Classifying Gene Expression Data of Cancer Using Classifier Ensemble With Mutually Exclusive Features

SUNG-BAE CHO, MEMBER, IEEE, AND JUNGWON RYU

Invited Paper

The explosion of DNA and protein sequence data in public and private databases has been encouraging interdisciplinary research on biology and information technology. Gene expression profiles are just sequences of numbers, and the necessity of tools analyzing them to get useful information has risen significantly. In order to predict the cancer class of patients from the gene expression profile, this paper presents a classification framework that combines a pair of classifiers trained with mutually exclusive features. The idea behind feature selection with nonoverlapping correlation is to encourage classifier ensemble, which consists of multiple classifiers, to learn different aspects of training data, so that classifiers can search in a wide solution space. Experimental results show that the classifier ensemble produces higher recognition accuracy than conventional classifiers.

Keywords*—Cancer classification, classifier ensemble with mutually exclusive features, gene expression classification, neural network ensemble classifier.*

I. INTRODUCTION

Precisely classifying tumors is crucial to cancer diagnosis and treatment, because targeting specific therapies to pathogenetically distinct tumor types is important for cancer treatment. Conventional classification methods, however, are unable to discriminate among tumors with similar histopathologic features as they rely on a variety of morphological, clinical, and molecular variables. Recently, microarray technologies have been developed that can simultaneously assess the level of expression of thousands of genes in colon, breast, and other tumors [1], [2]. They may lead to a more complete understanding of the molecular variations among tumors and hence to a more reliable classification.

Manuscript received March 15, 2002; revised July 15, 2002. This work was supported by Brain Science and Engineering Research Program sponsored by Korean Ministry of Science and Technology.

The authors are with the Department of Computer Science, Yonsei University, Seoul 120-749, Korea (e-mail: sbcho@cs.yonsei.ac.kr; rjungwon@yahoo.com).

Digital Object Identifier 10.1109/JPROC.2002.804682

From the pattern classification point of view, however, the gene expression data have an unusual characteristic of the huge number of genes relative to the number of tumor samples, which poses a challenge to the researchers and leads to several attempts to apply a variety of conventional techniques in pattern recognition. Especially, it is the more challenging problem to distinguish acute leukemias, whose appearance is highly similar. Many papers dealing with predictive discrimination of a leukemia dataset have been published [3]–[8].

Furey *et al.* [4] used support vector machine (SVM) to classify the tissue samples with signal-to-noise ratio (SNR) for the feature selection method, resulting in the correct classification of 75.6% (8.3 misclassifications over the test data). Li *et al.* [5] carried out the model selection with the Akaike information criterion and the Bayesian information criterion with logistic regression to determine the number of genes that provide the best model. The best predictor was the single-gene logistic regression using g_{1882} and g_{760} producing the recognition rate of 94.1% on the test data.

Ben-Dor *et al.* [6] used the nearest neighbor (recognition accuracy of 91.6%), SVM with quadratic kernel (94.4%) and AdaBoost (95.8%), and obtained good results. Gene selection with TNoM scoring improved the performance of the AdaBoost and the other classifiers. Dudoit *et al.* [7] used BSS/WSS term (the ratio of between-groups to within-groups sum of squares of gene expressions) to measure feature relevancy, and nearest-neighbor diagonal linear discriminant analysis and BoostCART were used as classifiers. Using 40 genes on the basis of BSS/WSS, the recognition accuracies of classifiers were above 95.0%. Nguyen *et al.* [8] also worked out this problem, using principal component analysis and partial least square to extract features and linear discriminant and quadratic discriminant analysis to classify the cancer samples, obtaining 82.4%$\sim$97.1% of recognition accuracy.

So far, most published papers on cancer classification have applied a single conventional technique, yet we still

need a more sophisticated technique for this problem. In our previous work [9], in order to assess the merits of several features and classifiers that are for the classification of leukemias based on gene expression data, we have compared the performance of seven feature selection methods and five classifiers that are well known in the field of data mining and pattern recognition. It turns out that some of the feature selection methods are correlated, and several classifiers also make the same mistakes on the particular samples. In this paper, we take it a step further to propose an efficient classification framework to enhance the classification performance by combining multiple classifiers with mutually exclusive features.

The idea behind a classifier ensemble with mutually exclusive features is to encourage a pair of classifiers to learn different aspects of training data, so that the ensemble of classifiers can search in a wide solution space. In order to classify the gene expression profile, we suggest a classifier ensemble composed of multiple classifiers and show the usefulness of mutually exclusive features.

The paper is organized as follows. Section II contains a brief introduction to the technology of DNA microarrays and the related works for multiple classifiers. The classifier ensemble with mutually exclusive features is described in Section III, and thorough experimental results with the leukemia dataset of Golub *et al.* [1] are presented in Section IV.

II. Backgrounds

A. DNA 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 the DNA spot is less than 250 μm, and macroarrays, when the diameter is bigger than 300 μm. When the solid substrate is small, arrays are also referred to as DNA chips.

DNA microarrays contain thousands of individual DNA sequences printed in a high-density array on a glass microscope slide using a robotic arrayer (see Fig. 1). 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, shown as follows [10]–[12]:

$$gene\_expression = \log_2 \frac{\mathrm{Int(Cy5)}}{\mathrm{Int(Cy3)}} \qquad (1)$$

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, it is helpful so that we can investigate the genome-wide information in short time.

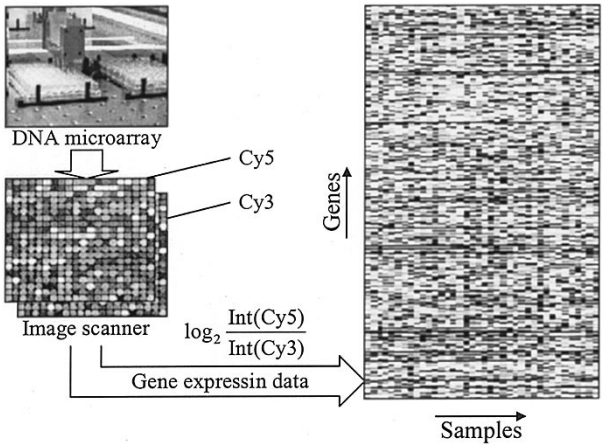

Fig. 1. Process of acquiring the gene expression data from a DNA microarray. Gene expression data are the log ratio of the intensities of cell spots acquired in Cy5 and Cy3 channels.

B. Related Works

Many researchers have been working on the ensemble of modular neural networks. According to the Osherson's definition, a neural network is said to be modular if and only if the following statement is satisfied [13]:

> "The computation performed by the network can be decomposed into two or more subsystems that operate on distinct inputs without communicating with each other."

After the notion of modular connectionist systems was first discussed in the mid-1980s by Barto and Hinton, Pollack proposed the cascaded backpropagation architecture [14] and Jacobs developed taxonomy for a class of modular hierarchical connectionist models (hierarchical mixture-of-experts, HME) [15]. Hampshire and Waibel proposed the Meta-Pi, which consists of a number of source-dependent subnetworks that are integrated by a combinational time-delay neural network [16]. Lincoln and Skrzypek proposed clustering multiple backpropagation networks [17]. Battiti and Colla suggested the concept of democracy to combine the outputs of different neural network classifiers [18]. These early examples have shown that integrating the multiple modules, often referred to as committee machines, could have enhanced the accuracy and generalization capacity.

Gutta *et al.* used multiple features of face images to train the ensemble of expert classifiers, in order to classify the gender, ethnic origin, and pose of face from the FacE REcognition Test (FERET) image database [19]. The proposed classifier, the hybrid of radial basis function (RBF) networks, gets Gaussian noise and 5° of geometric transformation put on the original image as the input features, as well as the original, and produced 93.3% correct of the gender classification over the 60 test sets.

Meanwhile, Liu *et al.* studied evolutionary learning of neural network classifiers with a negative correlation of classifiers [20], [21]. They used a penalty term in error function based on correlations between neural networks, so that the classifiers learn to be negatively correlated.

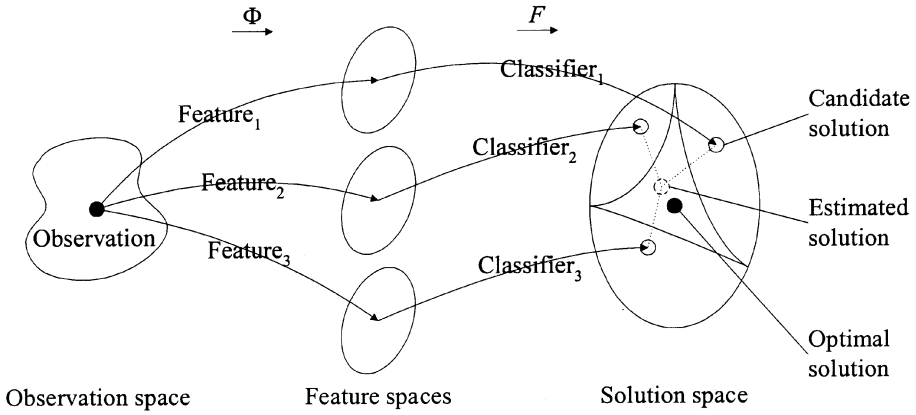

Fig. 2. Motivation of the use of multiple features for a classification problem. Φ and F are the nonlinear transformation functions for the feature and solution spaces, respectively. Using multiple features is helpful to explore wider solution space and produces the estimated solution that is closer to the optimal solution than the individual candidate solutions.

They have also shown that they can define the independent learning, often used in modular networks, as one of negative learning using the lambda term.

In this paper, multiple features are utilized to improve the classification accuracy of the classifier ensemble, focusing on the mutually exclusive features.

III. Classifier Ensemble

Fig. 2 illustrates the basic idea of multiple classifiers with multiple features. 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, it may be more effective for the classifier to use as many features as possible for the problems. However, many features may cause the redundancy of irrelevant information and result in the counter effect like over-fitting. Suppose $\Phi = \{\varphi_1, \varphi_2, \ldots, \varphi_m\}$ is a set of nonlinear functions of transforming from the observation space to the feature space. When $\Phi_k \in 2^{\Phi}$, the categorical information, $I(\Phi_k)$, that Φ_k holds can be expressed as

$$I(\Phi_k) = \frac{a \sum A_i}{\frac{N}{2} \sum_{j=1,\, j \neq i}^{N} d_{ij}} + b \tag{2}$$

where N is the number of elements of Φ_k, d_{ij} is the dependency between the ith and jth elements of Φ_k, A_i is the area of feature space covered by the ith element of Φ_k, and a and b are constants. Therefore, it is more important to explore and utilize independent (or mutually exclusive) features so that the classifiers have a greater chance to get categorical information when learning, rather than increase the number of features we use. Correlation between feature sets can be induced from the distribution of features or from the mathematical analysis using statistics.

On the other hand, there exist many classification algorithms from the machine learning approach, but none of them is perfect, and, according to the environments in which the classifier is embedded, some algorithms work well and others do not. This is because the classifier searches in different solution space by the algorithms, parameters of algorithms, and features used. Once they work together, however, a set of classifiers can explore a wider solution space.

This configuration of classifiers (classifier ensemble) is different from that of heirarchical mixture of experts (HME) in that an element classifier of ensemble uses holistic features whereas HME uses partial information (subset of input space) when learning. Ensemble classifiers, therefore, are not modular, but they are still experts in their feature subsets.

We have applied this idea to a classification framework as shown in Fig. 3. If there are k features, we choose the most mutually exclusive features through the correlation analysis between ${}_kC_2$ possible combinations of features. Classifiers are trained using the features selected, and a combining module is accompanied to combine the outputs of these classifiers.

A. Feature Selection

For efficient classification, we need to find out the informative features from input observation. This can be done by statistical tools, similarity measures, or information-theoretical methods. These are methods to score how informative each feature is and contain categorical information, and we finally choose k features from the top. There are three different approaches as follows.

Suppose that we have a $M \times N$ training set where M is the number of samples (input vector) and N is the number of features (dimensionality of input vector). The ith feature of samples, g_i, can be expressed as

$$g_i = (e_1, e_2, e_3, \ldots, e_M) \tag{3}$$

where e is the data and $i = 1, 2, \ldots, N$. We want to know the locations of informative k features out of N. If it is pos-

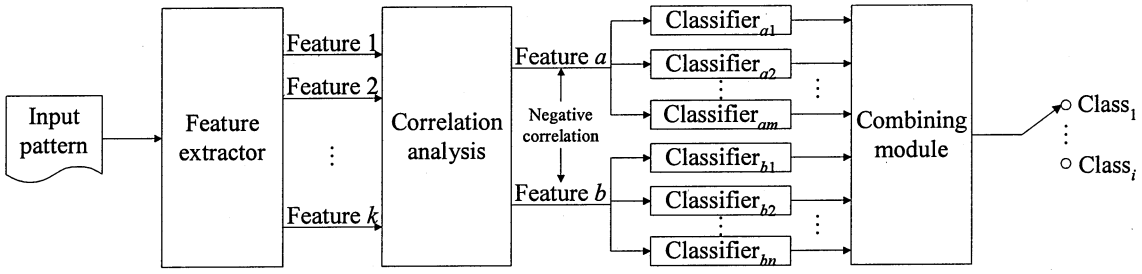

Fig. 3. Schematic diagram of a classifier ensemble with mutually exclusive features. After features a and b, which are most mutually exclusive, are chosen out of k possible ones, classifiers trained independently produce their own outputs. The final decision is judged by the combining module, where the neural network or voting method is adopted.

sible to know the representative vector g_{ideal} for class c_j, we can simply measure the correlation and similarity of g_i to classes, which tells the feature goodness. Modeling g_{ideal}, we should use prior knowledge and intuitional experience about classes. Suppose g_{ideal} is an ideal vector representing class c_j

$$g_{\text{ideal}} = (e'_1, e'_2, e'_3, \ldots, e'_M). \tag{4}$$

Cross analysis, such as Pearson (*PR*) and Spearman (*SP*) correlation coefficients, can measure the correlation between g_i and g_{ideal}. The coefficient can range from -1 to $+1$: a higher coefficient implies that g_i is more correlated to c_j.

The similarity between two variables can be thought of as the distance of those. Distance is a measure on how far the two vectors are located, and the distance between g_i and g_{ideal} implies how much g_i is likely to the class c_j. In this paper, we have adopted Euclidean distance (*ED*) and cosine coefficient (*CC*).

However, it is not always possible to know the ideal vectors for classes. Then, we have alternative ways to measure the relevance of feature using the frequency of g_i satisfying condition Q under the categorical situations. Information gain (*IG*) and mutual information (*MI*) are good examples. [For (9) and (10), $P(g_i)$ is the number of g_i satisfying Q, $P(\overline{g}_i)$ is the number of features satisfying Q other than g_i, and $P(g_i, c_j)$ is the number of cases when g_is satisfying Q and c_j occur simultaneously.]

Moreover, the SNR (*SN*) measures features from the information of the distribution of features in each class. g_i is composed of two parts: one from c_j and the other from $\overline{c}_j$. When we calculate the mean μ and standard deviation σ from the distribution of gene expressions within their classes, the SNR of gene g_i is defined as (13). Table 1 summarizes the mathematical formula for feature selection methods.

From comparative experiments with feature selection methods, we get a set of informative features from the data. In order to choose mutually exclusive features, we have plotted the distribution of g_i from pairs of feature selection methods. If the two features are mutually exclusive, the distribution will be in the $(-)$ direction, otherwise the $(+)$ direction.

B. Classification

Many promising machine learning algorithms have successfully been applied to classification problems. We have used multiplayer perceptron (MLP), SVM, and k-nearest neighbor (KNN) as classifiers. Each classifier has been trained independently with the feature selection methods that are mutually exclusive.

MLP is commonly used in such fields as pattern recognition due to its powerful and stable learning algorithms [22]. The power of the backpropagation algorithm on MLP 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 [23].

The SVM introduced by Vapnic in 1995 is a method to estimate the function classifying the data into two classes [24], [25]. The basic idea of SVM is to construct a hyperplane as the decision surface in such a way that the margin of separation between positive and negative examples is maximized. SVM achieves this by the structural risk minimization principle that is based on the fact 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 (X_i, y_i), where $X_i \in R^N$ and y_i is the associated label $y_i \in \{-1, 1\}$, the discriminant hyperplane is defined by

$$f(X) = \sum_{i=1}^{M} y_i \alpha_i k(X, X_i) + b \tag{5}$$

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 nonzeros. RBF (SVM_{RBF}) and Linear ($\text{SVM}_{\text{linear}}$) kernels have also been used.

KNN extracts the k closest vectors in the reference set based on similarity measures and makes the decision for the label of the input vector using the information of distribution and labels of k neighbors. Pearson correlation has been used for 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:

$$P(X, c_j) = \sum_{d_i \in kNN} \text{Sim}(X, d_i) P(d_i, c_j) - b_j \tag{6}$$

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

Table 1
Mathematical Formula for Feature Selection Methods. Each Method Tells How Much g_i is Informative for the Class. The Top k Genes Are Chosen According to These Numbers Calculated by Following Equations

Feature selection methods	
$PR(g_i, g_{ideal}) = \frac{\sum g_i g_{ideal} - \frac{\sum g_i \sum g_{ideal}}{N}}{\sqrt{\left(\sum g_i^2 - \frac{(\sum g_i)^2}{N}\right)\left(\sum g_{ideal}^2 - \frac{(\sum g_{ideal})^2}{N}\right)}}$	(7)
$SP(g_i, g_{ideal}) = 1 - \frac{6\sum (D_g - D_{ideal})^2}{N(N^2-1)}$, ($D_g$ and D_{ideal} are the rank matrices of g_i and g_{ideal})	(8)
$ED(g_i, g_{ideal}) = \sqrt{\sum (g_i - g_{ideal})^2}$	(9)
$CC(g_i, g_{ideal}) = \frac{\sum g_i g_{ideal}}{\sqrt{\sum g_i^2 \sum g_{ideal}^2}}$	(10)
$IG(g_i, c_j) = P(g_i, c_j) \log \frac{P(g_i, c_j)}{P(c_j) \cdot P(g_i)} + P(\bar{g}_i, c_i) \log \frac{P(\bar{g}_i, c_j)}{P(c_j) \cdot P(\bar{g}_i)}$	(11)
$MI(g_i, c_j) = \log \frac{P(g_i, c_j)}{P(g_i) \cdot P(c_j)}$	(12)
$SN(g_i, c_j) = \frac{\mu_{c_j}(g_i) - \mu_{\bar{c}_j}(g_i)}{\sigma_{c_j}(g_i) - \sigma_{\bar{c}_j}(g_i)}$	(13)

C. Classifier Combination

A neural network combines the outputs of multiple classifiers in our system. Outputs of individual classifiers can be thought of as classification status values (CSVs), which contain information on answer patterns of classifiers. A neural network has $m+n$ (dimensionality of CSV) input nodes and j output nodes. Using neural network, we can have the adaptivity of thresholds based on the entropy as opposed to *ad hoc* and hard thresholds [26]. For the comparison, majority voting has been also used [18].

IV. Experiments

A. Data Set

Dataset that we have used is a collection of expression measurements reported by Golub *et al.* [1]. Gene expression profiles have been constructed from 72 people who have either acute lymphoblastic leukemia (ALL) or acute myeloid leukemia (AML). Each person has submitted one sample of DNA microarray, so that the database consists of 72 samples. Each sample is composed of 7129 gene expressions, and finally the whole database is a 7129×72 matrix.

Thirty-eight samples are for training and the other thirty-four are for test of the classification. The training data has 27 ALL and 11 AML samples, whereas the test data has 20 ALL and 14 AML samples.

B. Gene Selection and Classification

Before the classification, we need to find out informative genes (feature) that are related to predict the cancer class out of 7129. In order to do this, each gene is scored based on the equations in Table 1, and the 25 top-ranked genes are chosen as the feature of the input pattern. A gene in the training set is expressed as: $g_i = (e_1, e_2, e_3, \ldots, e_{38})$, since there exist 38 samples. Suppose that the first 27 numbers $(e_1, e_2, \ldots, e_{27})$ are from samples of ALL, and the other 11 $(e_{28}, e_{29}, \ldots, e_{38})$ are from AML.

We define an ideal gene pattern which belongs to the ALL class, called $g_{\text{ideal\_ALL}} = (1, 1, \ldots, 1, 0, \ldots, 0)$, so that all numbers from the ALL samples are 1, and the others are 0. We have measured the correlation coefficient between g_{ideal} and each gene expression pattern, when $N = 7129$, which is the dimension of gene pattern.

For *IG* and *MI*, the condition Q is "if it is induced." The gene expression has positive values when the gene is more induced than the background condition and negative values when repressed. Therefore, we simply count the number of positive and negative gene expressions to get $P(g_i)$ and $P(\bar{g}_i)$.

For the classifiers, we have used three-layered MLP with 5∼15 hidden nodes, 2 output nodes, a learning rate of 0.03∼0.5, and a momentum of 0.1∼0.9. $\gamma = 0.5$ is used for the RBF kernel of SVM and KNN uses $k = 3$.

Additionally, a self-organizing map (SOM), decision tree (DT) and KNN with cosine coefficient similarity measure ($\text{KNN}_{\text{cosine}}$) have been used for the comparison. SOM uses 2×2∼5×5 maps with rectangular topology and a learning rate of 0.05. Quinlan's C4.5 has been adopted for DT. The parameters that produce the best results on the training set have been chosen. The final results are averaged by ten different runs.

C. Results

1) Single Classifier: As the result of feature selection, in total 175 (25×7) genes, with overlap, are selected. Only three genes, g_{1882}, g_{2288}, and g_{6201}, appear in three feature

Table 2
Genes Chosen by More Than Two Methods. g_{1882}, g_{2288}, and g_{6201} are Selected Three Times by the Feature Selection Methods and Remain Twice

ID #	Gene accession number	Gene description
2	AFFX-BioB-M_at	AFFX-BioB-M_at (endogenous control)
5	AFFX-BioC-3_at	AFFX-BioC-3_at (endogenous control)
6	AFFX-BioDn-5_at	AFFX-BioDn-5_at (endogenous control)
8	AFFX-CreX-5_at	AFFX-CreX-5_at (endogenous control)
12	AFFX-BioB-3_st	AFFX-BioB-3_st (endogenous control)
13	AFFX-BioC-5_st	AFFX-BioC-5_st (endogenous control)
14	AFFX-BioC-3_st	AFFX-BioC-3_st (endogenous control)
18	AFFX-CreX-3_st	AFFX-CreX-3_st (endogenous control)
22	AFFX-DapX-3_at	AFFX-DapX-3_at (endogenous control)
461	D49950_at	Liver mRNA for interferon-gamma inducing factor (IGIF)
1249	L08246_at	INDUCED MYELOID LEUKEMIA CELL DIFFERENTIATION PROTEIN MCL1
1745	M16038_at	LYN V-yes-1 Yamaguchi sarcoma viral related oncogene homolog
1834	M23197_at	CD33 CD33 antigen (differentiation antigen)
1882	M27891_at	CST3 Cystatin C (amyloid angiopathy and cerebral hemorrhage)
2020	M55150_at	FAH Fumarylacetoacetate
2043	M57710_at	LGALS3 Lectin, galactoside-binding, soluble, 3 (galectin 3) (NOTE: redefinition of symbol)
2111	M62762_at	ATP6C Vacuolar H+ ATPase proton channel subunit
2242	M80254_at	PEPTIDYL-PROLYL CIS-TRANS ISOMERASE, MITOCHONDRIAL PRECURSOR
2288	M84526_at	DF D component of complement (adipsin)
2402	M96326_rna1_at	Azurocidin gene
2759	U12471_cds1_at	Thrombospondin-p50 gene extracted from Human thrombospondin-1 gene, partial cds
3258	U46751_at	Phosphotyrosine independent ligand p62 for the Lck SH2 domain mRNA
3320	U50136_rna1_at	Leukotriene C4 synthase (LTC4S) gene
3847	U82759_at	GB DEF = Homeodomain protein HoxA9 mRNA
4196	X17042_at	PRG1 Proteoglycan 1, secretory granule
4847	X95735_at	Zyxin
5039	Y12670_at	LEPR Leptin receptor
6200	M28130_rna1_s_at	Interleukin 8 (IL8) gene
6201	Y00787_s_at	INTERLEUKIN-8 PRECURSOR
6990	U21689_at	SAT Spermidine/spermine N1-acetyltransferase

selection methods at the same time, which is very informative. However, only 30 genes appear in more than two feature selection methods as shown in Table 2. This fact indicates that the gene expression profile database has relatively disjoint feature spaces, produced by seven feature selection methods illustrated in Section III-A. The classifiers trained on these feature subspaces would search for the optimal solution exclusively in the solution space.

Table 3 shows the examples of frequent misclassification made by MLP, SVM_{RBF}, $\text{SVM}_{\text{linear}}$, and KNN, which are single classifiers to be combined in ensemble. As can be seen, sample #66 was always missed by every feature selection method, as other papers addressed this problem [5], [8], except Euclidean distance. On the other hand, the Euclidean distance was successful in classifying sample #66, but always failed for sample #60. This implies that the Euclidean distance provides the classifiers with different categorical information.

The results of recognition rate on the test data with a single feature and classifier are as shown in Table 4. Simply using 25 features, the MLP seems to have the best recognition rate among the classifiers on the average. A proportion correction of 97.1% is the best throughout all the classifiers and features. However, the performance of most classifiers seems to depend somewhat on the features they use. SVM_{RBF}, for example, has a recognition rate of 97.1% with Pearson's correlation, but 58.8% with information gain and mutual information, which is the worst. DT with Pearson correlation produces a recognition rate of 97.1% (one of the best cases in our experiments), but DT with information gain does not work for the classification at all (47.1%, the worst case). The mean and standard deviation of each classifier are shown in Fig. 4.

The SNR and Pearson's correlation are the best among the feature selection methods, resulting in 94.1% and 90.0% on the average, respectively.

Table 3
Sample IDs That Are Misclassified More Than Five Times During Ten Runs

Feature	MLP	KNN	SVM_{RBF}	SVM_{linear}
PR	66	64 66	66	52 58 61 62 64 65 66
SP	53 54 58 60 61 62 63 65 66 67	53 54 58 60 61 62 63 65 67 70	53 54 58 60 61 62 63 65 66 67	53 54 58 60 61 62 63 65 66 67
ED	60	52 58 60 62 63	52 58 60	52 57 58 60
CC	43 49 54 60 62 63 64 66	49 50 52 54 60 61 62 64 66	49 52 54 57 60 61 62 63 64 66	50 51 52 53 54 57 58 60 61 62 63 64 65 66
IG	50 51 52 53 54 57 58 60 61 62 63 64 65 66	40 51 54 57 58 60 61 62 64 65 66 67 71	50 51 52 53 54 57 58 60 61 62 63 64 65 66	50 51 52 53 54 57 58 60 61 62 63 64 65 66
MI	42 43 47 50 54 56 57 60 62 63 64 69	42 43 47 49 50 52 55 60 62 63 64 66 70 72	50 51 52 53 54 57 58 60 61 62 63 64 65 66	50 51 52 53 54 57 58 60 61 62 63 64 65 66
SN	66 67	66 67	66 67	66 67

Table 4
Recognition Rates by Feature and Classifier [%]. MLP and SNR Produce the Best Recognition Rates on the Average

Feature / Classifier	Pearson	Spearman	Euclidean distance	Cosine coefficient	Information gain	Mutual information	S/N ratio	Average
MLP	97.1	70.6	97.1	79.4	72.9	62.1	94.1	81.9
KNN	94.1	70.6	85.3	73.5	67.8	58.8	94.1	77.7
SVM_{RBF}	97.1	70.6	91.2	70.6	58.8	58.8	94.1	77.3
SVM_{linear}	79.4	70.6	88.2	58.8	58.8	58.8	94.1	72.7
KNN_{cosine}	91.2	67.6	82.4	73.5	67.6	55.9	94.1	76.0
SOM	74.1	67.4	70.6	70.6	63.8	68.8	97.1	73.2
DT	97.1	61.8	82.4	73.5	47.1	55.9	91.2	72.7
Average	90.0	68.5	85.3	71.4	62.4	59.9	94.1	75.9

2) Classifier Ensemble: Fig. 5 is the scatter plot that illustrates the dependency between pairs of feature selection methods. These are three representative cases of all 21 possible feature pairs ($_7C_2$). Each axis is the feature selection method, and 7129 genes' scores by the corresponding feature selection methods are marked as dark dots in the figure.

Case (a) is the correlation between Pearson's correlation and Euclidean distance. Dots are distributed in negative direction. Genes ranked high in Pearson's correlation get low scores from the Euclidean distance, and vice versa. Therefore, the feature sets chosen by Pearson's correlation and Euclidean distance must be very disjoint, and the classifiers with these feature selection methods are trained in less dependent feature spaces.

Case (b) is the correlation between Pearson's correlation and SNR methods. Dots are distributed in a triangular form. We cannot observe the radical direction of the correlation. However, when we see around the part of the right vertex of the triangle, it seems to have a tendency that genes chosen by a feature selection method are devaluated by another, just as in case (a), leading to neutral correlation. Actually, most of the cases in the experiments show neutral correlations, and (b) is one typical example of this category.

Case (c) shows a positive correlation between Pearson's correlation and cosine coefficient methods. Genes selected by one method also appear in the list of top-ranking genes by another method. In this case, there must be many common genes between two feature sets, so that the ensemble classifiers will learn from highly correlated feature sets. Since two sets of classifiers are trained in mutually dependent feature spaces, it is hard to expect the performance improvement when the classifiers are combined by the ensemble method.

MLP, KNN, SVM_{linear}, and SVM_{RBF} have been trained simultaneously from the same feature sets chosen by each feature selection method, and the outputs from this set of classifiers of two features [cases (a)–(c)] are combined by a

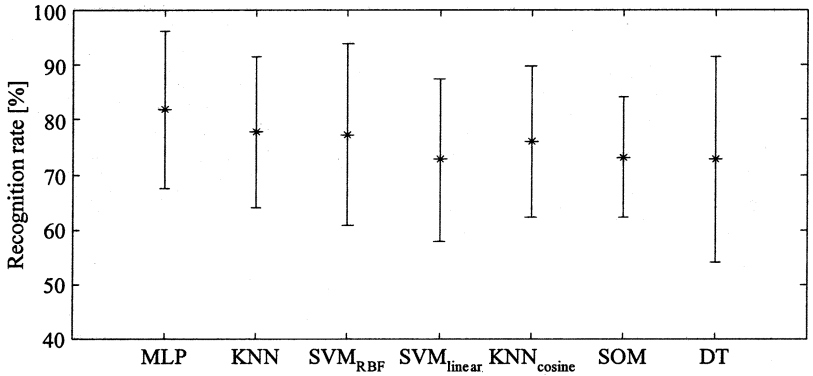

Fig. 4. Error bars of single classifiers. Standard deviations of single classifiers by the feature does not vary much; the lowest is 11.0 (SOM) and the biggest is 18.7 (DT). DT is relatively sensitive to the feature used when trained even though it produces a high recognition rate.

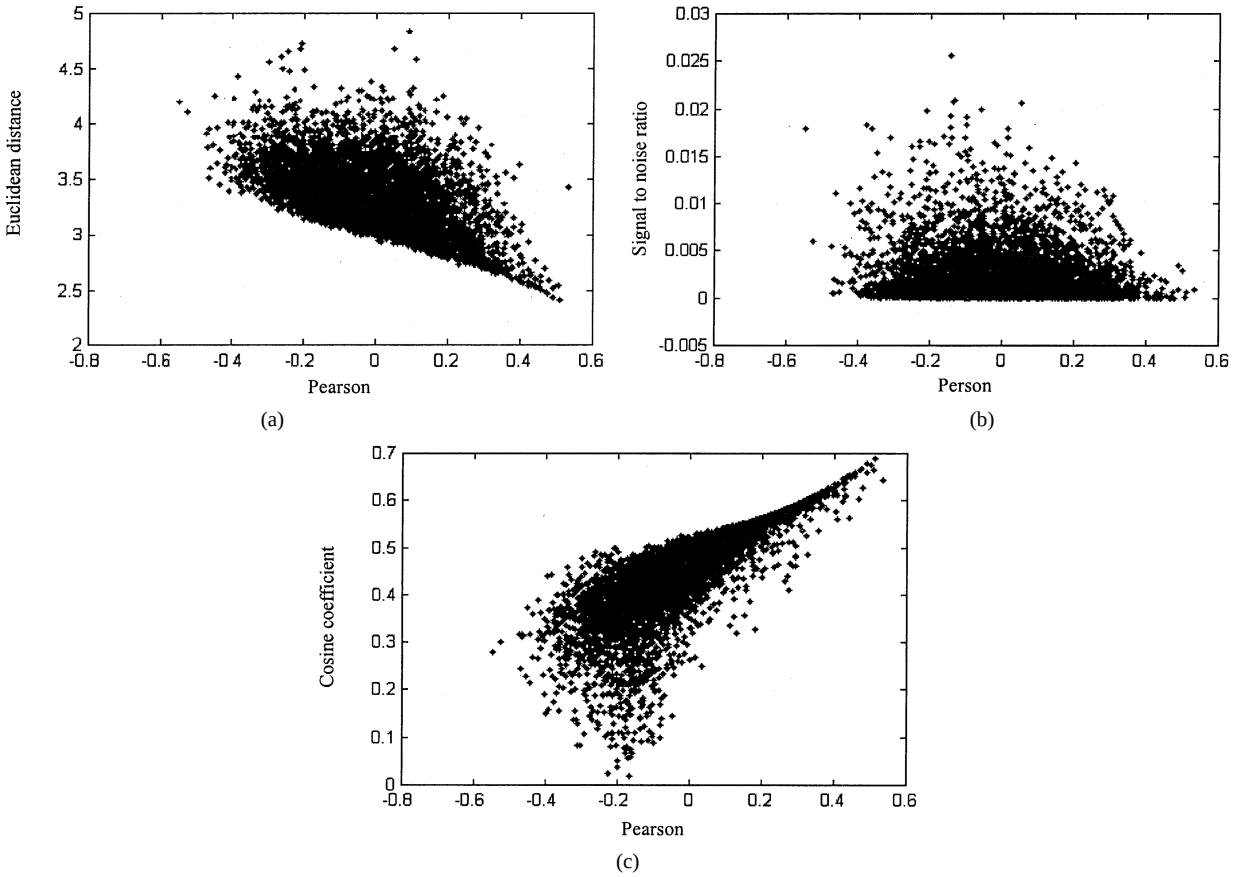

Fig. 5. Correlations of feature selection methods. Negative correlation implies that genes selected by a feature selection method are not apt to be selected by another method, which provides disjoint feature spaces. (a) Negative correlation. (b) Neutral. (c) Positive correlation.

neural network and majority voting. For the comparison, we have also combined the outputs of classifiers trained using all features.

Fig. 6 is the result of the ensemble classifiers. Cases (a)–(c) are investigated and we have also combined all the features for the comparison. Case (a) produces the best recognition rates (100% with a neural network and 97.1% with the voting method). Case (b) also produces relatively high rates. Case (c) and "all feature," however, turn out to be bad, which implies that combining mutually exclusive features is efficient, producing much higher performance than the case when all features are considered. This clearly shows that the suggested framework works and we can obtain better classification performance by combining mutually exclusive sets of classifiers learned from a pair of less dependent features, even when we use a simple combination with majority voting.

D. Discussion

It is well known that distinguishing acute leukemias is one of the challenging problems in bioinformatics area be-

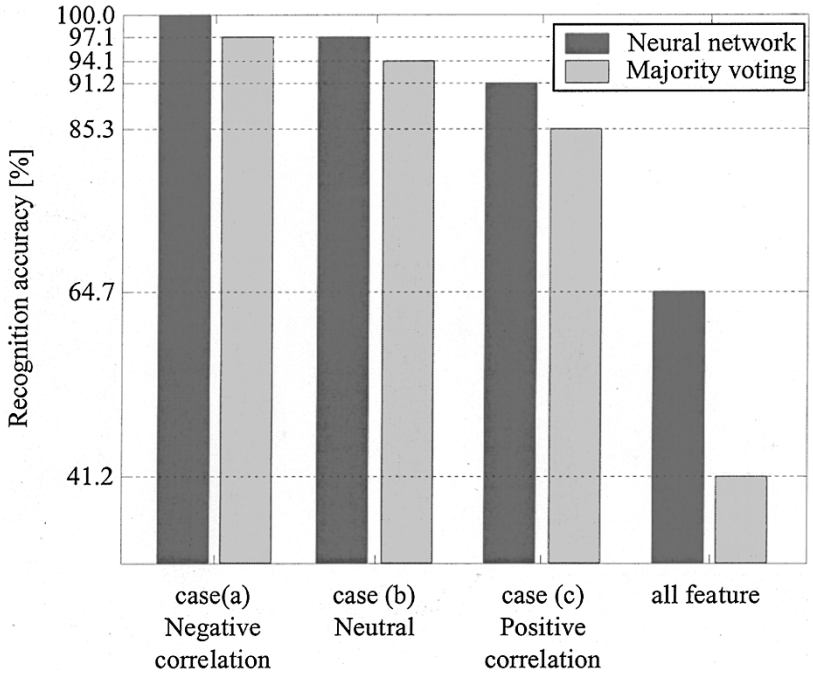

Fig. 6. Results of ensemble classifiers. Case (a) produces high recognition rates in both combining schemes, outperforming cases (b) and (c) and the one when all features are used. This shows that the ensemble classifier trained in mutually exclusive feature spaces is the most efficient.

Table 5
Relevant Works on the Leukemia Dataset

Authors	Method		Accuracy [%]
	Feature	Classifier	
Furey *et al.*	Signal to noise ratio	SVM	75.6
Li *et al.*	Model selection with Akaike information criterion and Bayesian information criterion with logistic regression		94.1
Ben-Dor *et al.*	All genes, TNoM score	Nearest neighbor	91.6
		SVM with quadratic kernel	94.4
		AdaBoost	95.8
Dudoit *et al.*	The ratio of between-groups to within-groups sum of squares	Nearest neighbor	95.0~
		Diagonal linear discriminant analysis	95.0~
		BoostCART	95.0~
Nguyen *et al.*	Principal component analysis	Logistic discriminant	97.1
		Quadratic discriminant analysis	82.4
	Partial least square	Logistic discriminant	97.1
		Quadratic discriminant analysis	88.2

cause the appearance is highly similar, and there have been many papers published with different features and classification methods as shown in Table 5. As can be seen, the performance varies significantly depending on the features and classifiers used, and our classifier is the best. However, it is hard to assess the merit of the classifier ensemble in the absence of a comprehensive comparative study with different datasets.

Some of the well-known datasets include lymphoma data[1] and colon data. Lymphoma data contain 4026 genes across 47 samples, of which 24 are referred to as germinal center B-like DLBCL and 23 as activated B-like DLBCL. The original colon data contain the expression levels of 2000 genes across 62 samples, of which 40 are tumor tissue and

[1]Available. [Online.] http://llmpp.nih.gov/lymphoma/

22 normal tissue [27]. In order to assess the real value of the proposed, we will be conducting the same experiments with those datasets.

V. Concluding Remarks

In order to predict the cancer class of patients, we have illustrated a classification framework that combines a pair of classifiers using the correlation information of seven feature selection methods. We have shown the usefulness of this framework with leukemia gene expression data. Experimental results show that the feature sets that are mutually exclusive yield high recognition result.

References

[1] T. R. Golub, "Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, pp. 531–537, 1999.

[2] P. Tamayo, "Interpreting patterns of gene expression with self-organizing map: Methods and application to hematopoietic differentiation," in *Proc. Nat. Acad. Sci. USA*, vol. 96, 1999, pp. 2907–2912.

[3] M. West, J. R. Nevins, J. R. Marks, R. Spang, C. Blanchette, and H. Zuzan, "DNA microarray data analysis and regression modeling for genetic expression profiling," Institute of Statistics and Decision Sciences, Duke University, Chapel Hill, NC, Tech. Rep. 00-15, 2001.

[4] 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*, vol. 16, no. 10, pp. 906–914, 2000.

[5] W. Li and Y. Yang, "How many genes are needed for a discriminant microarray data analysis," in *Proc. Critical Assessment of Techniques for Microarray Data Mining Workshop*, Dec. 2000.

[6] A. Ben-Dor, L. Bruhn, N. Friedman, I. Nachman, M. Schummer, and N. Yakhini, "Tissue classification with gene expression profiles," *J. Comput. Biol.*, vol. 7, pp. 559–584, 2000.

[7] S. Dudoit, J. Fridlyand, and T. P. Speed, "Comparison of discrimination methods for the classification of tumors using gene expression data," Department of Statistics, University of California, Berkeley, Tech. Rep. 576, June 2000.

[8] D. V. Nguyen and D. M. Rocke, "Tumor classification by partial least squares using microarray gene expression data," *Bioinformatics*, 2002, to be published.

[9] J. Ryu and S.-B. Cho, "Toward optimal feature and classifier for gene expression classification of cancer," *Lecture Notes in Artificial Intelligence*, vol. 2275, pp. 310–317, 2002.

[10] D. Lashkari, J. Derisi, J. McCusker, A. Namath, C. Gentile, S. Hwang, P. Brown, and R. Davis, "Yeast microarrays for genome wide parallel genetic and gene expression analysis," in *Proc. Nat. Acad. Sci. USA*, vol. 94, 1997, pp. 13 057–13 062.

[11] J. Derisi, V. Iyer, and P. Brosn, "Exploring the metabolic and genetic control of gene expression on a genomic scale," *Science*, vol. 278, pp. 680–686, 1997.

[12] M. Eisen, P. Spellman, P. Brown, and D. Bostein, "Cluster analysis and display of genome-wide expression patterns," in *Proc. Nat. Acad. Sci. USA*, vol. 95, 1998, pp. 14 863–14 868.

[13] D. N. Osherson, S. Weinstein, and M. Stob, "Modular learning," in *Computational Neuroscience*, E. L. Schwartz, Ed. Cambridge, MA: MIT Press, 1990, pp. 369–377.

[14] J. Pollack, "Cascaded back-propagation on dynamic connectionist networks," in *Proc. Ninth Ann. Conf. Cognitive Sci. Soc.*, 1987, pp. 391–404.

[15] R. Jacobs, "Initial experiments on constructing domains of expertise and hierarchies in connectionist systems," in *Proc. Connectionist Models Summer School*, San Mateo, CA, 1988, pp. 144–153.

[16] J. B. Hampshire and A. Waibel, "The Meta-Pi network: Building distributed knowledge representations for robust multisource pattern recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, pp. 751–769, July 1992.

[17] W. P. Lincoln and J. Skrzypek, "Synergy of clustering multiple backpropagation networks," in *Advances in Neural Information Processing Systems*, D. S. Touretzky, Ed. San Mateo, CA: Morgan Kauffmann, 1990, vol. 1, pp. 650–657.

[18] R. Battiti and A. M. Colla, "Democracy in neural nets: Voting schemes for classification," *Neural Networks*, vol. 7, pp. 691–707, July 1994.

[19] S. Gutta, R. J. Huang, P. Jonathon, and H. Wechsler, "Mixture of experts for classification for gender, ethnic origin, and pose of human faces," *IEEE Trans. Neural Networks*, vol. 11, pp. 948–960, July 2000.

[20] Y. Liu and X. Yao, "Ensemble learning via negative correlation," *Neural Networks*, vol. 12, pp. 1399–1404, Dec. 1999.

[21] ——, "Evolutionary ensembles with negative correlation learning," *IEEE Trans. Evol. Comput.*, vol. 4, pp. 1399–1404, Nov. 2000.

[22] R. P. Lippman, "An introduction to computing with neural nets," *IEEE Acoust., Speech, Signal Processing Mag.*, pp. 4–22, Apr. 1987.

[23] H. D. Beale, *Neural Network Design*: PWS, 1996, vol. 11, pp. 1–47.

[24] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer, 1995.

[25] B. Moghaddam and M.-H. Yang, "Gender classification with support vector machines," in *Proc. 4th IEEE Int. Conf. Automatic Face and Gesture Recognition*, 2000, pp. 306–311.

[26] J. Ryu and S.-B. Cho, "Gender recognition of human behaviors using neural network ensembles," in *Proc. IEEE-INNS Int. Joint Conf. on Neural Networks*, Washington, DC, July 2001, pp. 571–576.

[27] U. Alon *et al.*, "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," in *Proc. Nat. Acad. Sci. USA*, vol. 96, 1999, pp. 6745–6750.

Sung-Bae Cho (Member, IEEE) 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 Korea Advanced Institute of Science and Technology (KAIST), 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 Advanced Telecommunications Research (ATR) 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 the International Conference on Soft Computing in 1996 and 1998. Also, he received the best paper award at the World Automation Congress in 1998 and is 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.

Jungwon Ryu received the B.S. degree in information and communication engineering from Sunmoon University, Asan, Korea, in 2000, and the M.S. degree in computer science from Yonsei University, Seoul, Korea, in 2002.

His current research interests include pattern recognition, data mining, information retrieval, bioinformatics, text categorization, biological motion perception, and committee machines.