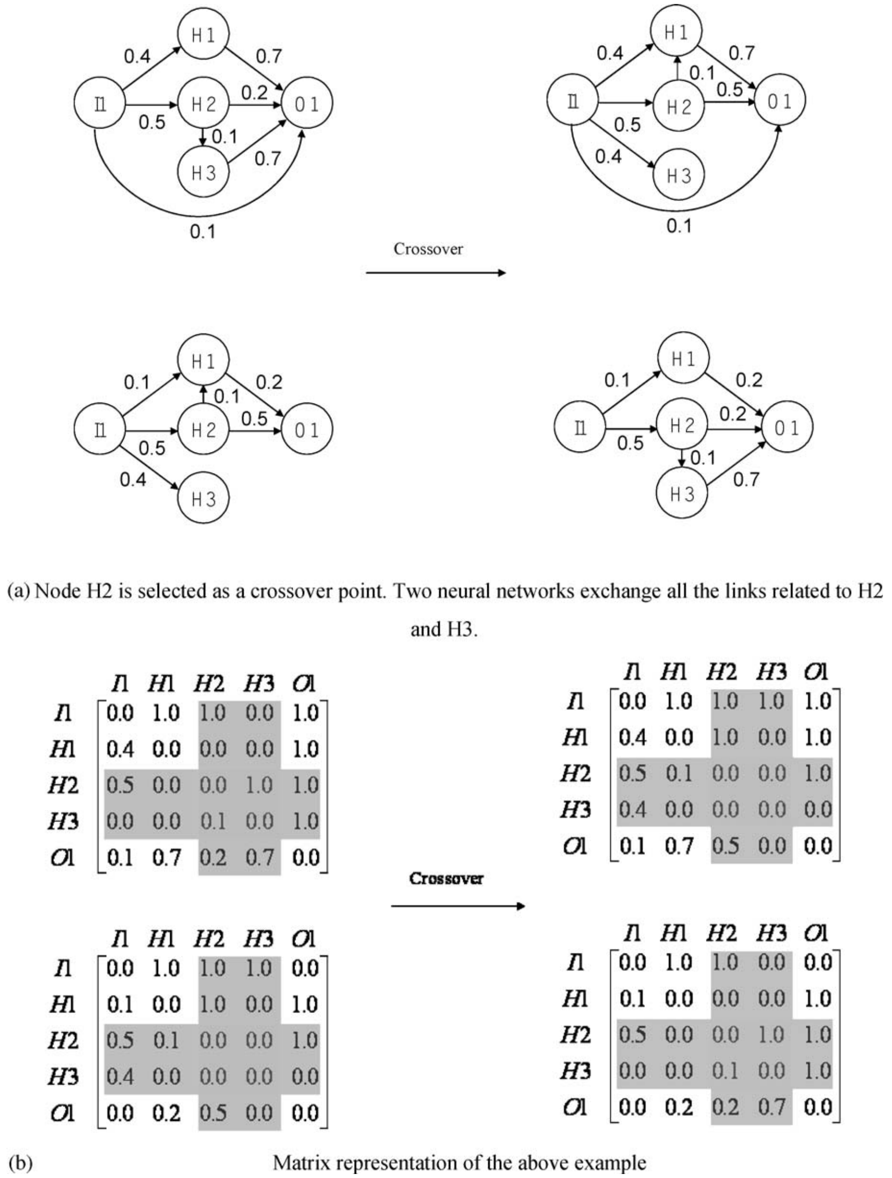
Fig. 3. An example of neural network representation.

such a representation scheme. Node deletion and addition involve flipping a bit in the matrix. Fig. 3 shows an example of encoding of an ANN that has one input node, three hidden nodes, and one output node. Each input node is mapped to one gene of a sample and two output nodes are used for indication of cancer.

The maximum number of hidden nodes must be pre-defined in this representation. The number of input nodes and output nodes is dependent on the problem as described before. Though the maximum number of hidden nodes is predefined, it is not necessary that all hidden nodes are used. Some hidden nodes that have no useful path to output nodes will be ignored. At the initialization stage, connectivity information of the matrix is randomly determined and if the connection value is 1, the corresponding weight is represented with a random real value. This representation allows some direct links between input nodes and output nodes.

3.2.2. Crossover

The crossover operator exchanges the architecture of two ANNs in the population to search ANNs with various architectures [29]. In the population of ANNs, crossover operator selects two distinct ANNs randomly and chooses one hidden node from each ANN selected. These two nodes should be in the same entry of each ANN matrix encoding the ANN to exchange the architectures. Once the nodes are selected, the two



(a) Node H2 is selected as a crossover point. Two neural networks exchange all the links related to H2 and H3.

$$\begin{array}{c|ccccc} & I1 & H1 & H2 & H3 & O1 \\ \hline I1 & 0.0 & 1.0 & 1.0 & 0.0 & 1.0 \\ H1 & 0.4 & 0.0 & 0.0 & 0.0 & 1.0 \\ H2 & 0.5 & 0.0 & 0.0 & 1.0 & 1.0 \\ H3 & 0.0 & 0.0 & 0.1 & 0.0 & 1.0 \\ O1 & 0.1 & 0.7 & 0.2 & 0.7 & 0.0 \end{array} \qquad \begin{array}{c|ccccc} & I1 & H1 & H2 & H3 & O1 \\ \hline I1 & 0.0 & 1.0 & 1.0 & 1.0 & 1.0 \\ H1 & 0.4 & 0.0 & 1.0 & 0.0 & 1.0 \\ H2 & 0.5 & 0.1 & 0.0 & 0.0 & 1.0 \\ H3 & 0.4 & 0.0 & 0.0 & 0.0 & 0.0 \\ O1 & 0.1 & 0.7 & 0.5 & 0.0 & 0.0 \end{array}$$

$$\xrightarrow{\textbf{Crossover}}$$

$$\begin{array}{c|ccccc} & I1 & H1 & H2 & H3 & O1 \\ \hline I1 & 0.0 & 1.0 & 1.0 & 1.0 & 0.0 \\ H1 & 0.1 & 0.0 & 1.0 & 0.0 & 1.0 \\ H2 & 0.5 & 0.1 & 0.0 & 0.0 & 1.0 \\ H3 & 0.4 & 0.0 & 0.0 & 0.0 & 0.0 \\ O1 & 0.0 & 0.2 & 0.5 & 0.0 & 0.0 \end{array} \qquad \begin{array}{c|ccccc} & I1 & H1 & H2 & H3 & O1 \\ \hline I1 & 0.0 & 1.0 & 1.0 & 0.0 & 0.0 \\ H1 & 0.1 & 0.0 & 0.0 & 0.0 & 1.0 \\ H2 & 0.5 & 0.0 & 0.0 & 1.0 & 1.0 \\ H3 & 0.0 & 0.0 & 0.1 & 0.0 & 1.0 \\ O1 & 0.0 & 0.2 & 0.2 & 0.7 & 0.0 \end{array}$$

(b) Matrix representation of the above example

Fig. 4. Crossover operation. (a) Node H2 is selected as a crossover point. Two neural networks exchange all the links related to H2 and H3. (b) Matrix representation of the above example.

ANNs exchange the connection links and corresponding weights information of the nodes and the hidden nodes after that. Fig. 4 shows an example of crossover operation. In this example, two ANNs have one input node, three hidden nodes, and one output

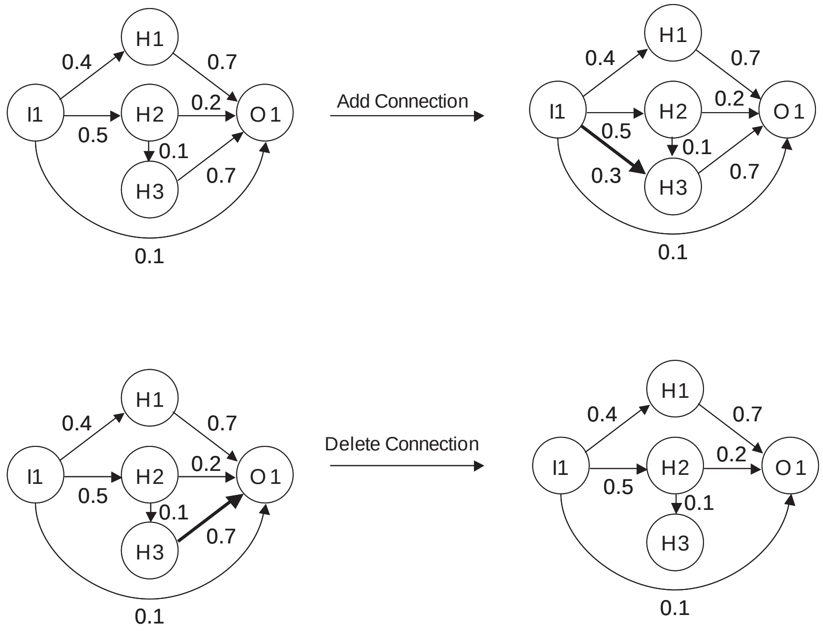


Fig. 5. Mutation operation.

node. For simplicity, it is assumed that the maximum number of hidden nodes is 3. Among the hidden nodes, one hidden node is randomly selected as a crossover point. In the figure, H2 node is chosen as the crossover point. The hidden nodes that have larger index than the point are considered for crossover. In this example, H2 and H3 are considered and the links related to them are exchanged. Fig. 4(a) shows topology change after crossover operation and Fig. 4(b) shows change of the matrix representation.

3.2.3. Mutation

The mutation operator changes a connection link and the corresponding weight of a randomly selected ANN from the population. Mutation operator performs one of the two operations that are addition of a new connection and deletion of an existing connection. Mutation operator selects an ANN from the population of ANNs randomly and chooses one connection link from it. If the connection link does not exist and the connection entry of the ANN matrix is the 0, the connection link is added. It adds new connection link to the ANN with random weights. Otherwise, if the connection link already exists, the connection is deleted. It deletes the connection link and weight information. Fig. 5 shows two examples of the mutation. In the figure, an entry (I1, H3) of the matrix is selected for mutation and there is no connection between them. In this case, a new connection is generated and weight is determined randomly. In the second case, (H3, O1) is selected for mutation and there is already a connection. In the case, the connection is eliminated and there is no link between H3 and O1.

In this study, we use only two mutation types but other methods can be used such as only modifying weights as different value. In EPNET, they use only mutations for evolution and does not use crossover [39]. Mutation is very useful to explore broad area of solution space but overuse of the operation can hinder convergence of the solution. In this reason, we have adopted only two mutation types with small mutation rate.

4. Experimental results

Colon dataset consists of 62 samples of colon epithelial cells taken from colon-cancer patients. Each sample is from one person and contains 2000 gene expression levels. Although original data consist of 6000 gene expression levels, 4000 out of 6000 were removed based on the confidence in the measured expression levels. 40 of 62 samples are colon cancer samples and the remainings are normal samples. Each sample was taken from tumors and normal healthy parts of the colons of the same patients and measured using high-density oligonucleotide arrays. 31 out of 62 samples were used as training data and the remainings were used as test data in this paper. (Available at http://www.sph.uth.tmc.edu:8052/hgc/default.asp)

As mentioned before, the feature size of colon dataset is 2000. There is no single solution for optimal number of features for classification but approximately 20–40 genes are appropriate for classification. In this study, we use 30 genes for classification that has high information gain in feature ranking. There are some systematic ways to determine the optimal number of features. Evolutionary approach is also useful to estimate optimal subset of genes [26]. Table 2 shows the name of 30 genes that are selected. Fig. 6 shows some of the features with color that represents the rank.

Parameters of genetic algorithm are as follows [15]. In EANN, the population size is 20 and the maximum generation number is 200. Each ANN is feed-forward ANN and back-propagation is used as learning algorithm. Learning rate is 0.1 and the partial training presents the training data 200 times and full training presents the training data 1000 times. Crossover rate is 0.3 and mutation rate is 0.1. Fitness function of EANN is defined as the recognition rate for validation data. In colon data set, the number of data sample is very small and we use test data as validation set. Parameters of the EANN are determined empirically. Usually, the number of population size is necessary to be large but it consumes much computational resource. In empirical test with population size as 40 shows no performance improvement and we set the size as 20.

4.1. *Classifiers compared*

SASOM (structure-adaptive self-organizing map) [6] is used by 4 × 4 map with rectangular topology, 0.05 of initial learning rate, 1000 of initial maximum iteration, 10 of initial radius, 0.02 of final learning rate, 10000 of final maximum iteration and 3 of final radius. We have used SVM (support vector machine) [9] with linear

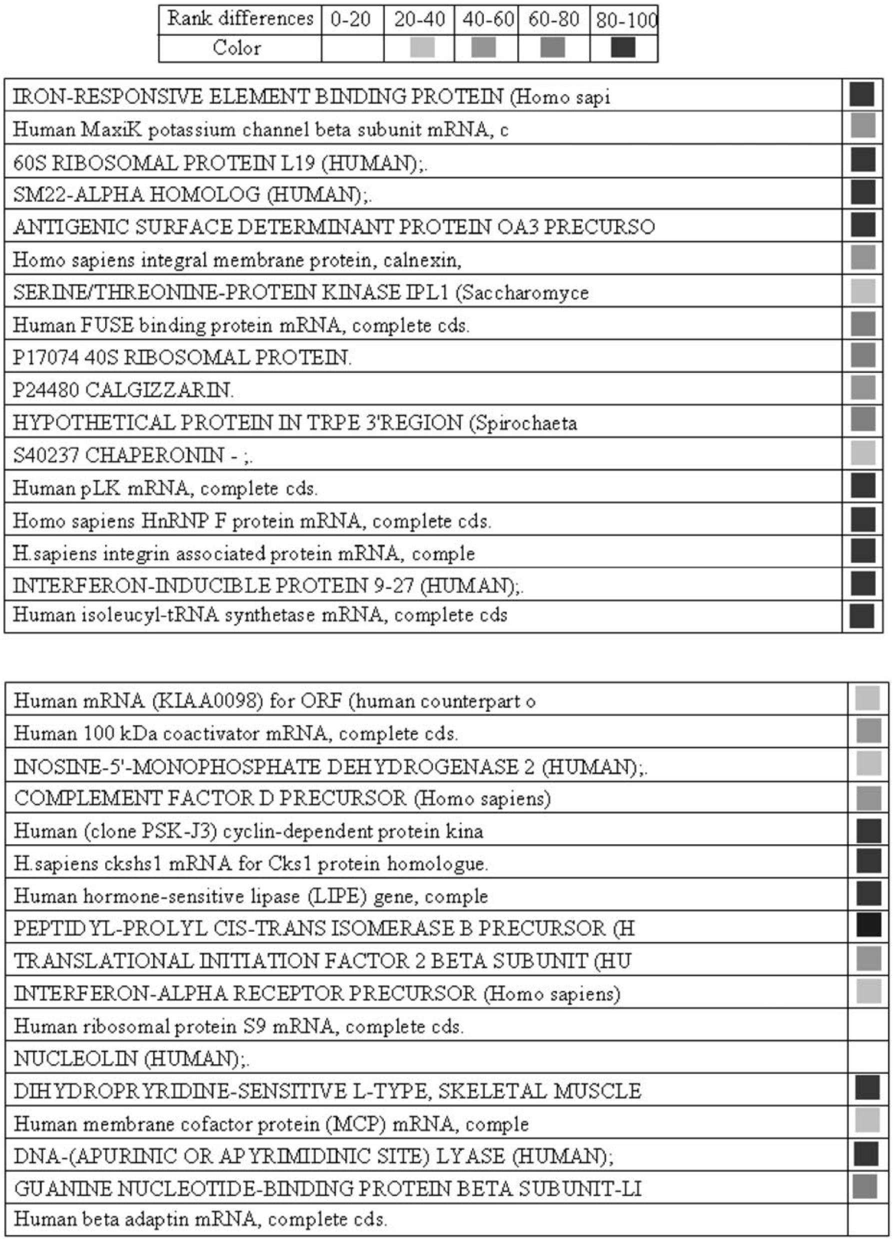
Table 2
30 genes selected by information gain

	Name		Name
1	Human monocyte-derived neutrophil-activating protein (MONAP) mRNA, complete cds.	16	Complement factor D precursor (*Homo sapiens*)
2	Human desmin gene, complete cds.	17	*H.sapiens* mRNA for p cadherin.
3	Myosin heavy chain, nonmuscle (*Gallus gallus*)	18	GTP-binding nuclear Protein ran (*Homo sapiens*)
4	Human cysteine-rich protein (CRP) gene, exons 5 and 6.	19	Prohibitin (*Homo sapiens*)
5	Collagen alpha 2(XI) chain (*Homo sapiens*)	20	Hypothetical protein in trpe $3'$region (*Spirochaeta aurantia*)
6	Human gene for heterogeneous nuclear ribonucleoprotein (hnRNP) core protein A1.	21	40S Ribosomal protein S6 (*Nicotiana tabacum*)
7	P03001 Transcription factor IIIA;.	22	Small nuclear ribonucleoprotein associated proteins B and B' (Human);.
8	Myosin regulatory light chain 2, smooth muscle isoform (Human); contains element TAR1 repetitive element;.	23	Human DNA polymerase delta small subunit mRNA, complete cds.
9	Mitochondrial matrix protein P1 precursor (Human);.	24	Human GAP SH3 binding protein mRNA, complete cds.
10	Human aspartyl-tRNA synthetase alpha-2 subunit mRNA, complete cds.	25	Human (Human);.
11	Human cysteine-rich protein (CRP) gene, exons 5 and 6.	26	Tropomyosin, fibroblast and epithelial muscle-type (Human);.
12	Human cysteine-rich protein (CRP) gene, exons 5 and 6.	27	Human serine kinase mRNA, complete cds.
13	Human homeo box c1 protein, mRNA, complete cds.	28	Thioredoxin (Human);.
14	Macrophage migration inhibitory factor (Human);.	29	S-100P Protein (Human).
15	Human splicing factor SRp30c mRNA, complete cds.	30	Human mRNA for integrin alpha 6.

function and RBF (radial basis function) as kernel function. In RBF, we have changed the gamma variable as 0.1–0.5. For classification, we have used 3-layered MLP (multilayer perceptron) [20] with 5–15 hidden nodes, 2 output nodes, 0.01–0.50 of learning rate and 0.9 of momentum. Similarity measures used in KNN [27] are Pearson's correlation coefficient and Euclidean distance. KNN (k nearest neighbor) has been used with $k = 1$–8.

4.2. Results and analysis

We have conducted 10 runs of experiments to get the average. Fig. 7 shows the results of 10 runs, and min, max and average of 20 individuals in the last generation for each run. Fig. 8 shows the comparison of classifiers' performance which confirms that EANN performs well. In this experiment, all classifiers including EANN use



Rank differences	0-20	20-40	40-60	60-80	80-100
Color		░	▒	▓	█

IRON-RESPONSIVE ELEMENT BINDING PROTEIN (Homo sapi	█
Human MaxiK potassium channel beta subunit mRNA, c	▒
60S RIBOSOMAL PROTEIN L19 (HUMAN);.	█
SM22-ALPHA HOMOLOG (HUMAN);.	█
ANTIGENIC SURFACE DETERMINANT PROTEIN OA3 PRECURSO	█
Homo sapiens integral membrane protein, calnexin,	▒
SERINE/THREONINE-PROTEIN KINASE IPL1 (Saccharomyce	░
Human FUSE binding protein mRNA, complete cds.	▓
P17074 40S RIBOSOMAL PROTEIN.	▓
P24480 CALGIZZARIN.	▒
HYPOTHETICAL PROTEIN IN TRPE 3'REGION (Spirochaeta	▓
S40237 CHAPERONIN - ;.	░
Human pLK mRNA, complete cds.	█
Homo sapiens HnRNP F protein mRNA, complete cds.	█
H.sapiens integrin associated protein mRNA, comple	█
INTERFERON-INDUCIBLE PROTEIN 9-27 (HUMAN);.	█
Human isoleucyl-tRNA synthetase mRNA, complete cds	█

Human mRNA (KIAA0098) for ORF (human counterpart o	░
Human 100 kDa coactivator mRNA, complete cds.	▒
INOSINE-5'-MONOPHOSPHATE DEHYDROGENASE 2 (HUMAN);.	░
COMPLEMENT FACTOR D PRECURSOR (Homo sapiens)	▒
Human (clone PSK-J3) cyclin-dependent protein kina	█
H.sapiens ckshs1 mRNA for Cks1 protein homologue.	█
Human hormone-sensitive lipase (LIPE) gene, comple	█
PEPTIDYL-PROLYL CIS-TRANS ISOMERASE B PRECURSOR (H	█
TRANSLATIONAL INITIATION FACTOR 2 BETA SUBUNIT (HU	▒
INTERFERON-ALPHA RECEPTOR PRECURSOR (Homo sapiens)	░
Human ribosomal protein S9 mRNA, complete cds.	
NUCLEOLIN (HUMAN);.	
DIHYDROPRYRIDINE-SENSITIVE L-TYPE, SKELETAL MUSCLE	█
Human membrane cofactor protein (MCP) mRNA, comple	░
DNA-(APURINIC OR APYRIMIDINIC SITE) LYASE (HUMAN);	█
GUANINE NUCLEOTIDE-BINDING PROTEIN BETA SUBUNIT-LI	▓
Human beta adaptin mRNA, complete cds.	

Fig. 6. Some features of colon cancer data.

features that are extracted using the information gain. To show the performance of the method clearly, 10-fold cross validations are conducted. Recognition rate of the cross validation is 75%. The neural network which shows 94% recognition rate

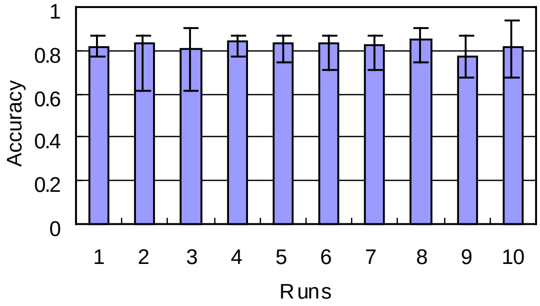


Fig. 7. Max, min and average accuracy of 10 runs.

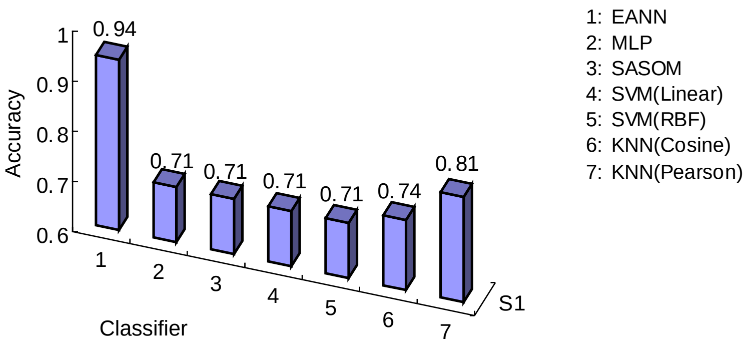
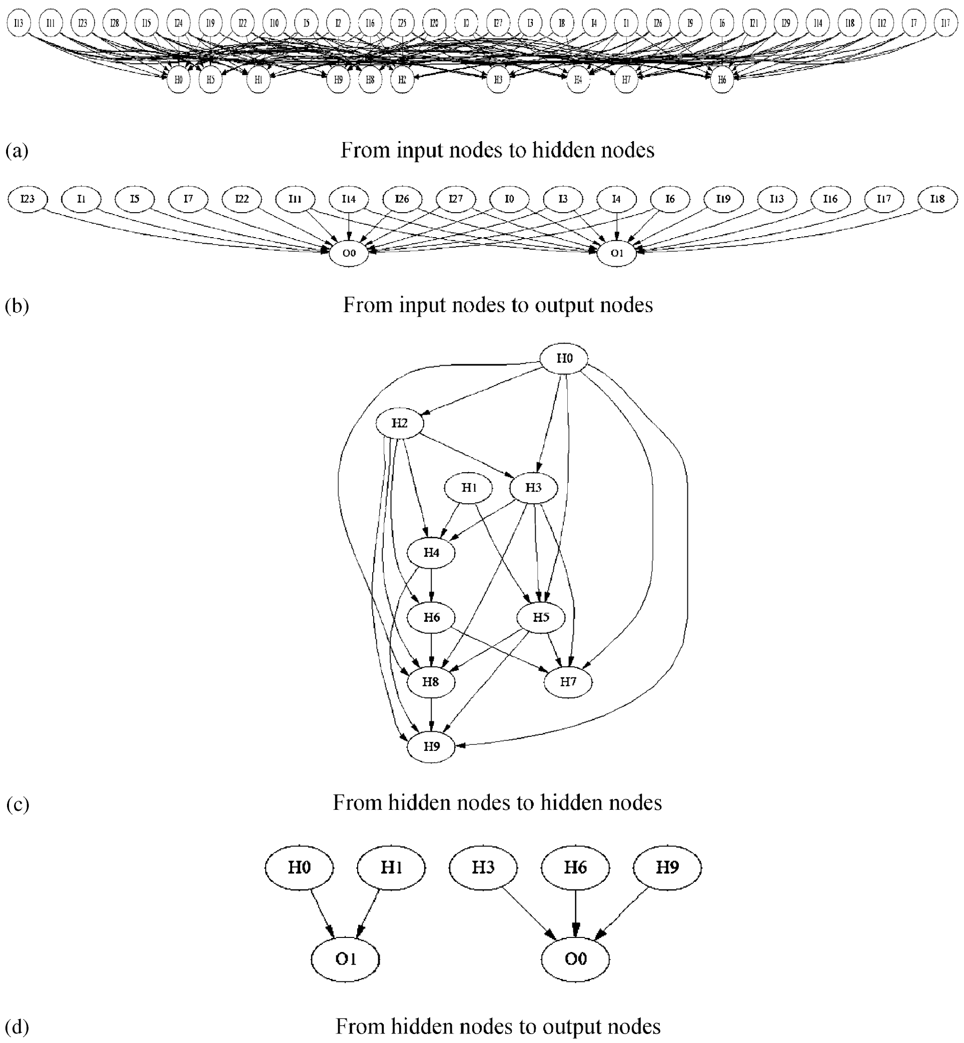


Fig. 8. Comparison of classification rate (maximum accuracy for each classifier) (All classifiers use the same features that are selected using the information gain).

contains 203 connections. The number of connections from input nodes to hidden nodes is 147, that from input nodes to output nodes is 26, that from hidden nodes to hidden nodes is 25, and that from hidden nodes to output nodes is 5. The neural network contains 42 nodes: 30 input nodes, 10 hidden nodes and 2 output nodes. Fig. 9 shows four different connections among nodes using a graph visualization tool [16]. How to extract meaningful information from the network structure is challenging task and one attempt is as follows. In Fig. 9(b), there are some direct links between input nodes and output nodes and it is possible to estimate relationship between features and the cancer. If there is a link between feature and cancer indication output node (which is set as 1.0 when patient is cancer) with high weight, it can be inferred that the feature is the most relevant one for the cancer. Meanwhile, some features are connected only to output node that is for normal person indication. Some are connected to two output nodes simultaneously. To analyze the meaning correctly, comparison with the clinical investigation is demanded.

Table 3 summarizes the confusion matrix of the best evolutionary neural network. The network produces wrong classification for the sample id 24 and 30. Sensitivity of



(a) From input nodes to hidden nodes

(b) From input nodes to output nodes

(c) From hidden nodes to hidden nodes

(d) From hidden nodes to output nodes

Fig. 9. Connection of nodes by the four different types. (a) from input nodes to hidden nodes, (b) from input nodes to output nodes, (c) from hidden nodes to hidden nodes, (d) from hidden nodes to output nodes.

the classifier is 100.0% and specificity is 81.8%. This means that the classifier does not classify patient into normal person but it classifies normal person into patient with the probability of 18.2%. This means that if the person whom the classifier decides as a patient is a normal person with the probability of 9%. The relationship between specificity and sensitivity is negatively correlated and the cost for misclassification for two cases is important point to decide the level of two measures. Prediction error is composed of two components (one is discriminating normal person as a cancer person and vice versa). The two cases have different cost for the misclassification. If normal

Table 3
Confusion matrix of the best EANN

EANN		Predicted	
		0 (Normal)	1 (Cancer)
Actual	0 (Normal)	9	2
	1 (Cancer)	0	20

A confusion matrix contains the information about actual and predicted classifications conducted by a classification system.

person is diagnosed as a cancer person only small cost is enough for deep investigation, whereas missing cancer person produces big loss such as death. The best neural network means the one that produces 94% of accuracy as depicted in Fig. 8.

5. Concluding remarks

It is important to distinguish normal from tumor samples. We have introduced an evolutionary neural network for the classification of tumors based on microarray gene expression data. The methodologies involve dimension reduction of the high-dimensional gene expression space followed by information gain. We have illustrated the effectiveness of the method in predicting normal and tumor samples in colon cancer data set. The methods can distinguish between normal and tumor samples with high accuracy. There are many approaches to predict cancer data using machine learning techniques including SASOM, SVM, MLP and KNN. EANN is a hybrid method of evolutionary algorithm and neural network to find a solution without expert knowledge. Comparison with other classifiers shows that EANN performs very well. Especially, including feature selection for evolution procedure can avoid too large network structure that requires huge computational resource and produces low performance.

The advantage of the proposed method can be summarized as follows. At first, human does not need any prior knowledge about neural network structure. Additional research can reveal the relationships between genes and classes from the emerged structure. For example, rule extraction from neural network can be used for this task. Disadvantage of the method is that it requires more computational resource than the conventional methods because evolutionary algorithm uses multiple points to search solutions.

Acknowledgements

This research was supported by Biometrics Engineering Research Center and Brain Science and Engineering Research Program sponsored by Korean Ministry of Science and Technology.

References

[1] U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, A.J. Levine, Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, Proc. Natl. Acad. Sci. USA 96 (1999) 6745–6750.

[2] J.-H. Ahn, S.-B. Cho, Combining multiple neural networks evolved by speciation, ICONIP 2000, 2000, pp. 230–234.

[3] J.-H. Ahn, S.-B. Cho, Speciated neural networks evolved with fitness sharing technique, Proceedings of the 2001 Congress on Evolutionary Computation, Vol. 1, 2001, pp. 390–396.

[4] J.C. Barrett, E.S. Kawasaki, Microarrays: The use of oligonucleotides and cDNA for the analysis of gene expression, Drug Discovery Today 8 (3) (2003) 134–141.

[5] A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, N. Yakhini, Tissue classification with gene expression profiles, J. Comput. Biol. 7 (2000) 559–584.

[6] S.-B. Cho, Self-organizing map with dynamical node splitting: application to handwritten digit recognition, Neural Comput. 9 (6) (1997) 1345–1355.

[7] G.A. Chung-Faye, D.J. Kerr, L.S. Young, P.F. Searle, Gene therapy strategies for colon cancer, Mol. Med. Today 6 (2) (2000) 82–87.

[8] M. Conrad, Computation: evolutionary, neural, molecular, 2000 IEEE Symposium on Combinations of Evolutionary Computation and Neural Networks, 2000, pp. 1–9.

[9] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines, Cambridge University Press, Cambridge, 2000.

[10] J. Derisi, V. Iyer, P. Brosn, Exploring the metabolic and genetic control of gene expression on a genomic scale, Science 278 (1997) 680–686.

[11] M.B. Eisen, P.T. Spellman, P.O. Brown, D. Bostein, Cluster analysis and display of genome-wide expression patterns, Proc. Natl. Acad. Sci. USA 95 (1998) 14863–14868.

[12] D.B. Fogel, K. Chellapilla, Verifying Anaconda's expert rating by competing against Chinook: experiments in co-evolving a neural checkers player, Neurocomputing 42 (1–4) (2002) 69–86.

[13] T.S. Furey, N. Cristianini, N. Duffy, D.W. Bednarski, M. Schummer, D. Haussler, Support vector machine classification and validation of cancer tissue samples using microarray expression data, Bioinformatics 16 (10) (2000) 906–914.

[14] G. Getz, E. Levine, E. Domany, Coupled two-way clustering analysis of gene microarray data, Proc. Natl. Acad. Sci. USA 97 (22) (2000) 12079–12084.

[15] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Reading, MA, 1989.

[16] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machine, Mach. Learning 46 (1–3) (2002) 389–422.

[17] GraphViz, Graph Visualization Project, http://www.graphviz.org/.

[18] G.M. Hampton, H.F. Frierson Jr., Classifying human cancer by analysis of gene expression, Trends Mol. Med. 9 (1) (2003) 5–19.

[19] C.A. Harrington, C. Rosenow, J. Retief, Monitoring gene expression using DNA microarrays, Curr. Opinion Microbiol. 3 (2000) 285–291.

[20] S.S. Haykin, Neural Networks: A Comprehensive Foundation, 2nd Edition, Prentice-Hall, Englewood Cliffs, NJ, 1998.

[21] H. Kitano, Designing neural networks using genetic algorithms with graph generation system, Complex Syst. 4 (4) (1990) 461–476.

[22] R.J. Lipshutz, S.P.A. Fodor, T.R. Gingeras, D.J. Lockhart, High density synthetic oligonucleotide arrays, Nat. Genet. 21 (1999) 20–24.

[23] S.-I. Lee, J.-H. Ahn, S.-B. Cho, Exploiting diversity of neural ensembles with speciated evolution, International Joint Conference on Neural Networks, Vol. 2, 2001, pp. 808–313.

[24] W. Li, I. Grosse, Gene selection criterion for discriminant microarray data analysis based on extreme value distributions, RECOMB03: Proceedings of the Seventh Annual International Conference on Computational Biology, 2003.

[25] Y. Li, C. Campbell, M. Tipping, Bayesian automatic relevance determination algorithms for classifying gene expression data, Bioinformatics 18 (2002) 1332–1339.

[26] L. Li, C.R. Weinberg, T.A. Darden, L.G. Pedersen, Gene selection for sample classification based on gene expression data: Study of sensitivity to choice of parameters of the GA/KNN method, Bioinformatics 17 (12) (2001) 1131–1142.

[27] T.M. Mitchell, Machine Learning, McGraw-Hill, New York, 1997.

[28] D. Mladenic, M. Grobelnik, Feature selection on hierarchy of web documents, Decision Support Syst. 35 (1) (2003) 45–87.

[29] D. Montana, L. Davis, Training feedforward neural networks using genetic algorithms, Proceedings of the 11th International Conference on Artificial Intelligence, 1989, pp. 762–767.

[30] D.V. Nguyen, D.M. Rocke, Tumor classification by partial least squares using microarray gene expression data, Bioinformatics 18 (1) (2002) 39–50.

[31] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, Los Altos, CA, 1992.

[32] D.A. Rew, DNA microarray technology in cancer research, Eur. J. Surg. Oncol. 27 (5) (2001) 504–508.

[33] I.N. Sarkar, P.J. Planet, T.E. Bael, S.E. Stanley, M. Siddall, R. DeSalle, D.H. Figurski, Characteristic attributes in cancer microarrays, J. Biomed. Inf. 35 (2) (2002) 111–122.

[34] R. Shamir, R. Sharan, Algorithmic approaches to clustering gene expression data, in: T. Jiang, T. Smith, Y. Xu, M.Q. Zhang (Eds.), Current Topics in Computational Biology, MIT Press, Cambridge, MA, 2001.

[35] D.W. Taylor, D.W. Corne, D.L. Taylor, J. Harkness, Predicting alarms in supermarket refrigeration systems using evolved neural networks and evolved rulesets, Congress on Evolutionary Computation, 2002, pp. 1988–1993.

[36] L. Wuju, X. Momiao, Tclass: tumor classification system based on gene expression profiles, Bioinformatics 18 (2002) 325–326.

[37] M. Xiong, W. Li, J. Zhao, L. Jin, E. Boerwinkle, Feature (Gene) selection in gene expression-based tumor classification, Mol. Genet. Metabolism 73 (3) (2001) 239–247.

[38] X. Yao, Evolving artificial neural networks, Proc. IEEE 87 (9) (1999) 1423–1447.

[39] X. Yao, Y. Liu, A new evolutionary system for evolving artificial neural networks, IEEE Trans. Neural Networks 8 (3) (1997) 694–713.

Kyung-Joong Kim received the B.S. and M.S. degree in computer science from Yonsei University, Seoul, Korea, in 2000 and 2002, respectively. Since 2002, he has been a Ph.D. student in the Department of Computer Science, Yonsei University. His research interests include evolutionary neural network, robot control, and agent architecture.

Sung-Bae Cho received the B.S. degree in computer science from Yonsei University, Seoul, Korea, in 1988 and the M.S. and Ph.D. degrees in computer science from KAIST (Korea Advanced Institute of Science and Technology), Taejeon, Korea, in 1990 and 1993, respectively. He worked as a Member of the Research Staff at the Center for Artificial Intelligence Research at KAIST from 1991 to 1993. He was an Invited Researcher of Human Information Processing Research Laboratories at ATR (Advanced Telecommunications Research) Institute, Kyoto, Japan from 1993 to 1995, and a Visiting Scholar at University of New South Wales, Canberra, Australia in 1998. Since 1995, he has been an Associate Professor in the Department of Computer Science, Yonsei University. His research interests include neural networks, pattern recognition, intelligent man-machine interfaces, evolutionary computation, and artificial life. Dr. Cho was awarded outstanding paper prizes from the IEEE Korea Section in 1989 and 1992, and another one from the Korea Information Science Society in 1990. He was also the recipient of the Richard E. Merwin prize from the IEEE Computer Society in 1993. He was listed in Who's Who in Pattern Recognition from the International Association for Pattern Recognition in 1994, and received the best paper awards at International Conference on Soft Computing in 1996 and 1998. Also, he received the best paper award at World Automation Congress in 1998, and listed in Marquis Who's Who in Science and Engineering in 2000 and in Marquis Who's Who in the World in 2001. He is a Member of the Korea Information Science Society, INNS, the IEEE Computer Society, and the IEEE Systems, Man, and Cybernetics Society.