A Faster cDNA Microarray Gene Expression Data Classifier for Diagnosing Diseases

Sun-Yuan Hsieh and Yu-Chun Chou

Abstract—Profiling cancer molecules has several advantages; however, using microarray technology in routine clinical diagnostics is challenging for physicians. The classification of microarray data has two main limitations: 1) the data set is unreliable for building classifiers; and 2) the classifiers exhibit poor performance. Current microarray classification algorithms typically yield a high rate of false-positives cases, which is unacceptable in diagnostic applications. Numerous algorithms have been developed to detect false-positive cases; however, they require a considerable computation time. To address this problem, this study enhanced a previously proposed gene expression graph (GEG)-based classifier to shorten the computation time. The modified classifier filters genes by using an edge weight to determine their significance, thereby facilitating accurate comparison and classification. This study experimentally compared the proposed classifier with a GEG-based classifier by using real data and benchmark tests. The results show that the proposed classifier is faster at detecting false-positives.

Index Terms—Bioinformatics, classification, gene-expression graph (GEG), GEG-based classifier, microarray

1 Introduction

Since the 1980s, microarray technology has evolved into genetic research. Recent advances in microarray technology generated a great deal of gene expression data. Numerous approaches for selecting and clustering gene have been proposed to analyze large-scale gene expression data [6], [17], [36], [39]. A DNA microarray is a small solid support, such as a membrane or glass slide, upon which DNA sequences are fixed in an orderly manner. A microarray can hold tens of thousands of DNA probes attached to a single slide, and the DNA probes can be used to analyze gene expression activity. Thousands of genes can be analyzed in a single experiment, thereby facilitating the identification of the complex relationships that exist among them. This technology has been used to achieve several novel goals in bioinformatics.

Scientists have used DNA microarrays to achieve the following two crucial goals of functional genomics: 1) the classification of diseases at the molecular level, which is achieved by measuring the difference between various gene expressions; and 2) the identification of genes responding to specific cellular phenotypes (e.g., a disease or a particular treatment), thereby identifying their activity as a model for differentiating between normal and abnormal behaviors. Although microarrays are a critical source of biological information, their application in clinical diagnostics remains challenging for classifying diseases by using gene expression at the molecular level. Various experiments can be conducted to evaluate gene expression levels and to identify relevant genes. The biological information of a target phenomenon can be acquired by applying accurate and readily interpretable classification rules [9].

Classifying microarray data by using a typical machine learning method is difficult. Microarray classification incurs the "small N, large P" problem of statistical learning, which occurs when the number of available samples is less than the number of variables (genes). Most experimental studies have used an insufficient number of samples to obtain statistically significant results [8], [33]. The given phenotype classification problem is irrelevant for most genes. Consequently, relevant genes that influence the classification process are restricted because of the high number of irrelevant genes. The key informative genes represent a fundamental analysis task, such as phenotype classification [10], [26].

Fig. 1 shows a microarray data set, where each column represents a gene, each row represents a sample, and each cell represents the quantitative expression value of a sample in the gene. Every sample that retains a set of values is relevant to the gene expression and its own class label: normal or tumor; thus, a microarray data set contains one set of samples labeled "normal" and another set labeled "tumor".

One of the main problems of traditional machine learning techniques is that using them in clinical diagnostic applications affects their ability to detect false-positive samples accurately. For example, if a sample is not a member of the class library employed to train the classifier, it is classified as an error class. This flaw is unacceptable because it could lead to a misdiagnosis.

Microarray classification can be considered a regression problem that can be solved using the partial least squares method [19] and linear discriminant analysis [32].

- *S.-Y. Hsieh is with the Department of Computer Science and Information Engineering, Institute of Manufacturing Information Systems, Tainan 70101, Taiwan. E-mail: hsiehsy@mail.ncku.edu.tw.*
- *Y.-C. Chou is with the Institute of Medical Informatics, National Cheng Kung University, No. 1, University Road, Tainan 701, Taiwan E-mail: fish8716@gmail.com.*

Manuscript received 7 Jan. 2015; revised 21 July 2015; accepted 29 July 2015. Date of publication 28 Aug. 2015; date of current version 3 Feb. 2016.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TCBB.2015.2474389

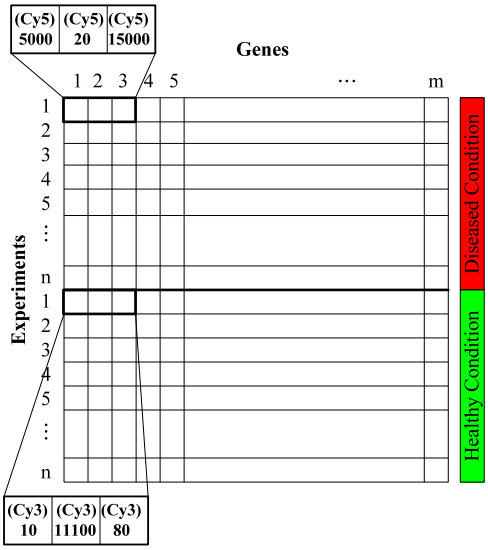

Fig. 1. A microarray sample.

Li et al. [20] classified cancers based on gene expression data by using the k-nearest neighbors rule, which can be used to classify unlabeled samples based on their similarity to the examples in the training set and by identifying the k-closest features in the set. Subsequently, the sample is distributed to the class that appears most frequently within the k-subset. Decision trees [5] and random forests [34], which generate solutions from the interactivities of multiple decision trees, can be used to remove irrelevant genes during the classification process. Neural networks [3], [22] can learn complex nonlinear regressions. Support vector machines [12], [27] are considered a general classification technique because of their robustness and correctness. The main disadvantage of these methods is that they are not designed to detect out-of-class samples; consequently, their application in clinical diagnostics is limited.

By using a gene expression graph (GEG)-based data structure to represent gene expression data, Benso et al. [4] constructed a clever classifier that topologically compared graphs of known classes with graphs of unknown assay samples. The GEG-based classifier can correctly classify samples into the corresponding classes and correctly detect out-of-class samples, and it is effective in clinical diagnostic applications because it reduces the rate of detecting false-positives. However, GEG-based classifiers require considerable computation time.

By using a weighted graph-based data structure, this study enhanced the GEG-based classifier proposed in [4]. Informative genes were selected by filtering weight values representing the relationships between genes, where greater weights indicate a stronger relationship between two genes. Therefore, the proposed approach can reduce the cost incurred by identifying irrelevant genes and improve the performance of existing methods. One of the main contributions of the proposed classifier is its ability to classify samples to the appropriate class and detect out-of-class samples that do not belong to any trained class. In addition, it reduces the computation time required for classification. To validate the efficiency of the proposed approach, this study experimentally compared the weighted graph-based classifier and a GEG-based classifier by conducting a set of microarray experiments of eight well-known diseases. The experimental results show that the weighted graph-based classifier can reduce the computation time required for classification, achieve comparable or superior performance in managing samples in the considered class library, and distinguish out-of-class samples that do not belong in any trained class. The main difference between the method proposed by Benso et al. [4] and the one proposed in this study is that the method proposed by Benso et al. compares pairwise interactions among all genes; however, the proposed method identifies genes associated with greater interaction with a given disease. To reduce the comparison time, the proposed method only selects the weighted values greater than 1, 5, or 7 for comparison.

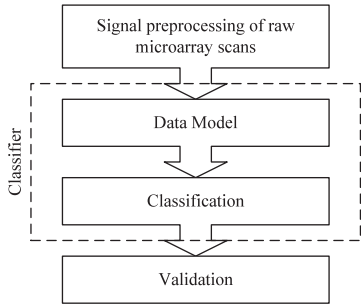

Fig. 2. Framework of microarray data processing.

The remainder of this paper is organized as follows: Section 2 presents the preliminaries of the graph-based data structure, and Section 3 proposes classification approach. The experimental results and a review of relevant research are presented in Sections 4 and 5, respectively. Finally, Section 6 offers the conclusion of this study.

2 Preliminaries

Classifying diseases based on microarray gene-expression information generally requires the raw microarray scan signals to undergo preprocessing, data modeling, classification, and validation. Fig. 2 depicts the framework of the microarray data process. At the signal processing stage, the raw image data are obtained from the microarray (sample) to calculate the expression level of each DNA probe. Because a microarray may be scratched, dusty, stained, dropped, spotted, or out of focus, this step must account for any type of acquisition error, hardware damage, and procedural difficulty. However, this procedure was beyond the scope of this study. The following sections focus on the problem of modeling microarray data and efficiently classifying diseases.

2.1 Data Model

Distinguishing relevant genes for a target disease can be achieved by identifying spots for genes that are differentially expressed between healthy and disease conditions [2]. To identify relevant genes through weighted GEGs (WGs), the definitions proposed by Benso et al. [4] are provided as follows:

Definition 1 (Gene expression profile). *[4] A microarray containing spots for m genes is a* gene expression profile $\vec{s}$. *Let*

$$\vec{s} = (gene_1, gene_2, \ldots, gene_m), \tag{1}$$

where $gene_j$ represents the gene expression level *of the jth gene (g_j) of the sample, where $j \in [1, m] \subset \mathbb{N}$. Define $gene_j$ as*

$$gene_j : \{\mathrm{Cy5}, \mathrm{Cy3}\} \to \mathbb{R}, \tag{2}$$

where Cy5 represents a diseased condition and Cy3 represents a healthy condition in a microarray.

In microarray technology, $gene_j$ may recognize an absolute expression level or a set of expression levels measured under various conditions. Microarrays provide each spot with two expression levels at two fluorescence intensities: the first level (i.e., Cy5) generates red fluorescence in the presence of a diseased condition, and the second level (i.e., Cy3) generates green fluorescence in the presence of a healthy condition. Define $gene_j$ as an index function and use $\vec{s}[j][Cy5]$ and $\vec{s}[j][Cy3]$ to indicate the two expression levels of gene g_j in the sample $\vec{s}$. The gene expression profiles obtained from n samples are typically organized in the form of a matrix called the gene expression matrix [16], where the ith row ($i \in [1, n] \subset \mathbb{N}$) represents the gene expression profile of the ith sample of the considered set. The following definition involves considering a multiclass classification problem.

Definition 2 (Training set). *[4] For K classes, where each class represents a disease, a classifier C is drawn from the space S of all possible gene expression profiles in a set of K classes, which can be expressed as*

$$C : S \to Y = \{y_1, y_2, \ldots, y_K\}, \tag{3}$$

where C is built from a training set Tr of l previously labeled samples, as follows:

$$Tr = \{t_1 = (\vec{s_1}, C(\vec{s_1})), \ldots, t_l = (\vec{s_l}, C(\vec{s_l}))\}. \tag{4}$$

Next, Tr can be split into K subsets:

$$Tr = \{Tr_1, Tr_2, \ldots Tr_K\}, \tag{5}$$

where each subset

$$Tr_i = \{(\vec{s_x}, C(\vec{s_x})) \in Tr \,|\, C(\vec{s_x}) = y_i\} \tag{6}$$

characterizes a separate class y_i. Therefore, samples with the same disease are distributed into one group.

Definition 3 (Z-score normalization). *[4] The standard score transformation converts the logratios of each gene of a sample into a new unit of the standard deviation where the normalized mean is equal to 0:*

$$z(Cy5, Cy3, \mu, \sigma) = \frac{\log_2 \frac{Cy5}{Cy3} - \mu}{\sigma}, \tag{7}$$

where μ and σ respectively denote the mean and standard deviation of logarithm ratios of all genes within the considered sample.

The presence of Cy3 or Cy5 components indicates a high number of corresponding DNA sequences, which enables the identification of overexpressed or silenced genes. Several studies on microarray data have recommended using the binary logarithm of the ratio $\frac{Cy5}{Cy3}$ to measure the differential expression of genes [11], [21]. This ratio considers the absolute intensity of the two channels enables genes with high-intensity Cy5 and Cy3 components to be distinguish from those with low-intensity Cy5 and Cy3 components, which, in the current study, facilitated the development of a rigorous and effective mathematical model for identifying relevant genes. Genes exhibiting a positive logratio are upregulated in the diseased condition (Cy5), whereas those in the healthy condition (Cy3) are identified by overexpressed relevant genes; otherwise, the sample is identified as silenced relevant genes. Because the experimental condition may introduce systematic biases to the experimental data that can shift or scale the expression levels of a sample, microarray normalization must be applied to compensate for these problems. The standard score (z-score) normalization is applied in Definition 3.

Definition 4 (Relevance). *[4] Let ε denote a threshold used to distribute overexpressed relevant genes, silenced relevant genes, and irrelevant genes. The expression range $ER \subset \mathbb{R}$ is used to indicate the full-scale range of Cy3 and Cy5 for the target technology, and $Rel_{\varepsilon,\mu,\sigma}$ is a function that is called to segment these components by assigning one of three possible values: 1 for overexpressed relevant genes, -1 for silenced relevant genes, and 0 for irrelevant genes:*

$$Rel_{\varepsilon,\mu,\sigma} : ER \times ER \to \{0, 1, -1\};$$

$$Rel_{\varepsilon,\mu,\sigma}(Cy5, Cy3) = \begin{cases} 1 & z(Cy5, Cy3, \mu, \sigma) > \varepsilon, \\ 0 & -\varepsilon \leqslant (Cy5, Cy3, \mu, \sigma) \leqslant \varepsilon, \\ -1 & z(Cy5, Cy3, \mu, \sigma) < -\varepsilon. \end{cases} \tag{8}$$

In Definition 4, there exists a severe cutoff between overexpressed and silenced genes, thereby limiting the algorithm's ability to identify irrelevant genes. The number of genes that present a perfectly null standard score is low. A threshold ε can be introduced to expand the irrelevant area, and it can be defined using various approaches. Although gene expression must be measured, a threshold can be associated with the intrinsic error. In addition, the threshold can be defined based on well-established methods for identifying differentially expressed genes: for example, a fold-change or t test [31].

A *graph* $G = (V, E)$ contains a pair of the *vertex set*V and *edge set* E, where V is a finite set and E is a subset of $\{(u, v) |\ (u, v)$ is an unordered pair of $V\}$. The two vertices, u and v, are *adjacent* if (u, v) is an edge in G; moreover, u and v are the *endpoints* of (u, v). A *loop* is an edge with equal endpoints, and *multiple edges* are those with the same pair of endpoints. A *simple graph* is one without loops or multiple edges. A *weighted graph* is one for which each vertex (edge) has an associated *weight*, which is typically derived using a *weight function*. A *complete graph* is a simple graph with pairwise adjacent vertices.

Definition 5 (Weighted gene expression graph). *Each Tr_i in $Tr = \{Tr_1, Tr_2, \ldots Tr_K\}$ can be modeled using a weighted graph, namely the* weighted GEG, *which is defined as $WG_i = (V_i, E_i, CRC_i, w_i)$, where*

TABLE 1
Initial Training Set Expression Levels Represented as a Gene Expression of Each Gene

Gene	a		b		c		d	
Tissue	Diseased	Healthy	Diseased	Healthy	Diseased	Healthy	Diseased	Healthy
Sam.1	30	10,000	100	12,000	2,000	10	15,000	100
Sam.2	30	2,500	120	15,000	5,000	30	900	1,030
Sam.3	50	30,000	50	2,000	1,000	1,099	15	5,000
Sam.4	45	20	8,500	100	2,400	20	10	100
Sam.5	30	5,000	13,000	80	4,000	150	1,050	50
Sam.6	100	6,000	80	15,000	6,000	25	10,100	20,150

- $V_i = \{v_x : v_x \in [1, m] \subset \mathbb{N}$ represents a gene g_x in the samples belonging to $Tr_i\}$,
- $CRC_i(v_x) = \sum_{\forall j \,|\, (\vec{s}_j, C(\vec{s}_j)) \in Tr_i} Rel_{\varepsilon,\mu,\sigma}(\vec{s}_j[x][Cy5], \vec{s}_j[x][Cy3])$,
- $E_i = \{(v_x, v_y) : x, y \in [1, m]$ for $x \neq y$ and $(CRC_i(v_x) \neq 0$ or $CRC_i(v_y) \neq 0)\}$, and
- $w_i(v_x, v_y) = \sum_{\forall j \,|\, (\vec{s}_j, C(\vec{s}_j)) \in Tr_i} (|Rel_{\varepsilon,\mu,\sigma}(\vec{s}_j[x][Cy5], \vec{s}_j[x][Cy3])| \wedge |Rel_{\varepsilon,\mu,\sigma}(\vec{s}_j[y][Cy5], \vec{s}_j[y][Cy3])|)$.

Each edge connects pairs of vertices representing corelevant genes that can be used to model the relationships among all relevant genes in a sample. A graph is a complete graph if n genes are corelevant in the same sample, in which each corresponding vertex is connected by an edge with each of the remaining $n-1$ vertices. This structural characteristic are crucial for identifying the features connected to each training subset Tr_i in building an efficient classifier. This concept is based on the hypothesis that corelevant genes within a sample are likely to be biologically relevant when once the target disease is characterized by considering a statistically significant number of experiments [28]. The weight, $w_i(v_x, v_y)$, of edge (v_x, v_y) is equivalent to the number of times that genes g_x (corresponding to v_x) and g_y (corresponding to v_y) are corelevant in the same sample over the entire set of samples comprising Tr_i. A higher weight indicates that the correlation between genes g_x and gene g_y is stronger.

The function $CRC_i : V_i \rightarrow \mathbb{Z}$ is called the *cumulative relevance count* function, which is associated with each node v_x in WG_i to ensure that a node with a positive CRC value represents gene g_x is overexpressed in most of the samples; otherwise, a node with a negative CRC value representing gene g_x is silenced in most of the samples; this function is summarized as follows:

- $CRC_i(v_x) > 0$: g_x is overexpressed in most of the samples of its training set.
- $CRC_i(v_x) < 0$: g_x is silenced in most of the samples of its training set.
- $CRC_i(v_x) = 0$: g_x is irrelevant in its training set.

In the definition of the edge weight, $\wedge$ denotes a logical AND operator returns a value of 1 when both of itsoperands are equal to 1. Simultaneously, $|\cdot|$ denotes the absolute value of the related relevance value. If both g_x and g_y are relevant in a case with an absolute value of 1, then each sample is considered as providing a unitary contribution to $w_i(v_x, v_y)$.

Example 1. Tables 1, 2, 3, and 4 and Figs. 3 and 4 depict an example of a WG_i construction from a Tr_i set of six samples. Table 1 shows an initial training set of gene expression levels. Each sample includes four genes and provides the Cy5 and Cy3 components for each gene are given. Table 2 shows the logratio calculated for each gene in each sample. The normalized z-scores, standard deviation, and logratios are presented in Table 3. Fig. 3 shows overexpressed relevant genes, silenced relevant genes, and irrelevant genes; the cutoff is performed according to a threshold $\varepsilon = 0.5$. Table 4 shows the CRC value of each vertex. Regarding the details at vertices A and B in this example, the gene expression relevance values in Fig. 3 shows that gene A is silenced in five samples (Samples 1, 2, and 3, Sample 5, and Sample 6), irrelevant in one sample (Sample 4), and not overexpressed in one sample. Hence, $CRC_i(v_a) = (-1) + (-1) + (-1) + 0 + (-1) + (-1) = -5$. Gene b is silenced in three samples, irrelevant in one sample, and overexpressed in two samples; hence, $CRC_i(v_b) = -1$. Fig. 4 shows the corresponding WG_i values, in which each vertex corresponds to a gene that is relevant in at least one sample. To calculate the weight of (v_a, v_b), the number of samples in which both genes are relevant were counted, including in the overexpressed and silenced genes, without considering the sign: Samples 1, 2, 5, and 6; hence, These are Samples 1, 2, 5 and 6; hence,

TABLE 2
Log-Ratios of Each Gene

Gene	a	b	c	d
Sam.1	−8.38	−6.91	7.64	7.23
Sam.2	−6.38	−6.97	7.38	−0.19
Sam.3	−9.23	−5.32	−0.01	−8.38
Sam.4	1.17	6.41	6.91	−3.32
Sam.5	−7.38	7.34	4.74	4.39
Sam.6	−5.91	−7.55	7.91	−1.00

TABLE 3
Normalization of the Microarray Data Set

Gene	a	b	c	d		
Sam.1	−0.95	−0.78	0.88	0.84	$\mu = -0.11$	$\sigma = 8.73$
Sam.2	−0.72	−0.81	1.33	0.20	$\mu = -1.54$	$\sigma = 6.69$
Sam.3	−0.84	0.10	1.38	−0.70	$\mu = -5.74$	$\sigma = 4.16$
Sam.4	−0.34	0.75	0.85	−1.47	$\mu = 2.79$	$\sigma = 4.83$
Sam.5	−1.47	0.77	0.38	0.32	$\mu = 2.27$	$\sigma = 6.57$
Sam.6	−0.61	−0.85	1.37	0.09	$\mu = -1.64$	$\sigma = 6.95$

TABLE 4
CRC Values

Gene	a	b	c	d
Sam.1	-1	-1	1	1
Sam.2	-1	-1	1	0
Sam.3	-1	0	1	-1
Sam.4	0	1	1	-1
Sam.5	-1	1	0	0
Sam.6	-1	-1	1	0
CRC	-5	-1	4	-1

$$w_i(v_a, v_b) = (|-1| \wedge |-1|) + (|-1| \wedge |-1|) + (|-1| \wedge |0|) + (|0| \wedge |1|) + (|-1| \wedge |1|) + (|-1| \wedge |-1|) = 4.$$

Relevant information can be easily added to the corresponding WG_i without any additional memory requirements, whereas new samples are available from new experiments involving the same pathology. The memory occupation of WG is determined by considering the number of genes independent of the number of experiments in the data set.

2.2 Classification Metrics

WG is an effective data structure for building an efficient classifier. The weighted graph-based classifier works by structurally comparing a pair of WG_{pat} and WG_{sam} to classify WG_{sam}, where WG_{pat} is built from the training set Tr_{pat} for a specified pathology and WG_{sam} is built from a sample $\vec{s}$. This comparison measures the overlap of WG_{sam} and WG_{pat} regarding the overexpressed/silenced genes (based on the CRC values of the vertices) and the relationships among gene expressions (weights of edges).

Definition 6 (Sample matching score (SMS)). *[4] The* sample matching score *determines the similarity between WG_{pat} and WG_{sam} by considering the vertices (genes) appearing in both graphs, which is defined as*$SMS(WG_{pat}, WG_{sam}) =$

$$\sum_{\forall (v_x, v_y) \in E(WG_{pat}) \cap E(WG_{sam})} \left(Z_{v_x} \cdot w_i(v_x, v_y) \cdot \frac{|Z_{v_x}|}{|Z_{v_x}| + |Z_{v_y}|} \right) + \left(Z_{v_y} \cdot w_i(v_x, v_y) \cdot \frac{|Z_{v_y}|}{|Z_{v_x}| + |Z_{v_y}|} \right), \tag{9}$$

where Z_{v_x} is calculated as $Z_{v_x} = CRC_{pat}(v_x) \cdot CRC_{sam}(v_x)$.

The term Z_{v_x} in Definition 6 is described as follows:

- $Z_{v_x} > 0$: g_x is both silenced and overexpressed in both WG_{sam} and WG_{pat}.

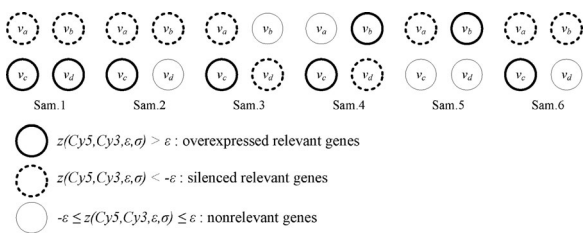

Fig. 3. Gene expression relevance and relevance with $\varepsilon = 0.5$.

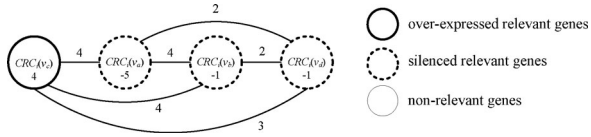

Fig. 4. Weighted gene expression graph.

- $Z_{v_x} < 0$: g_x is overexpressed in WG_{sam} and silenced in WG_{pat}, or vice versa.
- $Z_{v_x} = 0$: g_x is irrelevant in either WG_{sam} or WG_{pat}.

This term is used to quantify the magnitude at which the expression of g_x in $\vec{s}$ is similar to the expression of the same gene in Tr_{pat}. If more genes have positive Z values, then the SMS would be higher, indicating that WG_{sam} and WG_{pat} have higher degree of similarity.

Definition 7 (Maximum matching score (MMS)). *[4] The* maximum matching score *is a special case of SMS that measures SMS under the condition that the structure and associated values in WG_{sam} are identical to those in WG_{pat}, which is defined as*

$$MMS(WG_{pat}) = \sum_{\forall (v_x, v_y) \in E(WG_{pat})} \left(w_i(v_x, v_y) * \frac{CRC_{pat}(v_x)^2 + CRC_{pat}(v_y)^2}{|CRC_{pat}(v_x)| + |CRC_{pat}(v_y)|} \right). \tag{10}$$

Definition 8 (Proximity score (PS)). *The* proximity score *measures the similarity between two WG values as follows:*

$$PS(WG_{pat}, WG_{sam}) = \frac{SMS(WG_{pat}, WG_{sam})}{MMS(WG_{pat})}. \tag{11}$$

The PS is used to measure whether the WG_{sam} built based on a sample $\vec{s}$ is similar to the class characterized by WG_{pat}. A positive PS indicates that the similarities between the two graphs, where a higher score indicates a higher degree of similarity between the sample and the class. Conversely, a negative value indicates contrasting structural information between two graphs, indicating a high degree of dissimilarity between the sample and class.

The computation of each PS with all of the classes is independent of the number of classes considered in the problem, which makes it an absolute indicator of the similarity of a sample with a specified class. PS measures typically depend on the number of classes, and must be explained in relation to the measurement of all classes considered in the problem. Accordingly, the weighted graph-based classifier can construct an appropriate data model and completely rebuild the prediction model when new classes are included in the classification process.

3 Proposed Algorithms

The proposed weighted graph-based classifier performs classification by representing gene expression profiles as weighted gene expression graphs. The graph is built from raw gene expression measures obtained through experimentally scanning and preprocessing microarray data. The

graph is constructed to facilitate the classification process, prevent the preprocessing steps from affecting the classification process, and work with reduced dimensionality while minimizing the loss of raw data.

3.1 Algorithm for Calculating the Required Values

Algorithms 1 shows how Rel and CRC are computed. At Step 0, the threshold ε of $Rel_{\varepsilon,\mu,\sigma}$ is set to 0. At Step 1, the file of the microarray data set is assigned to a two-dimensional auxiliary array, denoted as $WGMatrix_i$. At Step 2, each gene in each sample (i.e., the value of each entry in $WGMatrix_i$) is normalized according to (7) in Definition 3. At Step 3, $Rel_{\varepsilon,\mu,\sigma}$ for each gene in a sample is determined according to (8) in Definition 4, and all Rel values are output into a file. At Step 4, $CRC_i(v_x)$ is calculated to express each gene g_x in the data set (corresponding to each vertex v_x in the graph) according to Definition 5, the values are stored into another one-dimensional auxiliary array $CRC_i[\cdot]$, and all of the CRC values are output into a file.

Algorithm 1. CALCULATE_REL_CRC

Require: A file of microarray data set.
Ensure: $Rel_{\varepsilon,\mu,\sigma}$ and CRC_i.
Step 0: Set $\varepsilon = 0$.
Step 1: Input the microarray data set into $WGMatrix_i$.
Step 2: Calculate the z-scores according to (7) in Definition 3.
2-1: Calculate μ for each sample.
2-2: Calculate σ for each sample.
2-3: For each entry in the $WGMatrix_i$, subtract μ and divide σ. Input the results into $WGMatrix_i$.
Step 3: Calculate $Rel_{\varepsilon,\mu,\sigma}$, according to (8) in Definition 4. For the value a of each entry in $WGMatrix_i$, execute the following substeps:
3-1: If $a > \varepsilon$, then assign 1 to the entry.
3-2: If $a < -\varepsilon$, then assign -1 to the entry.
3-3: If $-\varepsilon \leq a \leq \varepsilon$, then assign 0 to the entry.
3-4: Output the result values in $WGMatrix_i$, which are the desired Rel values, to a file.
Step 4: Calculate CRC_i according to Definition 5.
4-1: Sum up the Rel values of the same gene g_x from various samples and assign the results into $CRC_i[x]$.
4-2: Output the CRC values (stored in the array $CRC_i[\cdot]$) to a file.

3.2 Classification Algorithms

The CALCULATE_REL_CRC algorithm was applied to calculate CRC_{pat}, CRC_{sam}, and the Rel to classify the pathology samples to various classes. Subsequently, the CALCULATE_MMS and CALCULATE_SMS algorithms were applied to calculate the MMS and SMS, after which the CALCULATE_PS algorithm was applied to measure the similarity between WG_{pat} and WG_{sam}.

The CALCULATE_MMS algorithm has four inputs: $nrows$, which denotes the number of samples in the pathology; $ngenes$, which is the number of gene types in the pathology; the CRC of the pathology; and the Rel of the pathology. In the CALCULATE_MMS algorithm, let num and den be temporary values for calculating the MMS. Set the $weight\text{-}threshold$ at 5 to filter the significant genes of the pathology. When the weight of the corelevant gene pairs is higher than 5, the weight is contributed to measure the MMS based on (10) in Definition 7. The $MMS(WG_{pat})$ value can be derived when all vertices in the WG_{pat} have been computed.

Algorithm 2. CALCULATE_MMS ($nrows$, $ngenes$, Rel, CRC_{pat})

$mms := 0$
$w_{x,y} := 0$
for $x := 0$ to $ngenes - 2$ **do**
  for $y := x + 1$ to $ngenes - 1$ **do**
    for $r := 0$ to $nrows - 1$ **do**
      if $Rel[r][x] * Rel[r][y] \neq 0$ **then**
        $num = pow(CRC_{pat}[x], 2) + pow(CRC_{pat}[y], 2)$
        $den = abs(CRC_{pat}[x]) + abs(CRC_{pat}[y])$
        $w_{x,y} += 1$
        if $w_{x,y} > 5$ **then**
          $mms += (num/den) * w_{x,y}$
RETURN mms

The CALCULATE_SMS algorithm has five inputs: $nrows$, which is the number of samples in the pathology; $ngenes$, denoting the number of types of genes in the pathology; the CRC of the pathology; the CRC of the sample; and the Rel of the pathology. The algorithm ignores any vertex in a sample where CRC $= 0$, which indicates that it is an irrelevant gene. In the CALCULATE_SMS algorithm, let num and den be temporary values for calculating the SMS. Set the $weight\text{-}threshold$ at 5 to filter the significant genes of the pathology. Using a method similar to that used in the CALCULATE_MMS algorithm, the weight is contributed to measure SMS based on (9) in Definition 6. The value of $SMS(WG_{pat}, WG_{sam})$ can be derived when all vertices in the WG_{pat} have been computed.

Algorithm 3. CALCULATE_SMS ($nrows$, $ngenes$, Rel, CRC_{pat}, CRC_{sam})

$sms := 0$
$w_{x,y} := 0$
for $x := 0$ to $ngenes - 2$ **do**
  if $CRC_{sam}[x] \neq 0$ **then**
    $Z_x = CRC_{pat}[x] * CRC_{sam}[x]$
  for $y := x + 1$ to $ngenes - 1$ **do**
    if $CRC_{sam}[y] \neq 0$ **then**
    $Z_y = CRC_{pat}[y] * CRC_{sam}[y]$
    $num = (Z_x * abs(Z_x)) + (Z_y * abs(Z_y))$
    $den = abs(Z_x) + abs(Z_y)$
  for $r := 0$ to $nrows - 1$ **do**
    if $Rel[r][x] * Rel[r][y] = 1$ **then**
      $w_{x,y} += 1$
      if $w_{x,y} > 5$ **then**
        $sms += (num/den) * w_{x,y}$
RETURN sms

The CALCULATE_PS algorithm calculates $PS(WG_{pat}, WG_{sam})$ for classification. At Step 0, the $weight\text{-}threshold$ is initialized at 5, and used to filter the significant genes. At Step 1, the stored Rel values output by the CALCULATE_REL_CRC algorithm for the pathology are input into the classifier. At Step 2, the stored CRC values generated by the CALCULATE_REL_CRC algorithm for the pathology are input

TABLE 5
The Composition of the Experimental Data Sets: Number of Test Samples, Number of Training Samples, and Size of Microarray Chips

Classifiable samples				Out-of-class samples		
Disease	# of test samples	# of training samples	chip size	Disease	# of test samples	chip size
DLBCL	10	20	9 k	FL	5	37 k
CLLww	8	15	18 k	SLT	6	24 k
ALL	8	18	24 k	CBF-AML	6	45 k
HB	10	11	37 k			
CBCL	6	10	45 k			
total	42	74			17	

into the classifier. At Step 3, the stored CRC values generated by the CALCULATE_REL_CRC algorithm for the sample are input into the classifier. At Step 4, $MMS(WG_{pat})$ is calculated using the CALCULATE_MMS algorithm. At Step 5, the value of $SMS(WG_{pat}, WG_{sam})$ is calculated using the CALCULATE_SMS algorithm. At Step 6, $PS(WG_{pat}, WG_{sam})$ is calculated according to (11) in Definition 8. The sample represented by WG_{sam} was subsequently classified based on $PS(WG_{pat}, WG_{sam})$.

Algorithm 4. CALCULATE_PS

Require: Rel, CRC_{pat} and CRC_{sam}.
Ensure: $PS(WG_{pat}, WG_{sam})$.
Step 0: Set the $weight\text{-}threshold$ value at 5.
Step 1: Input the file containing the Rel values.
Step 2: Input the file containing the CRC_{pat} values.
Step 3: Input the file containing the CRC_{sam} values.
Step 4: Calculate $MMS(WG_{pat})$ by using the CALCULATE_MMS algorithm.
Step 5: Calculate $SMS(WG_{pat}, WG_{sam})$ by using the CALCULATE_SMS algorithm.
Step 6: Calculate $PS(WG_{pat}, WG_{sam})$ according to (11) in Definition 8.

Inputting a sample for comparison with various pathologies to determine whether it is associated with a particular disease (represented by out-of-class) enables the classification of samples according to their PS. The method proposed in [24] was employed to determine the manner in which the PSs are distributed between appropriately classified and out-of-class samples by estimating the kernel density. When the PS of an unknown sample with pathologies is higher than 0.04, it can be assumed that the sample belongs to the class with the highest PS. Otherwise, all PSs are less than 0.04, and this sample is determined as out-of-class.

4 EXPERIMENTAL RESULTS

This section presents an experimental validation of the weighted graph-based classifier.

4.1 Experimental Design

The proposed experimental design involved several classification experiments on various microarray data sets where both the proposed weighted graph-based classifier and GEG-based classifier were used [4]. Both classifiers were implemented using ANSI C code. Compared with the GEG-based classifier, the proposed weighted graph-based classifier can accurately classify samples, distinguish out-of-class samples, and shorten the computation time.

4.2 Data Source and Data Set

The data sets were obtained from the cDNA Stanford Microarray Database [25]. The Stanford Microarray Database collection contains a high number of experiments involving cDNA chip technology. cDNA chips use two colors to distinguish tissues: red for diseased tissue and green for healthy tissue. Every experiment was linked with the image of the identical microarray and a CSV format text file.

A total of eight pathologies were considered in this experiment, as follows [1], [23]:

- Diffuse Large B-Cell Lymphoma (DLBCL): a type of non-Hodgkin and aggressive lymphoma disease.
- B-cell Chronic Lymphocytic Leukemia wait and watch (CLLww): low-risk chronic lymphocytic leukemia patients who were treated using the watch and wait method.
- Acute Lymphocytic Leukemia (ALL): a form of leukemia. It is a type of blood cancer affecting bone marrow cells.
- Cutaneous B-Cell Lymphomas (CBCL): a type of B-cell lymphoma of the skin that appears as a reddish rash, lump, or nodule.
- Healthy Blood (HB): blood samples from healthy humans.
- Follicular Lymphoma (FL): a frequently reported type of non-Hodgkin's lymphoma. It is also referred to as an "indolent" or "low-grade" lymphoma.
- Solid Lung Tumor (SLT): a disease that has uncontrolled cell growth in the tissues of the lung.
- Core Binding Factor Acute Myeloid Leukemia (CBF-AML): two distinct subsets, CBF-alpha and CBF-beta, of AML that are characterized by recurrent favorable chromosome translocation.

Various types of microarray chip were tested to determine the performance of the proposed algorithm, including 9 K (9,216 spots), 18 K (18,432 spots), 24 K (24,168 spots), 37 K (37,632 spots), and 45 K (43,196 spots). Table 5 shows the composition of the experimental data set. As shown in Table 5, a training set comprising 74 samples and a test set of 42 classifiable samples were obtained from the first five pathologies; the remaining three pathologies were used to create a test set of 17 out-of-class samples. The training set did not include any samples from the test sets.

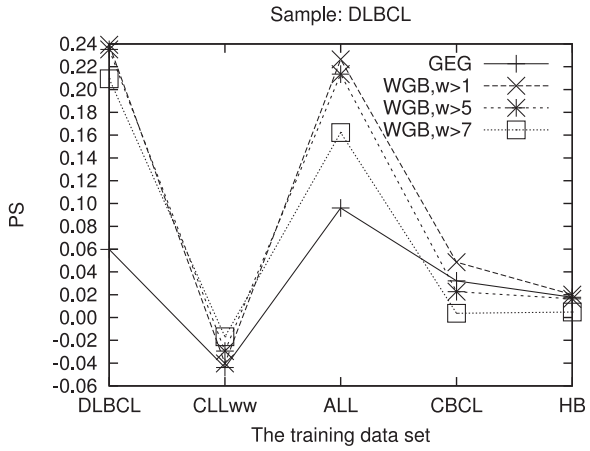

Fig. 5. A sample of DLBCL data set.

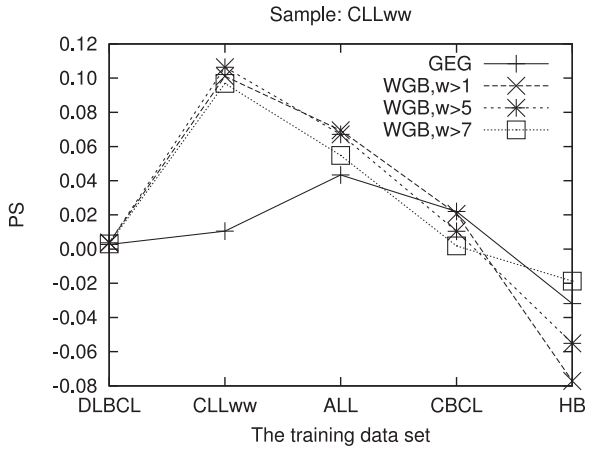

Fig. 6. A sample of CLLww data set.

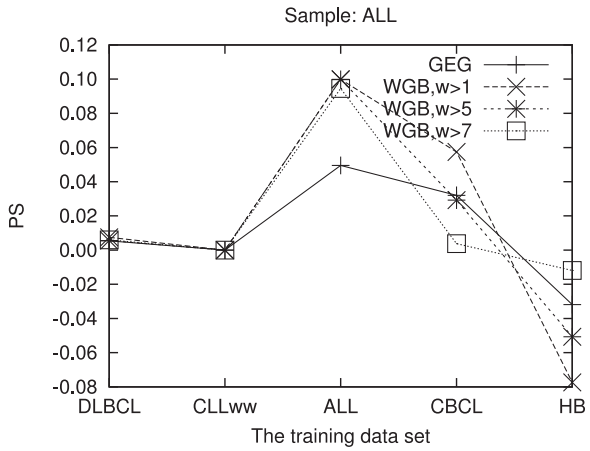

Fig. 7. A sample of ALL data set.

4.3 Comparison Results

This section shows a comparison between the performance of the proposed weighted graph-based classifier and that of the GEG-based classifier. All of the experiments were performed using logratios rather than raw gene expression data to compare the classification results. The analysis results indicate that the proposed weighted graph-based classifier is a novel valid microarray classification tool that is practicable in medical diagnostics. The proposed method can also shorten the computation time considerably.

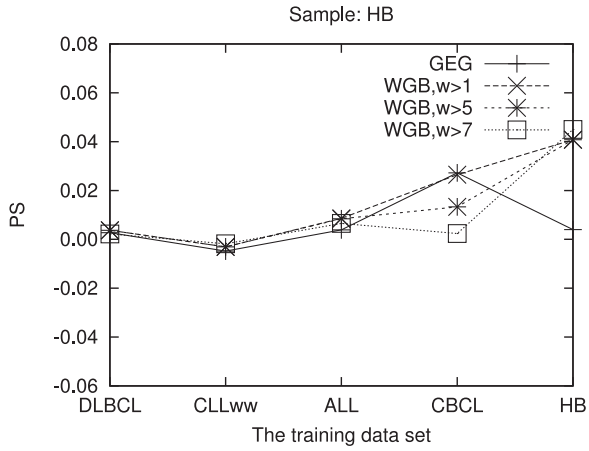

Fig. 8. A sample of HB data set.

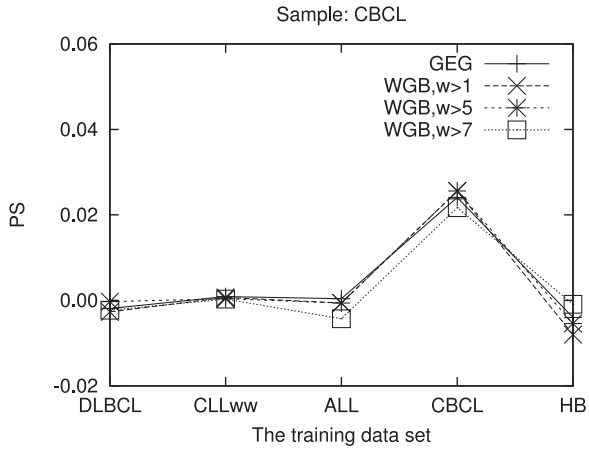

Fig. 9. A sample of CBCL data set.

4.3.1 *Classifiable Samples*

The performance of the weighted graph-based classifier in classifying samples was compared with that of the GEG-based classifier. The target class was constructed using the five classes of the classifiable samples. Figs. 5, 6, 7, 8, and 9 show the performance of the weighted graph-based classifier with weighted threshold > 1, > 5, and > 7, as well as the GEG-based classifier in the classifiable samples. Fig. 5 shows the performance of the PS values of the DLBCL data set in classifying various diseases. The PS value of ALL was the highest (> 0.04) in the GEG-based classifier. Hence, the GEG-based classifier classified this sample as ALL. Conversely, the proposed weighted graph-based classifier determined that the PS value in DLBCL was the highest (> 0.04) with a weighted threshold > 1, > 5, and > 7. Therefore, the proposed weighted graph-based classifier correctly classified the DLBCL sample more effectively compared with the GEG-based classifier.

Fig. 6 shows the performance of the PS values of the CLLww data set in classifying various diseases. The PS value for ALL was the highest (> 0.04) in the GEG-based classifier. Hence, the GEG-based classifier determined that this sample is ALL. Conversely, the proposed weighted graph-based classifier determined that the PS value in CLLww was the highest (> 0.04) with a weighted threshold > 1, > 5, and > 7. Therefore, the proposed weighted

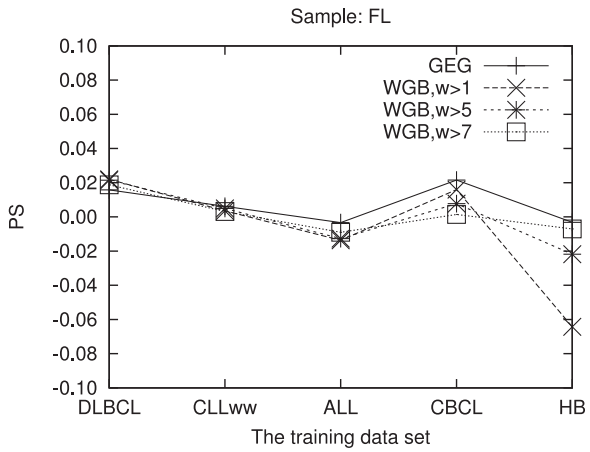

Fig. 10. A sample of FL data set.

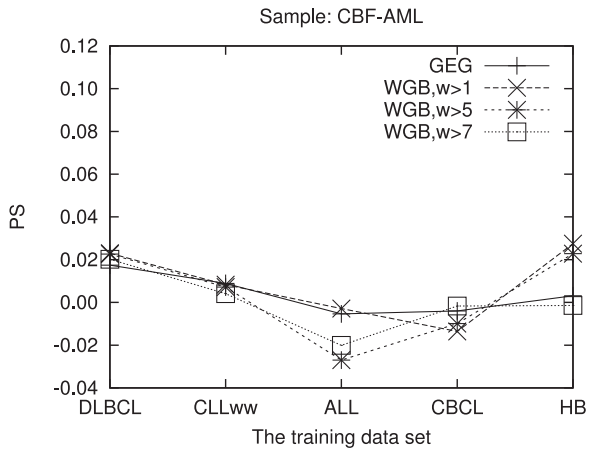

Fig. 11. A sample of CBF-AML data set.

graph-based classifier correctly classified the sample of CLLww more effectively compared with the GEG-based classifier.

Fig. 7 shows the performance of the PS values of the ALL data set in classifying various diseases. Both the GEG-based classifier and the proposed weighted graph-based classifier with a weighted threshold > 1, > 5, and > 7 determined that the PS values for ALL are the highest (> 0.04). Therefore, both classifiers correctly classified the ALL sample.

Fig. 8 shows the performance of the PS values of the HB data set in classifying various diseases. The PS values were less than 0.04 in the GEG-based classifier; hence, the GEG-based classifier determined that this sample was out-of-class. Conversely, the proposed weighted graph-based classifier determined that the PS value of the HB sample was the highest (> 0.04) with a weighted threshold > 1, > 5, and > 7. Therefore, the proposed weighted graph-based classifier correctly classified the HB sample more effectively compared with the GEG-based classifier.

Fig. 9 shows the performance of the PS values of the CBCL data set in classifying various diseases. The PS values were less than 0.04 in both the GEG-based classifier and the proposed weighted graph-based classifier with a weighted threshold > 1, > 5, and > 7. Thus, both classifiers determined that this sample was out-of-class. Therefore, both classifiers were unable to classify the CBCL sample.

4.3.2 Out-of-Class Samples

The performance of the proposed weighted graph-based classifier in detecting out-of-class samples was compared with that of the GEG-based classifier. The target class was constructed using the five classes of classifiable samples and the out-of-class samples from the other three classes. Figs. 10, 11, and 12 show the performance of the GEG-based classifier and proposed weighted graph-based classifier with a weighted threshold > 1, > 5, and > 7 in the out-of-class samples.

Fig. 10 shows the performance of the PS values of the FL data set in classifying various diseases. All PS values were less than 0.04 in both the GEG-based classifier and proposed weighted graph-based classifier with a weighted threshold > 1, > 5, and > 7. Thus, both classifiers determined that this sample is out-of-class.

Fig. 11 shows that the CBF-AML data set is out-of-class, and Fig. 12 shows that the SLT data set is out-of-class.

4.3.3 Computation Time

When comparing the performance of classifiers, the overall computation time is a crucial performance measure. Therefore, the computation time for calculating the Rel and CRC values was considered in this study. Table 6 shows the computation time of each step for the GEG-based classifier and the proposed weighted graph-based classifier with a weighted threshold > 1, > 5, and > 7.

The execution time for the proposed weighted graph-based classifier and GEG-based classifier to calculate the Rel and CRC values is negligible because these values are exceptionally low. Fig. 13 shows only the classification time. The classification of the proposed weighted graph-based classifier with a weighted threshold > 1, > 5, and > 7 is lower than that of the GEG-based classifier. This difference occurred because the proposed weighted graph-based classifier filtered the nonsignificant genes, thereby reducing the computation costs.

5 Related Work

This section reviews some relevant studies. Li et al. [18] proposed a manifold learning method for mapping the gene expression data into a low-dimensional space, and then

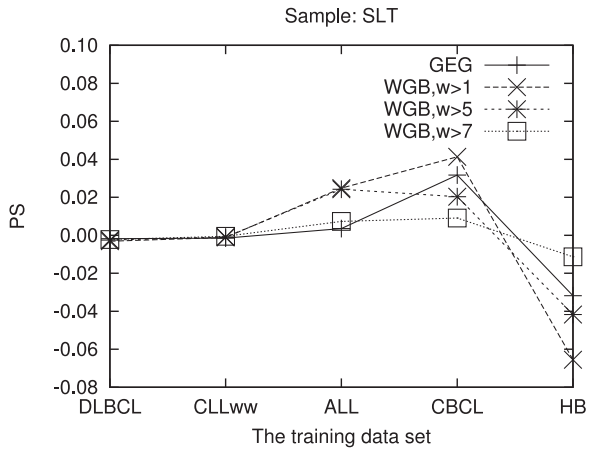

Fig. 12. A sample of SLT data set.

TABLE 6
Comparison of Execution Time (Seconds) for Various Classifiers

	GEG	WGB,w > 1	WGB,w > 5	WGB,w > 7
Calculating the Rel values and the CRC values	2	2	2	2
Classification	3,487	3,337	3,349	3,337
Total	3,489	3,339	3,351	3,339

explored the intrinsic structure of the features to improve the accuracy of classifying microarray data. The method proposed in that study projected the gene-expression data into a subspace with high intraclass compactness and interclass separability. Leung and Hung [17] employed a multifilter-multiwrapper (MFMW) approach to improve the accuracy and robustness of classifications and to identify potential biomarker genes.

Tumor classification is a critical application domain of gene-expression data. Huang and Zheng [15] proposed a method for tumor classification by using gene-expression data. They first performed an independent component analysis to model the gene-expression data, and then applied an optimal scoring algorithm to classify the data. The approach proposed in that study involved exploiting the high-order statistical information contained in the gene-expression data.

Zhang [35] et al. proposed a novel approach of tumor classification based on Wavelet packet transforms (WPTs) and neighborhood rough sets (NRSs). First, the classification features were extracted by the WPT and the decision tables were then formed. Second, the attributes of the decision tables were reduced by the NRS method. Third, a feature subset with few attributes and high classification ability was obtained.

Wang et al. [30] proposed a tumor classification approach based on an ensemble of probabilistic neural networks (PNNs) and gene-reduction-based NRS model. Informative genes were initially selected by a gene-ranking method using an iterative search margin algorithm, and they were further refined through gene reduction to select the minimal gene subsets. Finally, the candidate base PNN classifiers trained by each of the selected gene subsets were integrated using a majority voting strategy to construct an ensemble classifier.

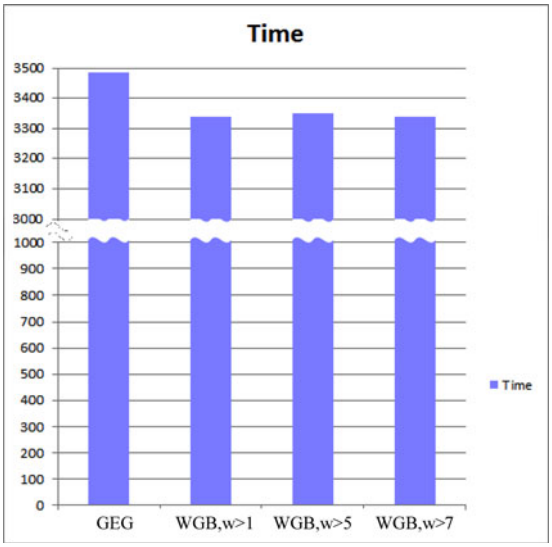

Fig. 13. Computation time for various classifiers.

Previous studies have shown that sparse representation (SR) by l_1-norm minimization is robust to noise, outliers, and incomplete measurements. Zheng et al. [38] proposed a novel SR-based method for classifying tumors based on gene-expression data. A set of metasamples were extracted from the training samples, and an input testing sample was then represented as a linear combination of these metasamples by using the l_1-regularized least square method. Classification was performed according to a discriminating function defined based on the representation coefficients.

Zheng et al. [37], [39] applied a penalized matrix decomposition (PMD) method to gene-expression data to extract metasamples for clustering. The extracted metasamples successfully captured the inherent structures of samples from the same class. Subsequently, the PMD factors of a sample over the metasamples were used as class indicators. In addition, the PMD factors were used as an index for determining the cluster number; thus, the method proposed in that study could identify modules in gene-expression data of conterminous developmental stages.

Based on gene expression profiles, Wang et al. [29] proposed a novel method for classifying tumors according to correlation filters in order to identify the overall pattern of tumor subtype hidden in differentially expressed genes. Two correlation filters (minimal average correlation energy and optimal tradeoff synthetic discriminant function) were proposed to determine whether test samples matched the templates synthesized for each subclass.

6 Conclusion

This paper proposes a weighted graph-based classifier for classifying microarray data sets. The results show that the performance of the proposed classifier is comparable with or superior to that of current methods. The proposed classifier can correctly detect out-of-class samples in addition to correctly classifying samples in the corresponding classes. Furthermore, biological features were used to reduce the computation costs. Future research will address the identification of similar diseases and the PSs of large-scale data sets.

Acknowledgments

This research was supported in part by (received funding from) the Headquarters of University Advancement, National Cheng Kung University, which is sponsored by the Ministry of Education, Taiwan, R.O.C. S.Y. Hsieh is the corresponding author.

REFERENCES

[1] A. A. Alizadeh, M. B. Eisen, R. E. Davis, C. Ma, I. S. Lossos, A. Rosenwald, J. C. Boldrick, H. Sabet, T. Tran, X. Yu, J. I. Powell, L. Yang, G. E. Marti, T. Moore, J. J. Hudson, L. Lu, D. B. Lewis, R. Tibshirani, G. Sherlock, W. C. Chan, T. C. Greiner, D. D. Weisenburger, J. O. Armitage, R. Warnke, R. Levy, W. Wilson, M. R. Grever, J. C. Byrd, D. Botstein, P. O. Brown, and L. M. Staudt, "Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling," *Nature*, vol. 403, no. 6769, pp. 503–511, Feb. 2000.

[2] D. B. Allison, X. Cui, G. P. Page, and M. Sabripour, "Microarray data analysis: From disarray to consolidation to consensus," *Nature Rev. Genetics*, vol. 7, no. 1, pp. 55–65, May 2006.

[3] F. Azuaje, "A computational neural approach to support the discovery of gene function and classes of cancer," *IEEE Trans. Biomed. Eng.*, vol. 48, no. 3, pp. 332–339, Mar. 2001.

[4] A. Benso, S. Di Carlo, and G. Politano, "A cDNA microarray gene expression data classifier for clinical diagnostics based on graph theory," *IEEE/ACM Trans. Comput. Biol. Bioinformat.*, vol. 8, no. 3, pp. 577–591, Jun. 2011.

[5] J. Breiman, L. ad Friedman, C. J. Stone, and R. Olshen, *Classification and Regression Trees*. New York, NY, USA: Talyor and Francis, 1984.

[6] P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, "Metagenes and molecular pattern discovery using matrix factorization," *Proc. Nat. Acad. Sci. USA*, vol. 101, no. 12, pp. 4164–4169, 2004.

[7] C. Cheadle, M. P. Vawter, W. J. Freed, and K. G. Becker, "Analysis of microarray data using Z score transformation," *J. Mol. Diagnostics*, vol. 5, no. 2, pp. 73–81, May 2003.

[8] H.-Y. Chuang, H. Liu, S. Brown, C. McMunn-Coffran, C.-Y. Kao, and D. F. Hsu, "Identifying significant genes from microarray data," in *Proc. 4th IEEE Symp. Bioinformat. Bioeng.*, May 2004, pp. 358–365.

[9] E. R. Dougherty, "The fundamental role of pattern recognition for gene-expression/microarray data in bioinformatics," *Pattern Recognit.*, vol. 38, no. 12, pp. 2226–2228, Dec. 2005.

[10] B. Dost, C. Wu, A. Su, and V. Bafna, "TCLUST: A fast method for clustering genome-scale expression data," *IEEE/ACM Trans. Comput. Biol. Bioinformat.*, vol. 8, no. 3, pp. 808–818, May/Jun. 2011.

[11] B. P. Durbin, J. S. Hardin, D. M. Hawkins, and D. M. Rocke, "A variance-stabilizing transformation for gene-expression microarray," *Bioinformatics*, vol. 18, no. 1, pp. 105–110, Mar. 2002.

[12] T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler, "Support vector machine classification and validation of cancer tissue sample using microarray expression data," *Bioinformatics*, vol. 16, no. 10, pp. 906–914, May 2000.

[13] K. Han and J. Lee, "GeneNetFinder2: Improved inference of dynamic gene regulatory relations with multiple regulators," *IEEE/ACM Trans. Comput. Biol. Bioinformat.*, Doi: 10.1109/TCBB.2015.2450728.

[14] S.-Y. Hsieh and Y.-C. Chou, "A weighted graph-based classifier for microarray gene expression data," presented at the 10th International Conf. Intelligent Computing, Taiyuan, Shanxi Province, China, Aug. 3–6, 2014.

[15] D.-S. Huang and C.-H. Zheng, "Independent component analysis-based penalized discriminant method for tumor classification using gene expression data," *Bioinformatics*, vol. 22, no. 15, pp. 1855–1862, 2006.

[16] D. Jiang, C. Tang, and A. Zhang, "Cluster analysis for gene expression data: A survey," *IEEE/ACM Trans. Knowl. Data Eng.*, vol. 16, no. 11, pp. 1370–1386, Nov. 2004.

[17] Y. Leung and Y. Hung, "A multiple-filter-multiple-wrapper approach to gene selection and microarray data classification," *IEEE/ACM Trans. Comput. Biol. Bioinformat.*, vol. 7, no. 1, pp. 108–117, Jan./Mar. 2010.

[18] B. Li, C.-H. Zheng, D.-S. Huang, L. Zhang, and K. Han, "Gene expression data classification using locally linear discriminant embedding," *Comput. Biol. Med.*, vol. 10, pp. 802–810, 2010.

[19] D. Nguyen and D. Rocke, "Tumor classification by partial least squares using microarray gene expression data," *Bioinformatics*, vol. 18, no. 1, pp. 39–50, Jan. 2002.

[20] 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*, vol. 17, no. 12, pp. 1131–1142, Jun. 2001.

[21] M. K. Kerr, M. Martin, and G. A. Churchill, "Analysis of variance for gene expression microarray data," *J. Comput. Biol.*, vol. 7, no. 6, pp. 819–837, Dec. 2000.

[22] 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 Med.*, vol. 7, no. 6, pp. 673–679, Jun. 2001.

[23] C. Palmer, M. Diehn, A. Alizadeh, and P. O. Brown, "Cell-type specific gene expression profiles of leukocytes in human peripheral blood," *BMC Genomics*, vol. 7, p. 115, May 2006.

[24] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, vol. 33, no. 3, pp. 1065–1076, Sep. 1962.

[25] (2012). cDNA Stanford Microarray Database. [Online]. Available: http://genome-www.stanford.edu/

[26] Q. Song, J. Ni, and G. Wang, "A fast clustering-based feature subset selection algorithm for high dimensional data," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 1, pp. 1–14, Aug. 2011.

[27] A. Statnikov, L. Wang, and C. Aliferis, "A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification," *BMC Bioinformat.*, vol. 9, no. 1, p. 319, Jul. 2008.

[28] J. Stuart, E. Segal, D. Koller, and S. Kom, "A gene-coexpression network for global discovery of conserved genetic modules," *Science*, vol. 302, no. 5643, pp. 249–255, Oct. 2003.

[29] S.-L. Wang, Y.-H. Zhu, W. Jia, and D.-S. Huang, "Robust classification method of tumor subtype by using correlation filters," *IEEE/ACM Trans. Comput. Biol. Bioinformat.*, vol. 9, no. 2, pp. 580–591, Mar./Apr. 2012.

[30] S.-Lin Wang, X. Li, S. Zhang, J. Gui, and D.-S. Huang, "Tumor classification by combining PNN classifier ensemble with neighborhood rough set based gene reduction," *Comput. Biol. Med.*, vol. 40, pp. 179–189, 2010.

[31] D. M. Witten and R. Tibshirani. (2009). A comparison of fold-change and the t-statistic for microarray data analysis. [Online]. Available: http://www-stat.standford.edu/tibs/ftp/FCTComparison.pdf

[32] P. Xu, G. N. Brock, and R. S. Parrish, "Modified linear discriminant analysis approaches for classification of high-dimensional microarray data," *Comput. Stat. Data Anal.*, vol. 53, no. 5, pp. 1674–1687, Mar. 2009.

[33] Y. Yoon, S. Bien, and S. Park, "Microarray data classifier consisting of k-top rank-comparison decision rules with a variable number of genes," *IEEE Trans. Syst., Man, Cybern.-Part C: Appl. Rev.*, vol. 40, no. 2, pp. 216–226, Mar. 2010.

[34] H. Zhang, C.-Y. Yu, and B. Singer, "Cell and tumor classification using gene expression data: Construction of forests," in *Proc. Nat. Acad. Sci. USA*, vol. 100, no. 7, pp. 4168–4172, Apr. 2003.

[35] S.-W. Zhang, D.-S. Huang, and S.-L. Wang, "A method of tumor classification based on wavelet packet transforms and neighborhood rough set," *Comput. Biol. Med.*, vol. 40, pp. 430–437, 2010.

[36] C. H. Zheng, D. S. Huang, and L. Shang, "Feature selection in independent component subspace for microarray data classification," *Neurocomputing*, vol. 69, no. 16, pp. 2407–2410, 2006.

[37] C.-H. Zheng, D.-S. Huang, L. Zhang, and X.-Z. Kong, "Tumor clustering using nonnegative matrix factorization with gene selection," *IEEE Trans. Inf. Technol. Biomed.*, vol. 13, no. 4, pp. 599–607, Jul. 2009.

[38] C.-H. Zheng, L. Zhang, T.-Y. Ng, S. C. Shiu, and D.-S. Huang, "Metasample-based sparse representation for tumor classification," *IEEE/ACM Trans. Comput. Biol. Bioinformat.*, vol. 8, no. 5, pp. 1273–1282, Sep./Oct. 2011.

[39] C.-H. Zheng, L. Zhang, V. T.-Y. Ng, C.-K. Shiu, and D.-S. Huang, "Molecular pattern discovery based on penalized matrix decomposition," *IEEE/ACM Trans. Comput. Biol. Bioinformat.*, vol. 8, no. 6, pp. 1592–1603, Nov./Dec. 2011.

[40] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.

Sun-Yuan Hsieh received the PhD degree in computer science from National Taiwan University, Taipei, Taiwan, in June 1998. He then served the compulsory two-year military service. From August 2000 to January 2002, he was an assistant professor in the Department of Computer Science and Information Engineering, National Chi Nan University. In February 2002, he joined the Department of Computer Science and Information Engineering, National Cheng Kung University, and currently he is a distinguished professor. His awards include the 2007 K. T. Lee Research Award, President's Citation Award (American Biographical Institute) in 2007, Engineering Professor Award of Chinese Institute of Engineers (Kaohsiung Branch) in 2008, National Science Councils Outstanding Research Award in 2009, IEEE Outstanding Technical Achievement Award (IEEE Tainan Section) in 2011, Outstanding Electronic Engineering Professor Award of Chinese Institute of Electrical Engineers in 2013, and Outstanding Engineering Professor Award of Chinese Institute of Engineers in 2014. He is a fellow of the British Computer Society (BCS). He is also an experienced editor who has provided editorial services to a number of journals, including serving as an associate editor of *IEEE Access*, *Theoretical Computer Science* (Elsevier), *Discrete Applied Mathematics* (Elsevier), the *Journal of Supercomputing* (Springer), the *International Journal of Computer Mathematics* (Taylor & Francis Group), *Fundamental Informaticae* (Polish Mathematical Society), the *Journal of Interconnection Networks* (World Scientific), and *Discrete Mathematics Algorithms and Applications* (World Scientific). In addition, he has served on the organization committee and/or program committee of several dozens international conferences in computer science and computer engineering. His current research interests include design and analysis of algorithms, fault-tolerant computing, bioinformatics, parallel and distributed computing, and algorithmic graph theory. He is a senior member of the IEEE.

Yu-Chun Chou received the BS degree from the Department of Computer Science and Information Engineering, National Chung Hsing University University, Taiwan, and the MS degree from Institute of Medical Informatics at National Cheng Kung University, Taiwan, in 2011. His research interests include bioinformatics and computational biology.