International Journal of Software Engineering
and Knowledge Engineering
Vol. 13, No. 6 (2003) 593–608

World Scientific
www.worldscientific.com

DATA MINING FOR GENE EXPRESSION PROFILES FROM DNA MICROARRAY

SUNG-BAE CHO* and HONG-HEE WON†

Department of Computer Science, Yonsei University,
134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea
** sbcho@cs.yonsei.ac.kr*
† cool@candy.yonsei.ac.kr

Microarray technology has supplied a large volume of data, which changes many problems in biology into the problems of computing. As a result techniques for extracting useful information from the data are developed. In particular, microarray technology has been applied to prediction and diagnosis of cancer, so that it expectedly helps us to exactly predict and diagnose cancer. To precisely classify cancer we have to select genes related to cancer because the genes extracted from microarray have many noises. In this paper, we attempt to explore seven feature selection methods and four classifiers and propose ensemble classifiers in three benchmark datasets to systematically evaluate the performances of the feature selection methods and machine learning classifiers. Three benchmark datasets are leukemia cancer dataset, colon cancer dataset and lymphoma cancer data set. The methods to combine the classifiers are majority voting, weighted voting, and Bayesian approach to improve the performance of classification. Experimental results show that the ensemble with several basis classifiers produces the best recognition rate on the benchmark datasets.

Keywords: Biological data mining; feature selection; classification; gene expression profile; MLP; KNN; SVM; SASOM; ensemble classifier.

1. Introduction

The need for whole genome study such as the Human Genomic Project (HGP) is recently increasing because fragmentary knowledge about life phenomenon with complex control functions of molecular-level is limited. DNA chips have been developed during that process because understanding the functions of genome sequences is essential at that time. The development of DNA microarray technology has produced large amount of gene data and has made it easy to monitor the expression patterns of thousands of genes simultaneously under particular experimental environments and conditions [1]. Also, we can analyze the gene information very rapidly and precisely by managing them at one time [2].

Microarray technology has been applied to the field of accurate prediction and diagnosis of cancer and expected that it would help them. Especially accurate classification of cancer is a very important issue for treatment of cancer. Several

researchers have been studying many problems of cancer classification using gene expression profile data and attempting to propose the optimal classification technique to work out these problems [3, 4]. Some produce better results than others, but there is still no comprehensive work to evaluate the possible feature selection methods and classifiers. We need a thorough effort to give an evaluation of the possible methods to solve the problems of analyzing gene expression data. The gene expression data usually consist of huge number of genes, and the necessity of tools analyzing them to get useful information gets radical. There is research that systematically analyzes the results of test using a variety of feature selection methods and classifiers for selecting informative genes to help classification of cancer and classifying cancer [5, 6]. However, the results were not verified enough because only one benchmark dataset was used. It is necessary to analyze systematically the performance of classifiers using a variety of benchmark datasets.

In this paper, we attempt to explore many classifiers with several different features and propose ensemble classifiers that precisely classify cancer in three benchmark datasets. We adopt seven feature selection methods and four classifiers, which are commonly used in the field of data mining and pattern recognition. Feature selection methods include Pearson's and Spearman's correlation coefficients, Euclidean distance, cosine coefficient, information gain, mutual information and signal to noise ratio. Also, classification methods include multi-layer perceptron (MLP), k-nearest neighbour (KNN), support vector machine (SVM) and structure adaptive self-organizing map (SASOM). The methods for ensemble classifier to combine some of the classifiers are majority voting, weighted voting, and Bayesian rule.

2. Background

2.1. *cDNA microarray*

DNA arrays consist of a large number of DNA molecules spotted in a systemic order on a solid substrate. Depending on the size of each DNA spot on the array, DNA arrays can be categorized as microarrays when the diameter of DNA spot is less than 250 microns, and macroarrays when the diameter is bigger than 300 microns. The arrays with the small solid substrate are also referred to as DNA chips. It is so powerful that we can investigate the gene information in a short time, because at least hundreds of genes can be put on the DNA microarray to be analyzed.

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. The relative abundance of these spotted DNA sequences in two DNA or RNA samples may be assessed by monitoring the differential hybridization of the two samples to the sequences on the array. For mRNA samples, the two samples are reverse-transcribed into cDNA, labeled using different fluorescent dyes mixed (red-fluorescent dye Cy5 and green-fluorescent dye Cy3). After the hybridization of these samples with the arrayed DNA probes, the slides are imaged using a scanner that makes fluorescence measurements for each dye. The log ratio between the two

intensities of each dye is used as the gene expression data [7].

$$\text{gene\_expression} = \log_2 \frac{\text{Int(Cy5)}}{\text{Int(Cy3)}} \tag{1}$$

where Int(Cy5) and Int(Cy3) are the intensities of red and green colors. Since at least hundreds of genes are put on the DNA microarray, we can investigate the genome-wide information in a short time.

2.2. *Oligonucleotide microarray*

Affymetrix Inc. (Santa Clara, CA) has developed GeneChip® oligonucleotide array. High-density oligonucleotide DNA probe array technology employs photolithography and solid-phase DNA synthesis. 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. Synthetic linkers, modified with a photochemically removable protecting group, are attached to a glass surface, and light is directed through a photolithographic mask to a specific area on the surface to produce localized deprotection.

Specific hydroxyl-protected deoxynucleotides are incubated with the surface, and chemical coupling occurs at those sites that have been illuminated in the preceding step. As the chemical cycle is repeated, each spot on the array contains a short synthetic oligonucleotide, typically 20-25 bases long. The oligonucleotides are designed based on the knowledge of the DNA target sequences, to ensure high-affinity and specificity 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 [1].

2.3. *Related works*

It is essential to efficiently analyze DNA microarray data because the amount of DNA microarray data is usually very large. The analysis of DNA microarray data is divided into four branches: clustering, classification, gene identification, and gene regulatory network modeling. Many machine learning and data mining methods have been applied to solve them.

Information theory has been applied to gene identification problem. Also, boolean network, Bayesian network [8], and reverse engineering method have been applied to gene regulatory network modeling problem.

Several machine learning techniques have been previously used for classifying gene expression data, including Fisher linear discriminant analysis [3], k nearest neighbour [9], decision tree, multi-layer perceptron [10], support vector machine [11], boosting, and self-organizing map [12]. Also, many machine learning techniques have been used in clustering gene expression data, such as hierarchical clustering [8], self-organizing map [12], and graph theoretic approaches [13].

Clustering methods do not use any tissue annotation (e.g., tumor vs. normal) in the partitioning step. In contrast, classification methods attempt to predict the label of new tissues, based on their gene expression profiles after training on examples (training data) that have been classified by an external "supervision" [4].

3. Data Mining for DNA Microarray

Data mining for DNA microarray is to select discriminative genes related to classification from gene expression data and train classifier with which classifies new data. After acquiring the gene expression data calculated from the DNA microarray, prediction system has 2 stages: feature selection and pattern classification.

The feature selection can be thought of as the gene selection, which is to get the list of genes that might be informative for the prediction by statistical and information theoretical methods. Since it is highly unlikely that all the genes have the information related to cancer and using all the genes results in too big dimensionality, it is necessary to explore the efficient way to get the best feature. We have extracted some informative genes using seven methods described in Sec. 3.1, and the cancer predictor classifies the category only with these genes.

Given the gene list, a classifier makes decision as to which category the gene pattern belongs at prediction stage. We have adopted four most widely used classification methods and an ensemble classifier.

3.1. *Gene selection*

Among thousands of genes whose expression levels are measured, not all are needed for classification. Microarray data consist of large number of genes in small samples. We need to select some genes highly related to particular classes for classification, which are called informative genes [12].

Using the statistical correlation analysis, we can see the linear relationship and the direction of relation between two variables. Correlation coefficient r varies from –1 to +1, so that the data distributed near the line biased to (+) direction will have positive coefficients, and the data near the line biased to (–) direction will have negative coefficients.

Suppose that we have a gene expression pattern $\boldsymbol{g}_i$ (i= 1 $\sim$ 7,129 in Leukemia data, i= 1 $\sim$ 2,000 in Colon data, and i= 1 $\sim$ 4,026 in Lymphoma data). Each $\boldsymbol{g}_i$ is a vector of gene expression levels from N samples, $\boldsymbol{g}_i = (e_1, e_2, \ldots, e_N)$. The first M elements $(e_1, e_2, \ldots, e_M)$ are examples of tumor samples, and the other N–M $(e_{M+1}, e_{M+2}, \ldots, e_N)$ are those from normal samples. An ideal gene pattern that belongs to tumor class is defined by $g_{\text{ideal\_tumor}} = (1, \ldots, 1, 0, \ldots, 0)$, so that all the elements from tumor samples are 1 and the others are 0. In this paper, we have calculated the correlation coefficient between this g_{ideal} and the expression pattern of each gene. When we have two vectors $\boldsymbol{X}$ and $\boldsymbol{Y}$ that contain N elements, r_{Pearson}

and r_{Spearman} are calculated as follows:

$$r_{\text{Pearson}} = \frac{\sum XY - \frac{\sum X \sum Y}{N}}{\sqrt{\left(\sum X^2 - \frac{(\sum X)^2}{N}\right)\left(\sum Y^2 - \frac{(\sum Y)^2}{N}\right)}} \tag{2}$$

$$r_{\text{Spearman}} = 1 - \frac{6\sum (D_x - D_y)^2}{N\left(N^2 - 1\right)} \tag{3}$$

where D_x and D_y are the rank matrices of $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively.

The similarity between two input vectors $\boldsymbol{X}$ and $\boldsymbol{Y}$ can be thought of as distance, which measures how far the two vectors are located. The distance between $g_{\text{ideal\_tumor}}$ and g_i tells us how much likely the g_i is to the tumor class. Calculating the distance between them, if it is bigger than certain threshold, the gene g_i would belong to the tumor class, otherwise g_i belongs to the normal class. We can adopt Euclidean distance ($r_{\text{Euclidean}}$) and cosine coefficient (r_{Cosine}) represented by the following equations:

$$r_{\text{Euclidean}} = \sqrt{\sum (X - Y)^2} \tag{4}$$

$$r_{\text{Cosine}} = \frac{\sum XY}{\sqrt{\sum X^2 \sum Y^2}} \tag{5}$$

We can also utilize the information gain and mutual information that are widely used in many fields such as text categorization and data mining. If we count the number of genes excited ($P(g_i)$) or not excited ($P(\bar{g}_i)$) in category $c_j(P(c_j))$, the coefficients of the information gain and mutual information become as follows:

$$IG(g_i, c_j) = P(g_i|c_j) \log \frac{P(g_i|c_j)}{P(c_j) \cdot P(g_i)} + P(\bar{g}_i|c_i) \log \frac{P(\bar{g}_i|c_j)}{P(c_j) \cdot P(\bar{g}_i)} \tag{6}$$

$$MI(g_i, c_j) = \log \frac{P(g_i, c_j)}{P(c_j) \cdot P(g_i)} \tag{7}$$

Mutual information indicates the dependency relationship between two probabilistic variables of events. If two events are completely independent, the mutual information is 0. The more they are related, the higher the mutual information gets. Information gain is used when the features of samples are extracted by inducing the relationship between gene and class by the presence frequency of the gene in the sample. Information gain measures the goodness of gene using the presence and absence within the corresponding class.

For each gene g_i, some are from tumor samples, and some are from normal samples. If we calculate the mean μ and standard deviation σ from the distribution of gene expressions within their classes, the signal to noise ratio of gene g_i, $SN(g_i)$, is defined by:

$$SN(g_i) = \frac{\mu_{\text{tumor}}(g_i) - \mu_{\text{normal}}(g_i)}{\sigma_{\text{tumor}}(g_i) + \sigma_{\text{normal}}(g_i)} \tag{8}$$

3.2. *Pattern classification*

Many algorithms designed for solving classification problems in machine learning have been applied to recent research of prediction and classification of cancer with gene expression data. The general process of classification in machine learning is to train classifiers to accurately recognize patterns from given training samples and to classify test samples with the trained classifier. Representative classification algorithms such as multi-layer perceptron, k-nearest neighbour, support vector machine, and structure-adaptive self-organizing maps are applied to the classification.

3.2.1. *MLP*

A feed-forward multilayer perceptron (MLP) is error backpropagation neural network that is applied to many fields due to its powerful and stable learning algorithm [14]. The neural network learns the training examples by adjusting the synaptic weight of neurons according to the error occurred on the output layer. The power of the backpropagation algorithm lies in two main aspects: local for updating the synaptic weights and biases, and efficient for computing all the partial derivatives of the cost function with respect to these free parameters. The weight-update rule in backpropagation algorithm is defined as follows:

$$\Delta w_{ji}(n) = \eta \delta_j x_{ji} + \alpha \Delta w_{ji}(n-1) \tag{9}$$

where $\Delta w_{ji}(n)$ is the weight update performed during the nth iteration through the main loop of the algorithm, η is a positive constant called the learning rate, δ_j is the error term associated with j, x_{ji} is the input from node i to node j, and $0 \leq \alpha < 1$ is a constant called the *momentum*.

3.2.2. *KNN*

k-nearest neighbor (KNN) is one of the most common methods among memory based induction. Given an input vector, KNN extracts k closest vectors in the reference set based on similarity measures, and makes decision for the input vector label using the labels of the k nearest neighbors.

Pearson's coefficient correlation and Euclidean distance have been used as the similarity measure. When we have an input X and a reference set $D = \{d_1, d_2, \ldots, d_N\}$, the probability that X may belong to class c_j, $P(X, c_j)$ is defined as follows:

$$P(X, c_j) = \sum_{d_i \in kNN} \mathrm{Sim}(X, d_i) P(d_i, c_j) - b_j \tag{10}$$

where $\mathrm{Sim}(X, d_i)$ is the similarity between X and d_i and b_j is a bias term.

3.2.3. *SASOM*

Self-organizing map (SOM) defines a mapping from the input space onto an output layer by unsupervised learning algorithm. SOM has an output layer consisting of

N nodes, each of which represents a vector that has the same dimension as the input pattern. For a given input vector X, the winner node m_c is chosen using the Euclidean distance between $\boldsymbol{x}$ and its neighbors, $\boldsymbol{m}_i$.

$$\|x - m_c\| = \min_i \|x - m_i\| \tag{11}$$

$$m_i(t+1) = m_i(t) + \alpha(t) \times n_{ci}(t) \times \{x(t) - m_i(t)\} \tag{12}$$

Even though SOM is well known for its good performance of topology preserving, it is difficult to apply it to practical classification since the topology should be fixed before training. A structure adaptive self-organizing map (SASOM) is proposed to overcome this shortcoming [15]. SASOM starts with 4×4 map, and dynamically splits the output nodes of the map, where the data from different classes are mixed, trained with the LVQ learning algorithm.

3.2.4. *SVM*

Support vector machine (SVM) estimates the function classifying the data into two classes [16]. SVM builds up a hyperplane as the decision surface in such a way as to maximize the margin of separation between positive and negative examples. SVM achieves this by the structural risk minimization principle that the error rate of a learning machine on the test data is bounded by the sum of the training-error rate and a term that depends on the Vapnik-Chervonenkis (VC) dimension. Given a labeled set of M training samples $(\boldsymbol{X}_i,\ \boldsymbol{Y}_i)$, where $\boldsymbol{X}_i \in R^N$ and $\boldsymbol{Y}_i$ is the associated label, $Y_i \in \{-1, 1\}$, the discriminant hyperplane is defined by:

$$f(X) = \sum_{i=1}^{M} Y_i \alpha_i k(X, X_i) + b \tag{13}$$

where $k(.)$ is a kernel function and the sign of $f(X)$ determines the membership of X. Constructing an optimal hyperplane is equivalent to finding all the nonzero α_i (support vectors) and a bias b. We have used $\text{SVM}^{\text{light}}$ module and SVM^{RBF} in this paper.

3.2.5. *Ensemble classifier*

Classification can be defined as the process to approximate I/O mapping from the given observation to the optimal solution. Generally, classification tasks consist of two parts: feature selection and classification. Feature selection is a transformation process of observations to obtain the best pathway to get to the optimal solution. Therefore, considering multiple features encourages obtaining various candidate solutions so that we can estimate a more accurate solution to the optimal than any other local optima.

When we have multiple features available, it is important to know which features should be used. Theoretically, for many features concerned, it may be more effective

for the classifier to solve the problems. But features that are overlapped in feature spaces may cause the redundancy of irrelevant information and result in the counter effect such as overfitting. Therefore, it is more important to explore and utilize independent features to train classifiers, rather than increase the number of features we use. Correlation between feature sets can be induced from the distribution of features.

Meanwhile, there are many algorithms for the classification from machine learning approach, but none of them is perfect. Moreover, it is always difficult to decide what to use and how to set up its parameters. According to the environments when the classifier is embedded, some algorithms work well and others do not. This is because the classifier searches in different solution space depending on the algorithms, features and parameters used. These sets of classifiers produce their own outputs, therefore the ensemble classifier can explore a wider solution space.

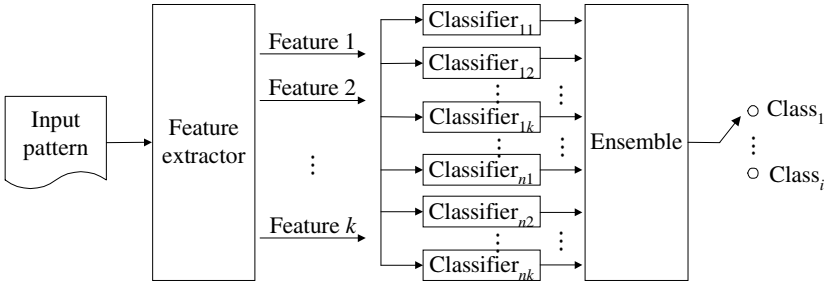

Fig. 1. Overview of the ensemble classifier.

We have applied this idea to a classification framework as shown in Fig. 1. Given k features and n classifiers, there are $k \times n$ feature-classifier combinations. There are ${}_{k\times n}C_m$ possible ensemble classifiers when m feature-classifier combinations are selected for ensemble classifier. Classifiers are trained using the features selected, and finally a combining module is accompanied to combine their outputs. After classifiers are trained independently with some features to produce their own outputs, the final answer will be judged by a combining module, where the majority voting, weighted voting, or Bayesian combination can be adopted.

- Majority Voting: It is a simple ensemble method that selects the class most favored by base classifiers. Majority voting has some advantages that it does not require any previous knowledge nor does it require any additional complex computation to decide. Where c_i is the class i $(i = 1, \ldots, m)$, and $s_i(\mathit{classifier}_j)$ is 1 if the output of the jth classifier $\mathit{classifier}_j$ equals to the class i otherwise 0, majority voting is defined as follows:

$$c_{\text{ensemble}} = \arg\max_{1\le i\le m} \left\{ \sum_{j=1}^{k} s_i(\mathit{classifier}_j) \right\} \tag{14}$$

- Weighted Voting: Poor classifier can affect the result of the ensemble in majority voting because it gives the same weight to all classifiers. Weighted voting reduces the effect of poor classifier by giving a different weight to a classifier based on the performance of each classifier. Where w_j is the weight of the jth classifier, weighted voting is defined as follows:

$$c_{\text{ensemble}} = \underset{1 \le i \le m}{\arg\max} \left\{ \sum_{j=1}^{k} w_j s_i(\textit{classifier}_j) \right\} \tag{15}$$

- Bayesian Combination: While majority voting method combines classifiers with their results, Bayesian combination makes the error possibility of each classifier affect the final result. The method combines classifier with different weight by using the previous knowledge of each classifier. Where k classifiers are combined, $c(\textit{classifier}_j)$ is the class of the j th classifier, and η is a priori possibility of the class c_i, Bayesian combination is defined as follows:

$$c_{\text{ensemble}} = \underset{1 \le i \le m}{\arg\max} \left\{ \eta \prod_{j=1}^{k} P(c_i | c(\textit{classifier}_j)) \right\} \tag{16}$$

4. Experimental Results

4.1. *Datasets*

Three representative datasets, leukemia cancer dataset, colon cancer dataset and lymphoma cancer dataset, are used in this paper among several microarray datasets. Leukemia dataset consists of 72 samples: 25 samples of acute myeloid leukemia (AML) and 47 samples of acute lymphoblastic leukemia (ALL). 38 out of 72 samples were used as training data and the remaining were used as test data. Each sample contains 7129 gene expression levels. Colon dataset consists of 62 samples of colon epithelial cells taken from colon-cancer patients. Each sample contains 2000 gene expression levels. 31 out of 62 samples were used as training data and the remaining were used as test data. Lymphoma dataset consists of 24 samples of GC B-like and 23 samples of activated B-like. 22 out of 47 samples were used as training data and the remaining were used as test data.

4.2. *Environments*

For feature selection, each gene is scored based on the feature selection methods described in Sec. 3.1, and the 25 top-ranked genes are chosen as the feature of the input pattern. There is no report on the optimal number of genes, but our previous study indicates that 25 is reasonable [5, 6]. For classification, we have used 3-layered MLP with $5 \sim 15$ hidden nodes, 2 output nodes, $0.01 \sim 0.50$ of learning rate and 0.9 of momentum. KNN has been used with $k = 1 \sim 8$. Similarity measures used in KNN are Pearson's correlation coefficient and Euclidean distance. SASOM has

been 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, 10,000 of final maximum iteration and 3 of final radius. We have used SVM's with linear function and RBF function as kernel function. In RBF, we have changed 0.1∼0.5 gamma variable. We have produced 42 feature-classifier combinations using 7 feature selection methods and 6 classifiers. We have tried to combine 3, 5 and 7 classifiers among 42 classifiers, and analyzed the results of ensemble. We have conducted all ${}_{42}C_m$ (m = 3, 5, 7, and 42) combinations of ensemble, and have investigated the best recognition rate and average recognition rate. We have chosen an odd number of m such as 3, 5 and 7 for the purpose of tie-break.

4.3. *Analysis of results*

Table 1 shows the IDs of common genes by Pearson's correlation coefficient, cosine coefficient, and Euclidean distance for each dataset. Among these genes there are some genes overlapped by several feature selection methods. For example, gene 2288 of Leukemia has been ranked third in information gain. The number of overlapped genes of Leukemia dataset is 17. The number of overlapped genes of colon dataset is 9. The number of overlapped genes of Lymphoma dataset is 19. These overlapped genes are very informative. In particular, Zyxin, gene 4847 of Leukemia, has been reported as informative [12], but there are no genes which appeared commonly in every method.

The results of recognition rate on the test data are as shown in Tables 2, 3 and 4. The list of feature selection methods is raw: Pearson's correlation coefficient (PC), Spearman's correlation coefficient (SC), Euclidean distance (ED), cosine coefficient (CC), information gain (IG), mutual information (MI), and signal-to-noise ratio (SN). $\text{KNN}_{\text{Pearson}}$ and MLP seem to produce the best recognition rate among the classifiers on average. $\text{KNN}_{\text{Pearson}}$ is better than $\text{KNN}_{\text{cosine}}$. SVM is poorer than any other classifiers.

Table 1. The IDs of genes overlapped by Pearson's correlation coefficient, cosine coefficient, and Euclidean distance.

Leukemia	461	1249	1745	1834	2020
	2043	2242	2288	3258	3320
	4196	4847	5039	6200	6201
	6373	6803			
Colon	187	619	704	767	1060
	1208	1546	1771	1772	
Lymphoma	36	75	76	77	86
	86	678	680	1636	1637
	2225	2243	2263	2412	2417
	2467	3890	3893	3934	

Table 2. Recognition rate with features and classifiers in Leukemia dataset (%).

	MLP	SASOM	SVM		KNN	
			Linear	RBF	Cosine	Pearson
PC	97.1	76.5	79.4	79.4	97.1	94.1
SC	82.4	61.8	58.8	58.8	76.5	82.4
ED	91.2	73.5	70.6	70.6	85.3	82.4
CC	94.1	88.2	85.3	85.3	91.2	94.1
IG	97.1	91.2	97.1	97.1	94.1	97.1
MI	58.8	58.8	58.8	58.8	73.5	73.5
SN	76.5	67.7	58.8	58.8	73.5	73.5
Mean	85.3	74.0	72.7	72.7	84.5	85.3

Table 3. Recognition rate with features and classifiers in Colon dataset (%.)

	MLP	SASOM	SVM		KNN	
			Linear	RBF	Cosine	Pearson
PC	74.2	74.2	64.5	64.5	71.0	77.4
SC	58.1	45.2	64.5	64.5	61.3	67.7
ED	67.8	67.6	64.5	64.5	83.9	83.9
CC	83.9	64.5	64.5	64.5	80.7	80.7
IG	71.0	71.0	71.0	71.0	74.2	80.7
MI	71.0	71.0	71.0	71.0	74.2	80.7
SN	64.5	45.2	64.5	64.5	64.5	71.0
Mean	70.1	62.7	66.4	66.4	72.7	77.4

Table 4. Recognition rate with features and classifiers in Lymphoma dataset (%).

	MLP	SASOM	SVM		KNN	
			Linear	RBF	Cosine	Pearson
PC	64.0	48.0	56.0	60.0	60.0	76.0
SC	60.0	68.0	44.0	44.0	60.0	60.0
ED	56.0	52.0	56.0	56.0	56.0	68.0
CC	68.0	52.0	56.0	56.0	60.0	72.0
IG	92.0	84.0	92.0	92.0	92.0	92.0
MI	72.0	64.0	64.0	64.0	80.0	64.0
SN	76.0	76.0	72.0	76.0	76.0	80.0
Mean	69.7	63.4	62.9	63.4	69.1	73.1

Although the results are different among datasets, information gain is the best, and Pearson's correlation coefficient ranks second. Mutual information and Spearman's correlation coefficient are poor. The difference of performance in data sets might be caused by the characteristics of data.

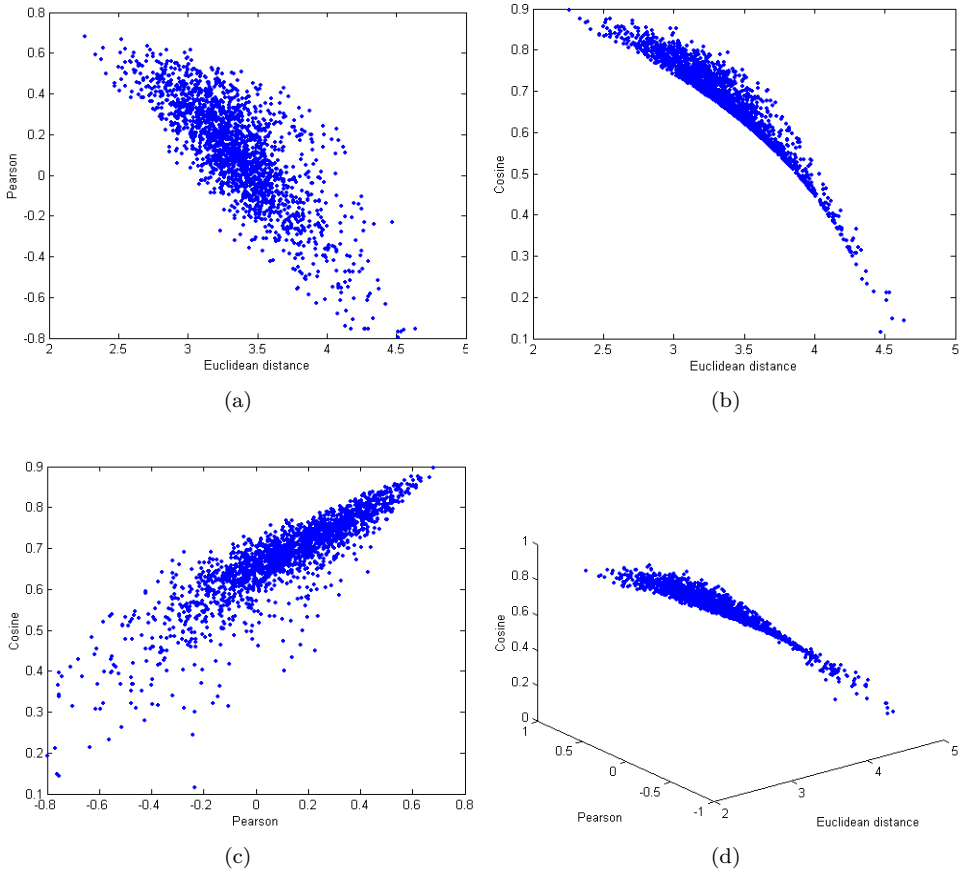

Fig. 2. Correlation of Euclidean distance, Pearson's correlation coefficient and cosine coefficient in colon dataset. (a) Correlation of Euclidean distance and Pearson's correlation coefficient. (b) Correlation of Euclidean distance and cosine coefficient. (c) Correlation of Pearson's correlation coefficient and cosine coefficient. (d) Correlation of Euclidean distance, Pearson's correlation coefficient and cosine coefficient.

As shown in Fig. 2, the Euclidean distance, Pearson's correlation coefficient and cosine coefficient are correlated in Colon dataset. There are some overlapped genes among them as shown in Table 1. This indicates that overlapped genes of correlated features can discriminate classes and the other genes not overlapped among combined features can supplement to search the solution spaces complementarily. For example, gene 1659 and gene 550 are ranked high in both Pearson's correlation coefficient and cosine coefficient, and gene 440 is ranked high in both Euclidean distance and cosine coefficient. This subset of two features might play an important role in classification.

Recognition rates by ensemble classifiers are shown in Table 5 and Fig. 3. Average recognition rate means the average of all possible ${}_{42}C_m$ (m = 3, 5, 7 and 42) combinations of ensemble classifiers. MV means the ensemble classifier using

Table 5. The best recognition rate by ensemble classifier.

		Leukemia	Colon	Lymphoma
MV	3	97.1	93.5	96.0
	5	97.1	93.5	100.0
	7	97.1	93.5	100.0
	All	91.2	71.0	80.0
WV	3	97.1	93.5	96.0
	5	97.1	93.5	100.0
	7	97.1	93.5	100.0
	All	97.1	71.0	88.0
BC	3	97.1	93.5	96.0
	5	97.1	93.5	100.0
	7	97.1	93.5	100.0
	All	97.1	74.2	92.0

majority voting method, WV means the ensemble classifier using weighted voting method, and BC means the ensemble classifier using Bayesian combination. 3, 5, 7 and all (42) are the numbers of combined classifiers.

The best recognition rate of ensemble classifier is 97.1% in Leukemia dataset, 93.5% in Colon dataset, and 100.0% in Lymphoma dataset. Compared with the best recognition rates of base classifier, at 97.1%, 83.9%, and 92.0% on each dataset in Tables 2, 3 and 4 respectively, the performance of ensemble is superior. The best result of Leukemia 97.1% is obtained by all the basis classifiers except SASOM. The result of the best basis classifier is the same as that of the best ensemble classifier. In other datasets, the performance of ensemble classifier outperforms the best classifier. For all the datasets, ensemble classifier with all classifiers produces the worst result.

While there is little difference in the best recognition rate of the ensemble classifier according to the ensemble method or the number of combined classifiers, there is a difference in the average recognition rate of the ensemble classifier. As the number of combined classifiers is increasing, the average recognition rate of the ensemble classifier is also increasing in Leukemia dataset and Lymphoma dataset but slightly decreasing in Colon dataset. Figure 3 shows the average recognition rate of the ensemble classifier on the benchmark dataset. For all the datasets, Bayesian combination is the best among three ensemble methods. Increasing the number of combined classifiers, weighted voting is superior to majority voting.

If we observe the classifiers of the best ensemble classifier where its recognition rate is 93.6% in Colon dataset, we find more important features which affect the result than classifiers. In other words, in ensemble classifiers there must be classifiers with ED and PC. The other classifier is the one with CC, MI or IG. This fact is also prominent in Lymphoma dataset. Most of the classifiers of the best ensemble classifiers are classifiers with IG, SN and ED, or the classifiers with IG, SN and PC.

This paper shows that the ensemble classifier works and we can improve the

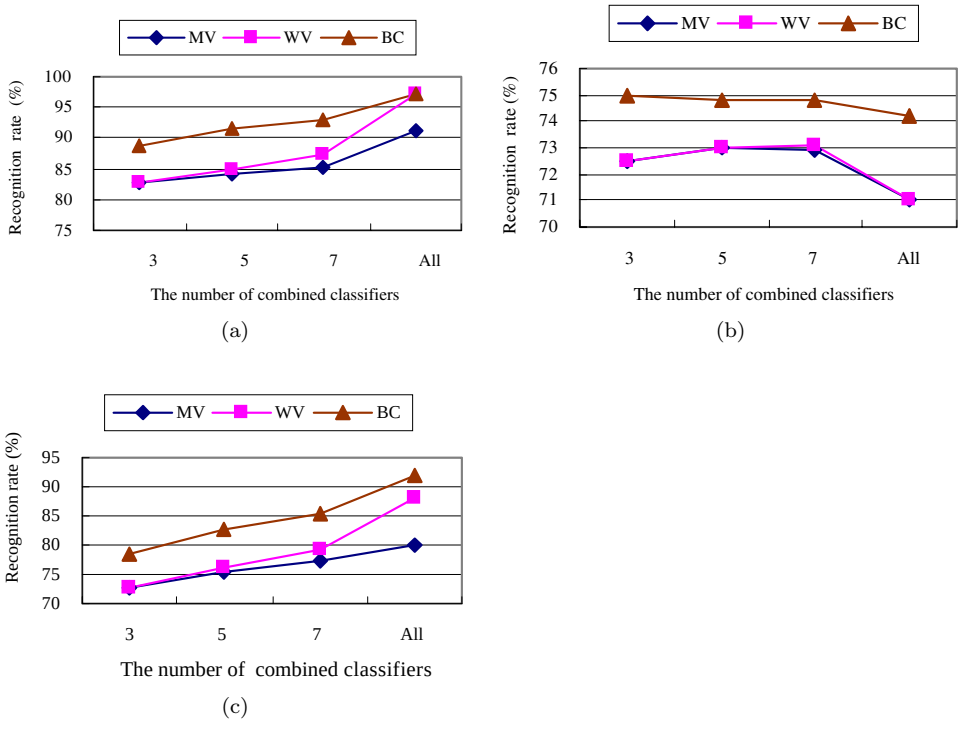

Fig. 3. Average recognition rate of the ensemble. (a) Leukemia dataset. (b) Colon dataset. (c) Lymphoma dataset.

classification performance by combining complementary common sets of classifiers learned from three independent features, even when we use simple combination method like majority voting, weighted voting, and Bayesian combination.

5. Concluding Remarks

We have conducted a thorough quantitative comparison among the 42 combinations of features and classifiers for three benchmark datasets. Information gain and Pearson's correlation coefficient are the top feature selection methods, and MLP and KNN are the best classifiers. The experimental results also imply some correlations between features and classifiers, which might guide the researchers to choose or devise the best classification method for their problems in bioinformatics. Based on the results, we have developed the optimal feature-classifier combination to produce the best performance on the classification. We have combined 3, 5 and 7 classifiers among 42 classifiers using majority voting, weighted voting, and Bayesian combination. Bayesian combination is the best among three ensemble methods for all datasets. The best recognition rate of ensemble classifier is 97.1% in Leukemia dataset, 93.5% in Colon dataset, and 100.0% in Lymphoma dataset. The results show that the ensemble classifiers surpass the basis classifiers. We can confirm that

the ensemble classifier of complementarily correlated features works better than ensemble of uncorrelated features or base classifiers.

Moreover, our method of combining classifiers is very simple, and there are many methods of combining classifiers in machine learning and data-mining fields. We will have to apply more sophisticated methods of combining classifiers to the same datasets to confirm the results obtained and get better results.

Acknowledgements

This work was supported by Biometrics Engineering Research Center and a grant of Korea Health 21 R&D Project, Ministry of Health & Welfare, Republic of Korea.

References

1. C. A. Harrington, C. Rosenow, and J. Retief, "Monitoring gene expression using DNA microarrays", *Curr. Opin. Microbiol.* **3** (2000) 285–291.
2. M. B. Eisen and P. O. Brown, "DNA arrays for analysis of gene expression", *Methods Enzymbol* **303** (1999) 179–205.
3. S. Dudoit, J. Fridlyand, and T. P. Speed, "Comparison of discrimination methods for the classification of tumors using gene expression data", Technical Report 576, Department of Statistics, University of California, Berkeley, 2000.
4. A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and N. Yakhini, "Tissue classification with gene expression profiles", *Journal of Computational Biology* **7** (2000) 559–584.
5. J. Ryu and S.-B. Cho, "Towards optimal feature and classifier for gene expression classification of cancer", *Lecture Note in Artificial Intelligence* **2275** (2002) 310–317.
6. S.-B. Cho and J. Ryu, "Classifying gene expression data of cancer using classifier ensemble with mutually exclusive features", *Proc. IEEE* **90**(11) (2002) 1744–1753.
7. J. Derisi, V. Iyer, and P. Brosn, "Exploring the metabolic and genetic control of gene expression on a genomic scale", *Science* **278** (1997) 680–686.
8. N. Friedman, M. Linial, I. Nachman, and D. Pe'er, "Using Bayesian networks to analyze expression data", *Journal of Computational Biology* **7** (2000) 601–620.
9. L. Li, C. R. Weinberg, T. A. Darden, and 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.
10. J. Khan, J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer, "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks", *Nature Medicine* **7**(6) (2001) 673–679.
11. T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler, "Support vector machine classification and validation of cancer tissue samples using microarray expression data", *Bioinformatics* **16**(10) (2000) 906–914.
12. T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. GaasenBeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Blomfield, and E. S. Lander, "Molecular classification of cancer: Class discovery and class prediction by gene-expression monitoring", *Science* **286** (1999) 531–537.
13. E. Hartuv, A. Schmitt, J. Lange, S. Meier-Ewert, H. Lehrach, and R. Shamir, "An algorithm for clustering cDNA fingerprints", *Genomics* **66**(3) (2000) 249–256.

14. R. P. Lippman, "An introduction to computing with neural nets", *IEEE ASSP Magazine*, 1987, pp. 4–22.
15. H. D. Kim and S.-B. Cho, "Genetic optimization of structure-adaptive self-organizing map for efficient classification", in *Proc. Int. Conf. on Soft Computing*, World Scientific Publishing, 2000, pp. 34–39.
16. V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, 1995.