Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2008, Article ID 612397, 10 pages
doi:10.1155/2008/612397

Research Article

Detect Key Gene Information in Classification of Microarray Data

Yihui Liu

School of Computer Science and Information Technology, Shandong Institute of Light Industry, Jinan, Shandong 250353, China

Correspondence should be addressed to Yihui Liu, yxl@sdili.edu.cn

Received 10 November 2007; Revised 1 March 2008; Accepted 14 April 2008

Recommended by P.-C. Chung

We detect key information of high-dimensional microarray profiles based on wavelet analysis and genetic algorithm. Firstly, wavelet transform is employed to extract approximation coefficients at 2nd level, which remove noise and reduce dimensionality. Genetic algorithm (GA) is performed to select the optimized features. Experiments are performed on four datasets, and experimental results prove that approximation coefficients are efficient way to characterize the microarray data. Furthermore, in order to detect the key genes in the classification of cancer tissue, we reconstruct the approximation part of gene profiles based on orthogonal approximation coefficients. The significant genes are selected based on reconstructed approximation information using genetic algorithm. Experiments prove that good performance of classification is achieved based on the selected key genes.

1. INTRODUCTION

Recently, hugeadvances in DNA microarrayhave allowed the scientist to test thousands of genes in normal or tumor tissues on a single array and check whether those genes are active, hyperactive, or silent. Therefore, there is an increasing interest in changing the criterion of tumor classification from morphologic to molecular. In this perspective, the problem can be regarded as a classification problem in machine learning. Generally, microarray expression experiments allow the recording of expression levels of thousands of genes simultaneously. These experiments primarily consist of either monitoring each gene multiple times under various conditions [1], or alternately evaluating each gene in a single environment but in different types of tissues, especially for cancerous tissues [2]. Those of the first type have allowed for the identification of functionally related genes due to common expression patterns, while the experiments for the latter have shown a promise in classifying tissue types.

Generally speaking, approaches to classify the microarray data usually use a criterion relating to the correlation degree to rank and select key genes, such as signal-to-noise ratio (SNR) method [3], the partial least squares method [4], Pearson correlation coefficient method [5] and t-test statistic method [6]. Independent component analysis [7] also is used in the analysis of DNA microarray data. To equip the system with the optimum combination of classifier, gene selection, and cross-validation methods, researchers perform a systematic and comprehensive evaluation of several major algorithms [8]. A very promising solution to combine the two ensemble schemes bagging and boosting, called BagBoosting, is proposed in the paper [9]. The predictive potential is confirmed by comparing BagBoosting to several established class prediction tools for microarray data. Li et al. [10] discover many diversified and significant rules from high-dimensional profiling data and propose to aggregate the discriminating power of these rules for reliable predictions. The discovered rules are found to contain low-ranked features; these features are found to be sometimes necessary for classifiers to achieve perfect accuracy. Tan and Gilbert [11] focus on three different supervised machine learning techniques in cancer classification, namely C4.5 decision tree, and bagged and boosted decision trees. They have performed classification tasks on seven publicly-available cancerous microarray data and compared the classification/prediction performance of these methods. They have observed that ensemble learning (bagged and boosted decision trees) often performs better than single decision trees in this classification task. Zhou et al. [12] propose using a mutual information-based feature selection method where features

are wavelet-based. They select Daubechies basis which has four nonzero coefficients of the compact support wavelet orthogonal basis. They use approximation coefficients and wavelet coefficients to perform mutual information-based feature selection. For transformations, a set of new basis is normally chosen for the data. The selection of the new basis determines the properties that will be held by the transformed data. Principle component analysis (PCA) is used to extract the main components from microarray data; linear discriminant analysis (LDA) is used to extract discriminant information from microarray data. Instead of transforming uncorrelated components, like PCA and LDA, independent component analysis (ICA) attempts to achieve statistically independent components in the transform for feature extraction. But all these methods do not detect the localized features of microarray data.

For wavelet transform, the first advantage is that a set of wavelet basis aims to represent the localized features contained in microarray data. Approximation coefficients compress the microarray data and hold the major information of data, not losing time property of data. The transforms, such as PCA, LDA, and ICA, are based on training dataset. When training dataset changes, the new basis is computed based on new training dataset. For wavelet transform it is wavelet basis to represent each sample vector. The second advantage of wavelet transform is that when the training sample vector is deleted, added, or changed, this change does not affect the computation of other sample vectors. The third important advantage of wavelet transform is that the significant genes can be detected based on the reconstruction information of decomposition coefficients at different level. For the transforms of PCA, LDA, and ICA, it is impossible to find the genes based on the reconstruction information because these transforms lose the time property of data.

In this research multilevel wavelet decomposition is performed to break gene profile into approximations and details. Approximation coefficients compress gene profiles and act as the "fingerprint" of microarray data. We use approximation coefficients at 2nd level to characterize the main components and reduce dimensionality. In order to find the significant genes, we reconstruct wavelet approximation coefficients to build the approximation. Experiments are carried out on four datasets, and key genes are detected based on GA features selected from reconstructed approximation.

2. WAVELET ANALYSIS

Wavelet technology is applied widely in many research areas. The wavelet-transform method, proposed by Grossmann and Morlet [13], analyzes a signal by transforming its input time domain into a time-frequency domain. For wavelet analysis for gene expression data, a gene expression profile can be represented as a sum of wavelets at different time shifts and scales using discrete wavelet analysis (DWT). The DWT is capable of extracting the local features by separating the components of gene expression profiles in both time and scale. According to DWT, a time-varying function $f(t) \in L^2(R)$ can be expressed in terms of $\phi(t)$ and $\psi(t)$ as follows:

$$\begin{aligned} f(t) &= \sum_k c_0(k)\phi(t-k) \\ &\quad + \sum_k \sum_{j=1} d_j(k)2^{-j/2}\psi(2^{-j}t-k) \\ &= \sum_k c_{j0}(k)2^{-j0/2}\phi(2^{-j0}t-k) \\ &\quad + \sum_k \sum_{j=j0} d_j(k)2^{-j/2}\psi(2^{-j}t-k), \end{aligned} \tag{1}$$

where $\phi(t)$, $\psi(t)$, c_0, and d_j represent the scaling function, wavelet function, scaling coefficients (approximation coefficients) at scale 0, and detail coefficients at scale j, respectively. The variable k is the translation coefficient for the localization of gene expression data. The scales denote the different (low to high) scale bands.

The wavelet filter-banks approach was developed by Mallat [14]. The wavelet analysis involves two compounds: approximations and details. For one-dimensional wavelet decomposition, starting from signal, the first step produces two sets of coefficients: approximation coefficients (scaling coefficients) c_1, and detail coefficients (wavelet coefficients) d_1. These coefficients are computed by convolving signal with the low-pass filter for approximation, and with the high-pass filter for detail. The convolved coefficients are downsampled by keeping the even indexed elements. Then the approximation coefficients c_1 are split into two parts by using the same algorithm and are replaced by c_2 and d_2, and so on. This decomposition process is repeated until the required level is reached:

$$\begin{aligned} c_{j+1}(k) &= \sum_m h(m-2k)c_j(m), \\ d_{j+1}(k) &= \sum_m h_1(m-2k)c_j(m), \end{aligned} \tag{2}$$

where $h(m-2k)$ and $h_1(m-2k)$ are the low-pass filters and high-pass filters. The coefficient vectors are produced by downsampling and are only half the length of signal or the coefficient vector at the previous level. Conversely, approximations and details are constructed inverting the decomposition step by inserting zeros and convolving the approximation and detail coefficients with the reconstruction filters.

Figure 1 shows wavelet decomposition tree at level 2. Figure 2 shows approximations at 2nd level and details at 2 levels for the sample selected from prostate cancer dataset. In this research we selected approximations coefficients at 2nd level to characterize the main components of microarray data.

The microarray data has high dimensionality and a lot of the information corresponds to genes that do not show any key changes during the experiment [15]. To make it easier to find the significant genes, we remove small change contained in the high frequency part based on wavelet decomposition. If the first levels of the decomposition can be used to eliminate a large part of "small change," the successive approximations appear less and less "noise";

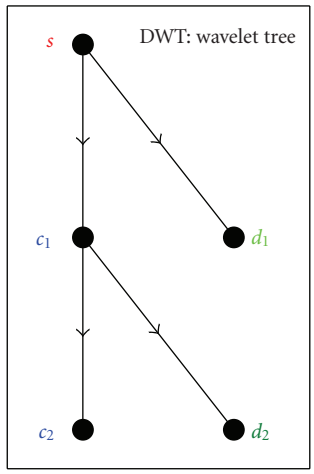

Figure 1: Wavelet decomposition tree at 2 levels. Symbol s represents microarray profiles; c_1 and d_1 represent approximation coefficients and detail coefficients at 1st level; c_2 and d_2 represent approximation coefficients and detail coefficients at 2nd level.

however, they also lose progressively more high-frequency information. In our previousresearch [16, 17], we perform multilevel wavelet decomposition of 4 levels on microarray vector, we got 97.06%, 100%, 94.12%, 94.12% performance using approximation coefficients from first to fourth levels respectively. The experiments prove that the approximation coefficients at 2nd level achieve best results. Figure 3 shows the approximation coefficients at 4 levels, we can see that the coefficient vectors at each level are produced by downsampling and are only half the length of signal or the coefficient vector at the previous level. We perform wavelet decomposition on gene profiles at 2 levels in order to keep major information of microarray data.

Li et al. [18] extract two kinds of features, which are the approximation coefficients of DWT, together with some useful features from the high-frequency coefficients selected by the maximum modulus method at 3rd and 4th level. The combined coefficients are then forwarded to an SVM classifier. For leukemia dataset, they got 93.06% accuracy based on Daubechies basis (db8), and 100% and 97.22% accuracy based on Biorthogonal basis (bior2.6), using the combined features of 3rd level and 4th level. In their research they did not show how to select the key genes based on the combined features.

Figure 4 describes the algorithm based on wavelet features. After wavelet decomposition, 3159 orthogonal wavelet coefficients areobtained based on wavelet decomposition at 2nd level. The transforms of PCA, LDA, and ICA need large matrix computation, because microarray data is of high dimensionality. So a large computation load is needed for the transforms of PCA, LDA, ICA, and so forth. However, wavelet transform uses wavelet basis to represent the each sample vector. Each sample vector is convolved with wavelet filter and then obtained wavelet coefficients are downsampled. Wavelet transform does not involve the large matrix computation and needs small computation load, so it is more practical. Figure 5 shows how to find the significant genes of microarray vector based on wavelet reconstructed information. In order to find the significant genes, we reconstruct approximation based on the decomposed coefficients and reconstructed approximation has the same dimensionality with the original data.

In our previous experiments, for leukemia dataset, 96.72% accuracy of 2 fold cross validation experiments is achieved based on approximation coefficients at 2nd level. We compare our results with other feature extraction methods. In Huang and Zheng's study [7], they reshuffled the dataset randomly. They performed the experiments with 20 random splittings of the original datasets, which means that each randomized training and test set contains the same amount of samples of each class compared with the original training and test set. They concluded the results of different methods, such as least 92.86% of squares support vector machine (LS-SVM), 94.40% of PCA, 93.58% of kernel PCA (KPCA), 94.65% of penalized independent component regression (P-ICR), 93.83% of penalized principal component regression (P-PCR), and nearest shrunken centroid classifier (PAM). Readers can see the details from Huang and Zheng's paper.

3. GENETIC ALGORITHM

The genetic algorithm (GA) is an evolutionary computing technique that can be used to solve problems efficiently for which there are many possible solutions [19]. A potential solution to the problem is encoded as a *chromosome*. Genetic algorithms create a group of chromosomes, called the *population*, to explore the search space. A *fitness function* evaluates the performance of each chromosome. Genetic algorithm is based on the Darwinian principle of evolution through natural selection, which the better individual has higher chance of survival and tends to pass on its favorable traits to its offspring. Thus, chromosomes with higher fitness scores have higher chances of producing offspring.

3.1. Chromosome encoding

In our optimization problems, it is more natural to represent the genes directly as real numbers, which means that there are no differences between the genotype (coding) and the phenotype (search space) [20]. A thorough review related to real-coded genetic algorithms can be seen in [21]. In our research, we perform GA on wavelet features to select the best discriminant features and reduce dimensionality of wavelet feature space further. We define a chromosome C as a vector consisting of m genes x_k, $1 \le k \le m$.

$$C = \{(x_1, \ldots, x_k, \ldots, x_m) \mid 1 \le \forall i \le m : 1 \le x_i \le d_{\max};\ 1 \le i,\ j \le m,\ i \ne j : x_i \ne x_j\}, \quad (3)$$

where $d_{\max}$ is the number of original wavelet features. We select different number of features in our study respectively to evaluate the performance of classification. Firstly, the algorithm creates initial population by ranking key features based on a two-way t-test with pooled variance estimate. The algorithm then creates a sequence of new populations.

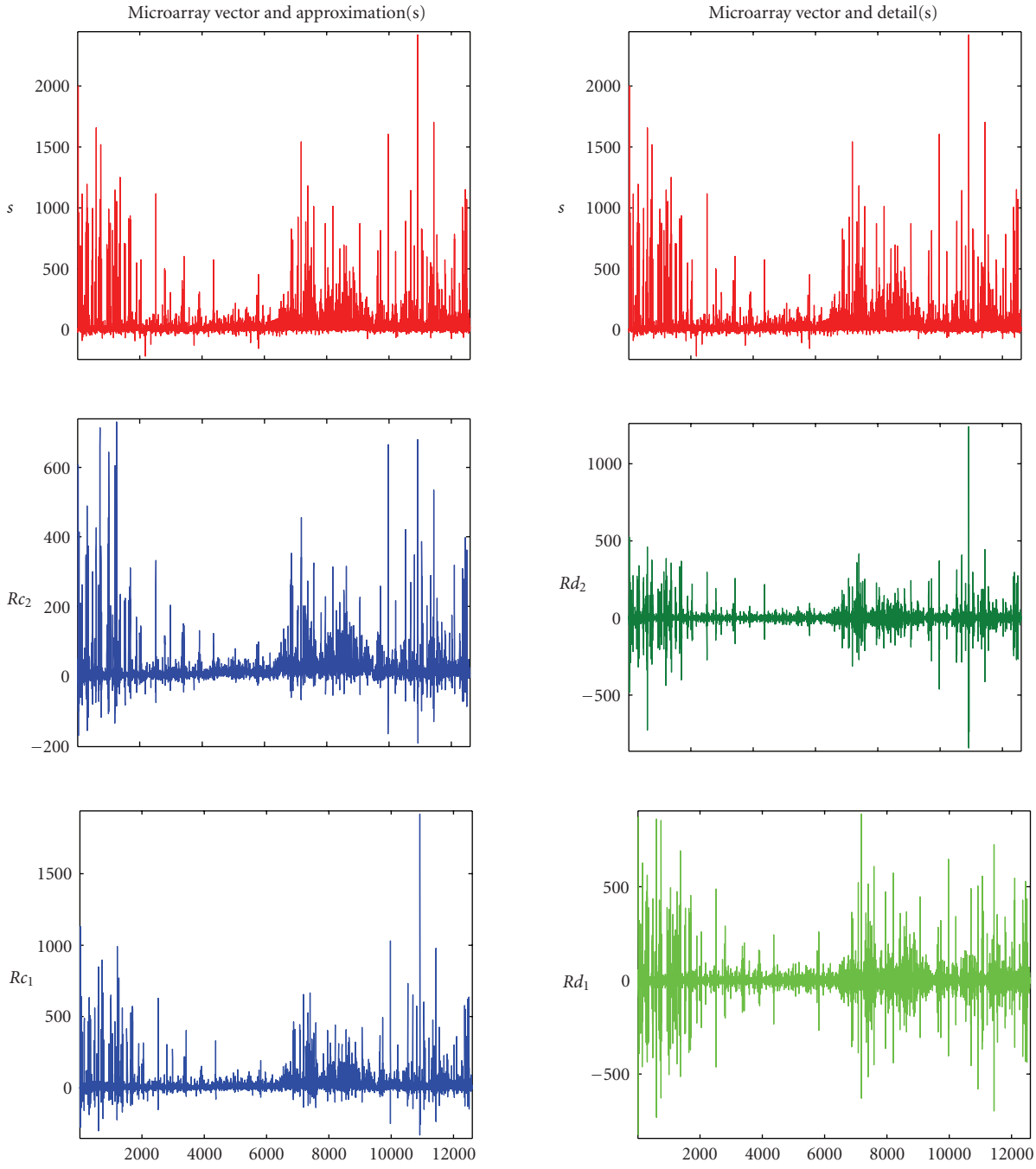

Figure 2: Approximations at 2 levels and details at 2 levels.

At each step, the algorithm uses the individuals in the current generation to create the next population. Each member of the current population is scored by computing its fitness value. The algorithm usually selects individuals that have better fitness values as parents. A fitness function acts as selective pressure on all of the data points. This function determines which data points get passed on to or removed from each subsequent generation. To apply a genetic algorithm on the microarray data, we use LDA classifier as fitness function to evaluate how well the data gets classified.

3.2. Fitness function

LDA is a popular discriminant criterion, which is used to find a linear projection of the original vectors from a high-dimensional space to an optimal low-dimensional subspace in which the ratio of the between-class scatter

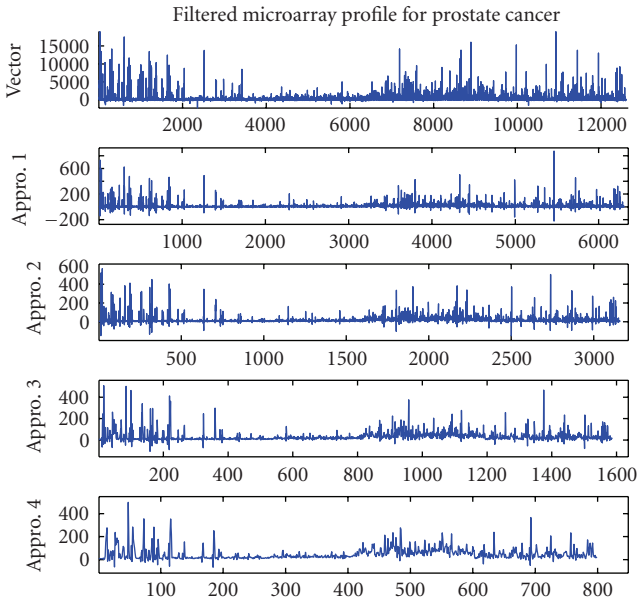

Figure 3: Approximation coefficients at 4 levels.

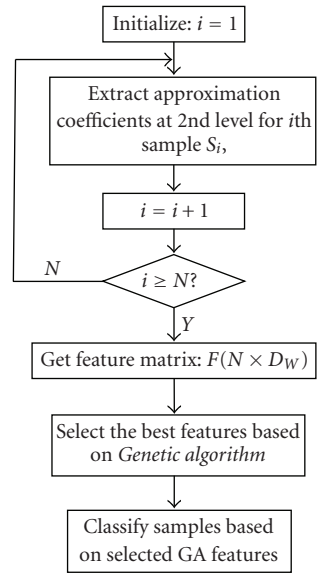

Figure 4: Classification based on wavelet features at 2nd level. N, D_w represent the number of samples and dimension number of wavelet features, respectively

and the within-class scatter is maximized [22]. Let $C_1, C_2, \ldots, C_L$ denote the classes of DNA microarray vector. Let $M_1, M_2, \ldots, M_L$ and M be the means of the classes and the grand mean. The within-and between-class scatter matrices, Σ_w and Σ_B, are defined as follows:

$$\begin{aligned} \Sigma_w &= \sum_{i=1}^{L} P(C_i) E\left\{ \frac{(y - M_i)(y - M_i)^T}{C_i} \right\}, \\ \Sigma_B &= \sum_{i=1}^{L} P(C_i)(M_i - M)(M_i - M)^T, \end{aligned} \tag{4}$$

where $P(C_i)$ is a priori probability, $E(\cdot)$ denotes the expectation operator, and L and y denote the number of classes and sample vector.

LDA derives a projection matrix that maximizes the ratio $|\Psi^T \Sigma_B \Psi| / |\Psi^T \Sigma_W \Psi|$. This ratio is maximized when Ψ consists of the eigenvectors of the matrix $\Sigma_W^{-1} \Sigma_B$:

$$\Sigma_W^{-1} \Sigma_B \Psi = \Psi \Delta, \tag{5}$$

where Ψ, Δ are the eigenvector and eigenvalue matrices of $\Sigma_W^{-1} \Sigma_B$, respectively.

The fitness function to evaluate the performance of DNA microarray data is defined as below:

$$f = 100^* \text{err} + 1 - \text{mean}(P_{\text{poster}}(C_i)), \tag{6}$$

where P_{poster} is a posterior probabilities, and err denotes the error rate.

3.3. Genetic operators

3.3.1. Selection operator

The selection operation is based on the fitness value of chromosomes. Chromosomes have high fitness value to be kept for next generation. In our algorithm, we adopt a roulette wheel selection scheme. Assume the population P has N chromosomes, for each chromosome C_j $(1 \leq j \leq N)$, the selection probability, $p_s(C_j)$, is calculated as

$$p_s(C_j) = \frac{f(C_j)}{\Sigma_{k=1}^{N} f(C_k)}. \tag{7}$$

In roulette wheel selection, a chromosome C_j is selected if a uniformly random number γ in $[0, 1]$ satisfies

$$\sum_{k=0}^{j-1} p_s(C_j) < \gamma \leq \sum_{k=0}^{j} p_s(C_j), \quad \text{where } p_s = 0 \text{ for } k = 0. \tag{8}$$

Elite children, that are the individuals in the current generation with the best fitness values, automatically survive to the next generation. In this research, the number of elite children is set to two.

3.3.2. Crossover operator

Since the real encoding is adopted in this study, the standard crossover operation for the binary encoding method cannot be used. We use a specific crossover operation for our problem. Crossover children are created by combining the vectors of a pair of parents. A gene at the same coordinate from one of the two parents is selected and assigned to the child. First, we create a random binary vector, select the genes where the vector is 1 from the first parent, and the genes where the vector is 0 from the second parent, and combine the genes to form the child. For example, if C1 and C2 are the parents, and the binary vector is [1 1 0 0 1 0 0 0],

$$\begin{aligned} \text{C1} &= \begin{bmatrix} \text{a} & \text{b} & \text{c} & \text{d} & \text{e} & \text{f} & \text{g} & \text{h} \end{bmatrix}, \\ \text{C2} &= \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{bmatrix}. \end{aligned} \tag{9}$$

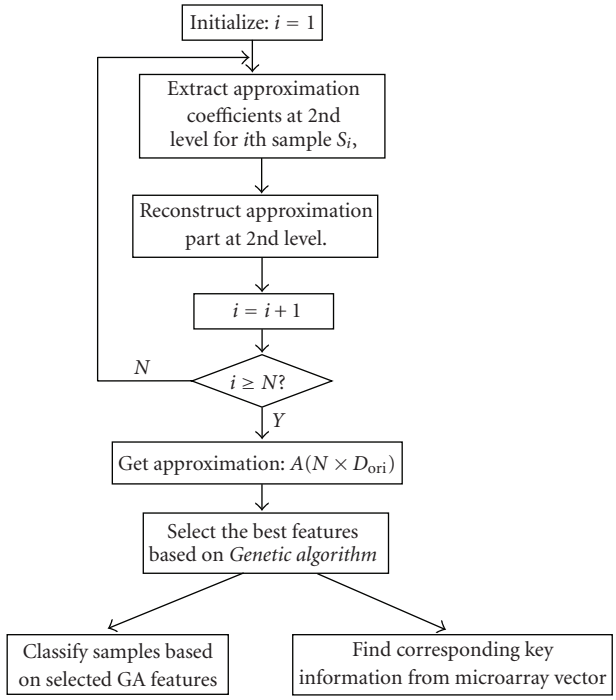

Figure 5: The method of finding significant information of microarray vector based on wavelet reconstructed information. N, D_{ori} represent the number of samples and original dimension number of microarray vectors, respectively

The crossover results are the following child:

$$\text{Child} = \begin{bmatrix} \text{a} & \text{b} & 3 & 4 & \text{e} & 6 & 7 & 8 \end{bmatrix}. \tag{10}$$

The crossover fraction, which specifies the fraction of each population besides elite children, is set to 0.8.

3.3.3. Mutation operator

The mutation algorithm creates mutation children by randomly changing the genes of individual parents. In this study the algorithm adds a random vector from a Gaussian distribution to the parent.

Gaussian mutation.

It is defined as follows:

$$\sigma_j = k \cdot \text{Min}\{x_j^{t-1} - a_j, b_j - x_j^{t-1}\} \left(1 - \frac{t}{M_g}\right)^s \quad (j = 1, 2, \ldots, N), \tag{11}$$

where k is a constant within the closed interval [0, 1]; t is the generation; x_j^{t-1} is the jth variable to be optimized in the $(t-1)$th generation; $[a_j, b_j]$ is the jth variable's scope; M_g is the maximum generation; s is a shape parameter; and N is the number of variables.

The mutation of the jth variable, x_j', is expressed as

$$x_j' = x_j + \varepsilon_j \quad (j = 1, 2, \ldots, N), \qquad \varepsilon_j \sim N(0, \sigma_j), \tag{12}$$

where ε_j is distributed as a Gaussian random variable with mean zero and standard deviation ε_j.

The algorithm stops when one of the stopping criteria is met. GA uses four different criteria to determine when to stop the solver. GA stops when the maximum number of generations is reached; the maximum number of generations is set to 70 in this research. Fitness limit is considered and the algorithm stops if the best fitness value is less than or equal to the value of fitness limit. GA also detects if there is no change in the best fitness value for some time given in seconds (stall time limit = 20), or for some number of generations (stall generation limit = 50).

In the computer (Intel Pentium processor 1.73 GHz, 512 MB) and MATLAB run environments, for prostate cancer dataset, it took 12 seconds to do wavelet decomposition and reconstruction, and about 19 minutes to run 10 times GA for selecting the best features varying from 2 features to 11 features based on the reconstructed approximation. The average time for running one time GA is nearly 2 minutes. When we perform 10 times GA on approximation coefficients to select the best features varying from 2 features to 11 features, it took about 12 minutes, which is much quicker than on the reconstructed approximation. This is because approximation coefficients at 2nd level only have 3159 dimensions and original data has 12 600 dimensions. After dimensionality reduction, the computation load is reduced.

4. EXPERIMENTS

In this study we use correct rate, sensitivity, specificity, positive predictive value (PPV), and negative predictive value (NPV) to evaluate the performance. Let TP, TN, FP, and FN stand for the number of true positive (cancer), true negative (control), false positive, and false negative samples, respectively. Sensitivity is defined as $TP/(TP+FN)$; specificity is defined as $TN/(TN+FP)$; PPV is defined as $TP/(TP+FP)$; NPV is defined as $TN/(TN+FN)$; correct rate is defined as $(TP+TN)/(TP+TN+FP+FN)$. Firstly, we do the preprocessing on microarray profiles by filtering gene profile vectors with 0 profile variance over time. After filtering, we extract wavelet approximation coefficients from the filtered data to remove noise. Firstly, approximation coefficients at 2nd level are selected to reduce dimensionality, remove noise hidden in microarray data. Then genetic algorithm is implemented to optimize the wavelet approximation coefficients and to evaluate the performance of classification. Here we use Daubechies basis (db7) [23] for wavelet analysis of DNA microarray data, which has seven nonzero coefficients of the compact support wavelet orthogonal basis. Secondly, in order to find the significant microarray information, we reconstruct the approximation coefficients to build the approximation at 2nd level. Then genetic algorithm is performed to find the key features based on the reconstruction information, and the corresponding key genes are identified based on selected reconstructed information. We set different feature number in GA to find the best performance.

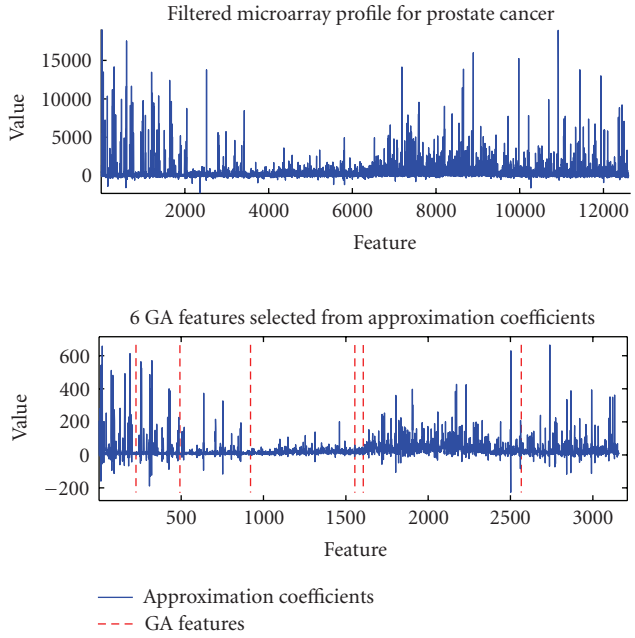

Figure 6: Approximation coefficients at 2nd level and selected GA features.

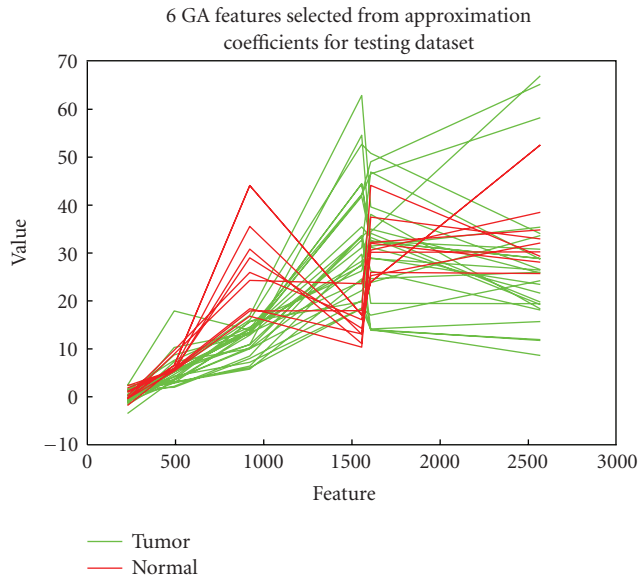

Figure 7: Selected approximation coefficients for test samples of prostate cancer dataset.

Prostate cancer

Prostate cancer data [24] contains training set of 52 prostate tumor samples and 50 nontumor (labeled as "Normal") prostate samples with 12 600 genes. An independent set of test samples is also prepared, which is from a different experiment. The test set has 25 tumor and 9 normal samples.

3159 approximation coefficients are obtained based on wavelet decomposition at 2nd level. Genetic algorithm is performed to select the 6 optimized features from approximation coefficients and 97.06% recognition rate is achieved. Figure 6 shows the 6 selected approximation coefficients, and Figure 7 shows 6 selected approximation coefficients for test samples of prostate cancer. Then we reconstruct the approximation at 2nd level based on 3159 orthogonal approximation coefficients. After genetic algorithm is implemented, 7 optimization features are obtained from approximation part and 97.06% accuracy is achieved. Figure 8 shows the 7 selected features from reconstructed approximation at 2nd level and Figure 9 shows 7 selected features for test samples of prostate cancer dataset. Table 1 shows the performance of 6 selected GA coefficients and 7 reconstructed GA features, which are corresponding with 7 key genes ("32789_at," "34728_g_at," "36310_at," "36623_at," "37329_at," "37640_at," "38100_at"). Figure 10 shows 7 significant genes for test samples of prostate cancer dataset. Table 2 shows that SingleC4.5, BaggingC4.5 and AdaBoostC4.5 methods achieve 67.65%, 75.53%, and 67.65% accuracy, which is inferior to our method.

Lung cancer

Lung cancer data [25] contains two kinds of tissue including malignant pleural mesothelioma (MPM) and adenocarcinoma (ADCA) of the lung. There are 181 tissue samples (31 MPM and 150 ADCA) including 32 training samples (16 MPM and 16 ADCA) and 149 test samples (15 MPM and 134 ADCA). The number of genes of each sample is 12 533.

After wavelet decomposition at 2nd level is performed, 3142 approximation coefficients are extracted. Genetic algorithm performs further dimensionalityreductionand selects the 5 optimized features from approximation coefficients. 98.66% accuracy is achieved. Then we reconstruct the approximation part. 20 GA features selected from reconstructed approximation achieve the 97.99% performance, which are corresponding to 20 significant

Table 1: Performance of selected GA features. This table shows the performance for prostate cancer dataset. The two experiments are based on 6 selected GA features from approximation coefficients at 2nd level and 7 selected GA features from reconstructed approximation at 2nd level. PPV stands for positive predictive value; NPV stands for negative predictive value.

GA feature number	Correct rate	Sensitivity	Specificity	PPV	NPV
6 (coefficients)	0.9706	1.0000	0.9600	0.9000	1.0000
7 (reconstructed)	0.9706	1.0000	0.9600	0.9000	1.0000

Table 2: Predictive accuracy of the classifiers [11].

Dataset	Accuracy		
	SingleC4.5	BaggingC4.5	AdaBoostC4.5
Leukemia (ALL versus AML)	91.18	91.18	91.18
Lung cancer	92.62	93.29	92.62
Prostate cancer	67.65	73.53	67.65

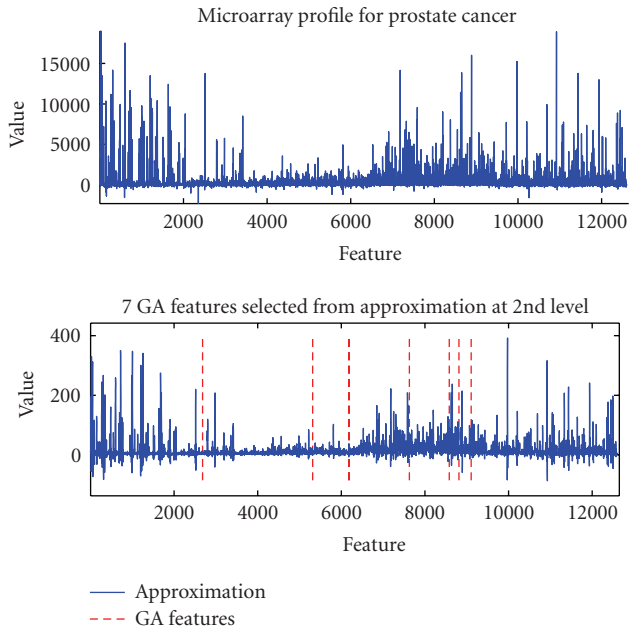

Figure 8: Reconstructed approximation at 2nd level and selected GA features.

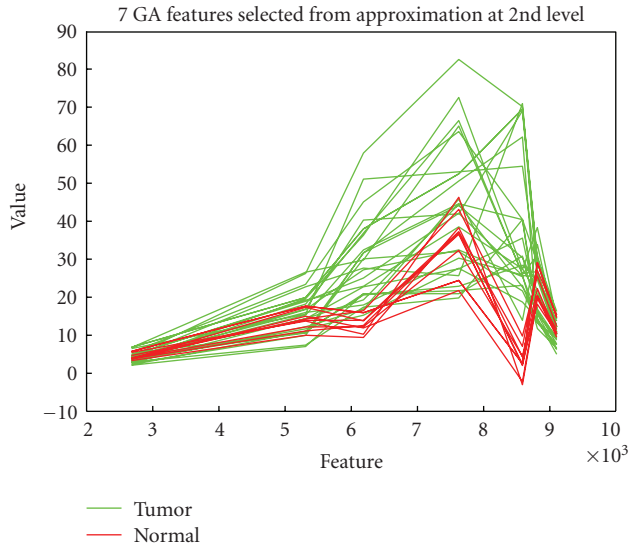

Figure 9: Selected features from reconstructed approximation for test samples of prostate cancer dataset.

genes ("1466_s_at," "31532_at," "32124_at," "32796_f_at," "33276_at," "33420_g_at," "34094_i_at," "36539_at," "36577_at," "37950_at," "38161_at," "38640_at," "38902_r_at," "40142_at," "40289_at," "40526_at," "40935_at," "557_s_at," "781_at," "894_g_at"). Table 3 shows the performance of 5 selected GA coefficients and 20 reconstructed GA feature. Tables 2 and 4 show the performance of other methods. Table 2 shows that SingleC4.5, BaggingC4.5, and AdaBoostC4.5 methods achieve 92.62%, 93.29%, and 92.62% accuracy, which is inferior to our method. Our

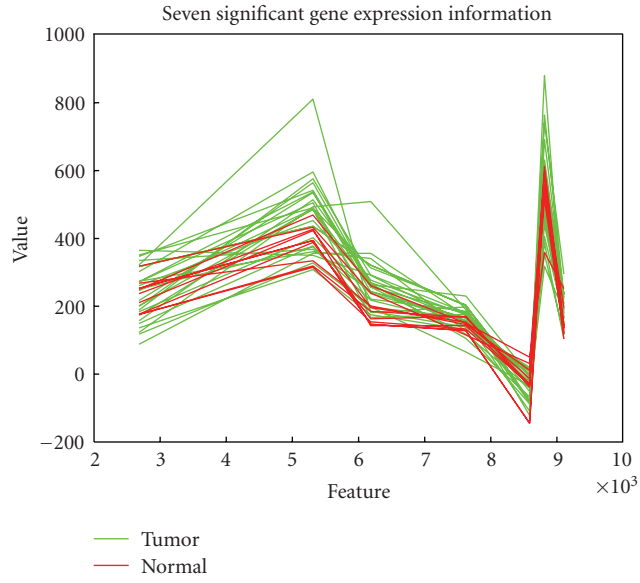

Figure 10: 97.06% performance of selected genes for test samples of prostate cancer dataset.

Table 3: Performance of selected GA features. This table shows the performance for lung cancer dataset. PPV stands for positive predictive value; NPV stands for negative predictive value.

GA feature number	Correct rate	Sensitivity	Specificity	PPV	NPV
5 (coefficients)	0.9866	0.9851	1.0000	1.0000	0.8824
20 (reconstructed)	0.9799	0.9851	0.9333	0.9925	0.8750

Table 4: Test error numbers of four models [10].

	Test error numbers (MPM : ADCA)			
Dataset	Li's method	SingleC4.5	BaggingC4.5	BoostingC4.5
Lung cancer	3(1 : 2)	27(4 : 23)	4(0 : 4)	27(4 : 23)

best performance 98.66% is also better than 97.99% of Li's method.

Leukemia (ALL versus AML)

Training dataset consists of 38 bone marrow samples (27 ALL and 11 AML) with 7129 attributes from 6817 human genes, and 34 test samples including 20 ALL and 14 AML [3].

After wavelet decomposition at 2nd level is performed on geneprofile,we obtain 1791 approximation coefficients. Genetic algorithm is used to select the optimized features from approximation coefficients of 38 × 1791 training matrix. When 15 GA selected features from approximation coefficients are obtained, 100% correct rate is achieved. After we reconstruct the approximation at 2nd level based on approximation coefficients, 4 GA features selected from reconstructed approximation achieve the 97.06%

Table 5: Performance of selected GA features. This table shows performance of leukemia (ALL versus AML) dataset. PPV stands for positive predictive value; NPV stands for negative predictive value.

GA feature number	Correct rate	Sensitivity	Specificity	PPV	NPV
15 (coefficients)	1.0000	1.0000	1.0000	1.0000	1.0000
4 (recontructed)	0.9706	0.9286	1.0000	1.0000	0.9524

Table 6: The test error numbers by four classification models [10].

	Test error numbers			
Dataset	Li's method	C4.5	Bagging	Boosting
MLL-leukemia	0	4(2 : 2 : 0)	2(1 : 1 : 0)	0

performance, which are corresponding to 4 significant genes ("attribute1773:," "attribute4620:," "attribute4846:", "attribute5124:"). Table 5 shows the performance of 15 selected GA coefficients and 4 reconstructed GA feature. Our best result is better than 97.06% of Bayesian variable method [26], 82.3% of the PCA disjoint models [27], and 88.2% of the between-group analysis [28]. Also Table 2 shows SingleC4.5, BaggingC4.5, and AdaBoostC4.5 methods achieve 91.18% accuracy, which is inferior to our method.

MLL-leukemia (ALL versus MLL versus AML)

Leukemia data [29] contains 57 training leukemia samples (20 ALL, 17 MLL, and 20 AML). Test data contains 4 ALL, 3 MLL, and 8 AML samples. The number of attributes is 12 582.

After wavelet decomposition at 2nd level, 3155 approximation coefficients act as the "fingerprint" of microarray data. When 16 GA features are selected based on training matrix 57×3155, 100% correct rate is achieved. After we reconstruct the approximation at 2nd level based on approximation coefficients, 7 GA features selected from reconstructed approximation achieve the 100% performance, which are corresponding to 7 significant genes ("32556_at," "33415_at," "33725_at," "34775_at," "36122_at," "36340_at," "40578_s_at"). We have the same performance with Li's method [10], boosting method, and better than C4.5, Bagging methods, which are shown in Table 6.

5. CONCLUSIONS

In this paper, we propose a hybrid method to find significant genes based on wavelet analysis and GA. We use approximation coefficients at 2nd level to remove noise and characterize the main features of gene profiles. Genetic algorithm is further implemented to select the optimal features from approximation coefficients. Experiments are carried out on four independent datasets based on selected GA features and good performance is achieved compared to the other research methods. Furthermore, we reconstruct the approximation information based on the orthogonal approximation coefficients at 2nd level, and significant genes are selected based on the reconstruction information.

ACKNOWLEDGMENTS

This study is supported by research funds of Shandong Institute of Light Industry (12041653), and by International Collaboration Project of Shandong Province Education Department, China.

REFERENCES

[1] C. J. Roberts, B. Nelson, M. J. Marton, et al., "Signaling and circuitry of multiple MAPK pathways revealed by a matrix of global gene expression profiles," *Science*, vol. 287, no. 5454, pp. 873–880, 2000.

[2] L. Zhang, W. Zhou, V. E. Velculescu, et al., "Gene expression profiles in normal and cancer cells," *Science*, vol. 276, no. 5316, pp. 1268–1272, 1997.

[3] T. R. Golub, D. K. Slonim, P. Tamayo, et al., "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, no. 5439, pp. 531–527, 1999.

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

[5] M. Xiong, L. Jin, W. Li, and E. Boerwinkle, "Computational methods for gene expression-based tumor classification," *BioTechniques*, vol. 29, no. 6, pp. 1264–1268, 2000.

[6] P. Baldi and A. D. Long, "A Bayesian framework for the analysis of microarray expression data: regularized t-test and statistical inferences of gene changes," *Bioinformatics*, vol. 17, no. 6, pp. 509–519, 2001.

[7] 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.

[8] A. Statnikov, C. F. Aliferis, I. Tsamardinos, D. Hardin, and S. Levy, "A comprehensive valuation of multicategory classification methods for microarray gene expression cancer diagnosis," *Bioinformatics*, vol. 21, no. 5, pp. 631–643, 2005.

[9] M. Dettling, "BagBoosting for tumor classification with gene expression data," *Bioinformatics*, vol. 20, no. 18, pp. 3583–3593, 2004.

[10] J. Li, H. Liu, S.-K. Ng, and L. Wong, "Discovery of significant rules for classifying cancer diagnosis data," *Bioinformatics*, vol. 19, supplement 2, pp. 93–102, 2003.

[11] A. C. Tan and D. Gilbert, "Ensemble machine learning on gene expression data for cancer classification," *Applied Bioinformatics*, vol. 2, supplement 3, pp. S75–83, 2003.

[12] X. Zhou, X. Wang, and E. R. Dougherty, "Nonlinear probit gene classification using mutual information and wavelet-based feature selection," *Journal of Biological Systems*, vol. 12, no. 3, pp. 371–386, 2004.

[13] A. Grossmann and J. Morlet, "Decomposition of Hardy functions into square integrable wavelets of constant shape," *SIAM Journal on Mathematical Analysis*, vol. 15, no. 4, pp. 723–736, 1984.

[14] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.

[15] I. S. Kohane, A. T. Kho, and A. J. Butte, *Microarrays for an Integrative Genomics*, MIT Press, Cambridge, Mass, USA, 2003.

[16] Y. Liu, "Wavelet feature selection for microarray data," in *Proceedings of the IEEE/NIH on Life Science Systems and Applications Workshop (LISA '07)*, pp. 205–208, Bethesda, Md, USA, November 2007.

[17] Y. Liu, "Feature extraction for DNA microarray data," in *Proceedings of the 20th IEEE Symposium on Computer-Based Medical Systems (CBMS '07)*, pp. 371–376, Maribor, Slovenia, June 2007.

[18] S. Li, C. Liao, and J. T. Kwok, "Wavelet-based feature extraction for microarray data classification," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '06)*, pp. 5028–5033, Vancouver, Canada, July 2006.

[19] J. H. Holland, *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge, Mass, USA, 1992.

[20] A. Blanco, M. Delgado, and M. C. Pegalajar, "A real-coded genetic algorithm for training recurrent neural networks," *Neural Networks*, vol. 14, no. 1, pp. 93–105, 2001.

[21] F. Herrera, M. Lozano, and J. L. Verdegay, "Tackling real-coded genetic algorithms: operators and tools for behavioural analysis," *Artificial Intelligence Review*, vol. 12, no. 4, pp. 265–319, 1998.

[22] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, New York, NY, USA, 2nd edition, 1991.

[23] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Communications on Pure and Applied Mathematics*, vol. 41, no. 7, pp. 909–996, 1988.

[24] D. Singh, P. G. Febbo, K. Ross, et al., "Gene expression correlates of clinical prostate cancer behavior," *Cancer Cell*, vol. 1, no. 2, pp. 203–209, 2002.

[25] G. J. Gordon, R. V. Jensen, L.-L. Hsiao, et al., "Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma," *Cancer Research*, vol. 62, no. 17, pp. 4963–4967, 2002.

[26] K. E. Lee, N. Sha, E. R. Dougherty, M. Vannucci, and B. K. Mallick, "Gene selection: a Bayesian variable selection approach," *Bioinformatics*, vol. 19, no. 1, pp. 90–97, 2003.

[27] S. Bicciato, A. Luchini, and C. Di Bello, "PCA disjoint models for multiclass cancer analysis using gene expression data," *Bioinformatics*, vol. 19, no. 5, pp. 571–578, 2003.

[28] A. C. Culhane, G. Perrière, E. C. Considine, T. G. Cotter, and D. G. Higgins, "Between group analysis of microarray data," *Bioinformatics*, vol. 18, no. 12, pp. 1600–1608, 2002.

[29] S. A. Armstrong, J. E. Staunton, L. B. Silverman, et al., "*MLL* translocations specify a distinct gene expression profile that distinguishes a unique leukemia," *Nature Genetics*, vol. 30, no. 1, pp. 41–47, 2002.