Classification of Two and Multi-Class Cancer Data Reliably Using Multi-Objective Evolutionary Algorithms

Kalyanmoy Deb and A. Raji Reddy
Kanpur Genetic Algorithms Laboratory (KanGAL)
Indian Institute of Technology Kanpur
Kanpur, PIN 208016, India
{deb,arreddy}@iitk.ac.in

KanGAL Report Number 2003006

ABSTRACT

Motivation: In the area of bioinformatics, the identification of gene subsets responsible for classifying available samples to two or more classes (such as 'malignant' or 'benign') based on the gene microarray data is an important task. The main challenges are the availability of only a few disease samples compared to the number of genes in samples and the exorbitantly large search space of solutions to search from. Also, in such problems many different gene combinations may provide similar classification accuracy. Thus, researchers are motivated to find a reliable gene classifier which is small in size and capable of producing as accurate a classification as possible.

Methods: Although there exist a number of studies which use an evolutionary algorithm (EA) for this task, here we treat the problem as a multi-objective optimization problem of minimizing the classifier size and simultaneously minimizing the number of misclassified instances in training and test samples. The standard weighted voting method is used to design a unified procedure for handling two and multi-class problems. The reliability in the classification process is ensured by using a prediction strength concept. The use of multi-objective EAs here is also unique in finding multiple high-performing classifiers in a single simulation run.

Results: Contrary to the past studies, the use of a multi-objective EA (non-dominated sorting GA or NSGA-II) has enabled us to discover a much smaller gene subset size to correctly classify 100% or near 100% samples for three two-class cancer datasets (Leukemia, Lymphoma, and Colon) and two well-studied multi-class datasets (GCM and NCI60). In all cases, the classification accuracy is more than that reported earlier. Moreover, an analysis of the multiple high-performing classifiers reveals important commonly-appearing gene combinations which should be of immediate importance to biologists. The flexibility and effectiveness of the proposed multi-objective EAs in tackling the classification task on two, nine, and 14 class samples amply demonstrate further and immediate use of the technique to other more complex classification problems.

Availability: Supplementary information, source codes, and results on other variations of the classification problem are available at http://www.iitk.ac.in/bioinformatics.
Contact: deb@iitk.ac.in

INTRODUCTION

The recent technological improvements in DNA microarray experiments enable to monitor the *expression levels* of thousands of genes simultaneously. Since the cancer cells generally evolve from normal cells due to mutations in genomic DNA, comparison of gene expression profiles from cancerous and normal tissues (e.g. in case of Leukemia cancer (Golub et al., 1999), ALL versus AML), or from tissues corresponding to different cancer types (e.g. in case of NCI60 dataset (Ross et al., 2000), breast, CNS, melanoma, ovarian, renal etc.) can provide useful insights into genes implicated in different cancer types. For this purpose, different machine learning approaches such as supervised and some unsupervised learning have been previously investigated with varying degrees of success (Alizadeh et al., 2000; Alon et al., 1999; Golub et al., 1999). However, the important issue in such problems is the availability of only a few samples compared to the large number of genes, a matter which makes the classification task difficult. Furthermore, many of the genes in a dataset are not relevant to the distinction between different tissue types and introduce noise in the classification process. Therefore, identification of small set of genes, sufficient to distinguish between different tissue types under investigation is one of the crucial tasks from cancer diagnostics point of view. This paper goes further in this direction and focuses on the topic for identification of smallest set of informative genes for reliable classification of cancer samples to two or more classes solely based on their expression levels.

The primary objective of the gene subset identification task is to find a classifier with minimum number of genes providing maximum classification accuracy. However, the optimality of a gene subset for a fixed training set can only be guaranteed by an exhaustive search over all pos-

sible combinations of gene subsets. In practice, this is computationally infeasible owing to the large number of possible gene combinations or subsets. Several statistical and analytical approaches have been developed to identify key-predictive genes for classification purpose. But many of these studies (Alizadeh et al., 2000; Fridlyand et al., 2002; Golub et al., 1999; Ramaswamy et al., 2001) have used simple gene-ranking techniques or correlation matrices, where a set of genes are ranked based on their expression patterns in a set of training samples and a fixed number of top-ranked genes are selected to construct the classifier. Therefore, these ranking-based techniques select the genes which individually provide better classification, but they may not result in meaningful gene combinations for an overall classification task. Hence, approaches which are capable of performing an efficient search in high dimensional spaces, such as evolutionary algorithms (EAs), should prove to be ideal candidates.

The gene subset identification problem is truly a multi-objective optimization problem (MOOP) consisting of a number of objectives (Liu et al., 2001; Liu and Iba, 2002). Although the optimization problem is multi-objective, all of these previous studies have scalarized multiple objectives into one. In this paper, we have used a multi-objective evolutionary algorithm (MOEA) to find the optimum gene subset in a number well-studied datasets (Leukemia, Lymphoma, Colon, NCI60, and GCM tumor data). By using three objectives for minimization (gene subset size, number of misclassifications in training samples, and number of misclassifications in test samples) several variants of a particular MOEA (modified non-dominated sorting genetic algorithm or NSGA-II) are applied to investigate if gene subsets exist with 100% correct classifications in both training and test samples. Since the gene subset identification problem may involve multiple gene subsets of the same size causing identical number of misclassifications (Kohavi and John, 1997), in this paper, we use a novel multi-modal NSGA-II for finding multiple gene subsets simultaneously in one single simulation run. One other important matter in the gene subset identification problem is the confidence level with which the samples are being classified. We introduce the classification procedure based on the prediction strength consideration, suggested in (Golub et al., 1999).

CLASS PREDICTION PROCEDURE

For the identification task, we begin with the gene expression values available for cancer disease samples obtained from the DNA microarray experiments. In addition to the gene expression values, each sample in the dataset is also labeled to belong to one class or the other. For identifying genes responsible for the classification of different tumor types, the available dataset is divided into two groups: one used for the training purpose and the other used for the testing purpose. Here, we use a leave-one-out-cross-validation (LOOCV) procedure to estimate the number of class prediction mismatches in the training samples (τ_{train}), in which one sample is excluded from the training set, and rest of the training samples are used to build the classifier. The classifier is used to predict the class of the left-out sample, and the same procedure is repeated for all training samples. Thereafter, we construct a classifier using all training samples and is used to predict the class of independent test samples. We call τ_{test} as the number of class prediction mismatches in test samples. The class prediction procedure based on weighted voting (WV) approach (Golub et al., 1999; Ramaswamy et al., 2001) is described in the following.

Two-Class Classification

For a given gene subset G, we can predict the class of any sample x (whether belonging to A or B) with respect to a known set of S samples in the following manner. Let us say that S samples are composed of two subsets S_A and S_B, belonging to class A and B, respectively. First, for each gene $g \in G$, we calculate the mean μ_A^g and standard deviation σ_A^g of the normalized gene expression levels $\bar{x}_g$ of all S_A samples. The same procedure is repeated for the class B samples and μ_B^g and σ_B^g are computed. Thereafter, we determine the class of the sample x in the following manner (Golub et al., 1999):

$$\text{class}(x) = \text{sign}\left\{\sum_{g\in G}\left(\frac{\mu_A^g - \mu_B^g}{\sigma_A^g + \sigma_B^g}\right)\left(\bar{x}_g - \frac{\mu_A^g + \mu_B^g}{2}\right)\right\}, \quad (1)$$

If the right term of the above equation is positive, the sample belongs to class A and if it is negative, it belongs to class B. One of the difficulties with the above classification procedure is that the sign of the right term in equation 1 is checked to identify if a sample belongs to one class or another. For each (g) of the 50 genes in a particular Leukemia sample x, we have calculated the statistic $S(x, g)$ (the term inside the summation in equation 1). The statistic $S(x, g)$ values are plotted in Figure 1 for each gene g. Although 27 genes cause negative values of $S(x, g)$ (thereby classifying individually that the sample belongs to AML) and the rest 23 genes detects the sample to be ALL, equation 1 finds the right side value of equation 1 to be 0.01, thereby correctly classifying the sample to be an ALL sample. But it has been argued elsewhere (Golub et al., 1999) that a correct prediction with such a small strength does not make the classification with any reasonable confidence.

For a more confident classification, we may fix a prediction strength threshold θ (> 0) and modify the classification procedure slightly. Let us say that the sum of the positive $S(x, g)$ values is S^A and the sum of the negative $S(x, g)$ values is S^B. Then, the prediction strength, as defined in (Golub et al., 1999) as $|(S^A - S^B)/(S^A + S^B)|$, is compared with θ. If it is more than θ, the classification is accepted, else the sample is considered to be undetermined for class A or B. Here, we assume these undetermined samples to be identical to mismatched samples and

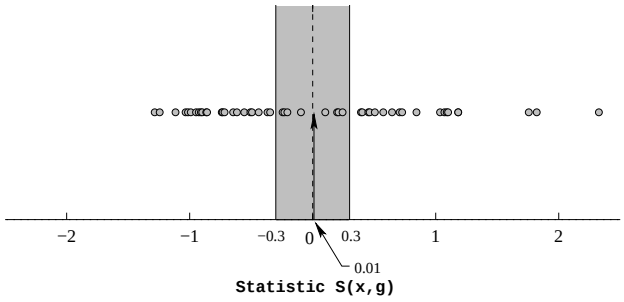

Figure 1: The statistic $S(x, g)$ of a sample for the 50-gene Leukemia data.

include this sample to increment τ_{train} or τ_{test}, as the case may be. This way, a 100% correctly classified gene subset will ensure that the prediction strength is outside $(-\theta, \theta)$. Figure 1 illustrates this concept with $\theta = 0.3$ (30% threshold) and demands that a match will be scored only when the prediction strength falls outside the shaded area.

Multi-Class Classification

For predicting the class of a sample belonging to more than two-classes, we use the above WV approach in conjunction with a one-versus-all (OVA) binary pair-wise classification procedure (Allwein et al., 2000; Bose and Ray-Chaudhari, 1960; Ramaswamy et al., 2001).

For a given gene subset G, the class of a new sample x is predicted in the following manner. Let us say there are total J different tumor types in a dataset such that these S samples are composed of J different subsets S_1,S_2,...,S_j,...,S_J, belonging to class 1, class 2,..., class j,..., class J, respectively. The class of x is predicted by assuming that the sample x belongs to one particular class j ($j \in J$). For each gene $g \in G$, we first calculate the mean μ_j^g and the standard deviation σ_j^g of the normalized expression values of all S_j samples. Similarly, calculate the mean $\mu_{\bar{j}}^g$ and the standard deviation $\sigma_{\bar{j}}^g$ of the remaining $J-1$ class of samples (S_1,S_2,...,S_{j-1}, S_{j+1},...,S_J) together. Thereafter, the class of sample x can be predicted with slight modification of the above Equation 1 in the following manner.

$$\text{class}(x_j) = \text{sign}\left\{\sum_{g \in G}\left(\frac{\mu_j^g - \mu_{\bar{j}}^g}{\sigma_j^g + \sigma_{\bar{j}}^g}\right)\left(\bar{x}_g - \frac{\mu_j^g + \mu_{\bar{j}}^g}{2}\right)\right\}, \quad (2)$$

If the sign of the above equation is positive the sample x belongs to class j, and if the sign is negative the sample x does not belong to that particular class. The above procedure is repeated for all j's (j = 1,2,...,J). Hence, the multi-class classification problem can be thought of as decomposition of a series of J one-versus-all (OVA) binary classification problems.

One of the difficulties with the above classification procedure is that the sign of the right term of the above equation can be positive for more than one classes while predicting the class of a sample x, which ultimately assigns more than one classes for only one sample. One way to overcome this difficulty is that the sample x is assigned to the class with the maximum value of $class(x_j)$ (out of all j's). In this paper, we have used a more stringent classification procedure for generating a generic classifier for reliable classification of multi-class cancer data. For a sample, we calculate the prediction strength (ps) (Ramaswamy et al., 2001) of its belonging to each class j and then find the maximum (ps_{max}) of them. If ps_{max} is greater than θ (the chosen threshold), then assign sample x to that particular class, else the sample is added to the mismatch counters τ_{train} or τ_{test} as the case may be. Thereafter, the predicted class of a sample is compared with its actual class. If the predicted class matches with the actual class, then the sample is said to be correctly-classified, else the sample is added to the mismatch counter.

EVOLUTIONARY GENE SELECTION PROCEDURE

Resulting Optimization Problem

One of the objectives of the classification task is to identify the smallest size of a gene subset for predicting the class of all samples correctly. Although not obvious, when a too small gene subset is used, the classification procedure becomes erroneous. Thus, minimizations of class prediction mismatches in the training and test samples are also important objectives. Here, we use these three objectives in a multi-objective optimization problem: The first objective f_1 is to minimize the size of gene subset in the classifier. The second objective f_2 is to minimize the number of mismatches in the training samples calculated using the LOOCV procedure described earlier and is equal to τ_{train} described above. The third objective f_3 is to minimize the number of mismatches τ_{test} in the test samples.

Solution Procedure Using Evolutionary Algorithms

Like in a previous study (Liu and Iba, 2002), we use a ℓ-bit binary string (where ℓ is the number of genes in a dataset) to represent a solution. For a particular string, the positions marked with a `1` are included in the gene subset for that solution. For example, in the following example of a 10-bit string (representing a total of 10 genes in a dataset), first, third, and sixth genes are considered in the gene subset (also called the *classifier*):

```
(1 0 1 0 0 1 0 0 0 0)
```

The procedure of evaluating a string is as follows. We first collect all genes for which there is a `1` in the string in a gene subset G. Thereafter, we calculate f_1, f_2, and f_3 as described above as three objective values associated with the string. Unlike classical methods, an EA uses a number

(population) of solutions in each iteration. We initialize each population member by randomly choosing at most 10% of string positions to have a 1. Since the gene subset size is to be minimized, this biasing for 1 in a string allows an EA to start with good population members.

To handle three objectives, we have used a modified non-dominated sorting genetic algorithm or NSGA-II (Deb et al., 2002), which we briefly describe here. NSGA-II has the following features:

1. It uses an elitist principle, meaning that best solution(s) from previous iteration is(are) always included in current iteration.

2. It uses an explicit diversity preserving mechanism, thereby allowing to maintain multiple trade-off solutions in an iteration.

3. It emphasizes the *non-dominated* solutions, thereby ensuring convergence close to the *Pareto-optimal* solutions.

In NSGA-II, an offspring population Q_t (of size N) is first created from the parent population P_t (of size N) by using usual genetic operators (such as single-point crossover and bit-wise mutation operators (Goldberg, 1989)). Thereafter, the two populations are combined together to form R_t of size $2N$. First, all duplicate solutions are deleted from R_t. Then, a non-dominated sorting procedure (Deb, 2001) is used to classify the entire population R_t according to increasing order of dominance. Once the non-dominated sorting is over, the new parent population P_{t+1} is created by choosing solutions of different non-dominated fronts from the top of the list in R_t, one at a time. Figure 2 illustrates this procedure. Since about half the members of R_t can be accommodated to P_{t+1}, the above procedure is continued till the last acceptable front $\mathcal{F}_l$. Let us say that solutions remaining to be filled before this last front is considered is N' and the number of non-duplicate solutions in the last front is N_l ($> N'$). Based on a *crowding distance* value (Deb, 2001), only those N' solutions which will make the diversity of the solutions maximum are chosen from N_l.

It is important to realize that in the classification problem there may exist multiple classifiers resulting in an identical set of objective values (having same classifier size and classification accuracy for example). The following fix-up to the original NSGA-II enables us to find multiple classifiers simultaneously. We compute the number of distinct objective solutions in the set $\mathcal{F}_l$ and let us say it is n_l (obviously, $n_l \leq N_l$). If $n_l \geq N'$ (the top case shown in the Figure 2), we follow the usual crowding distance procedure to choose N' most dispersed and distinct solutions from n_l solutions. The major modification to NSGA-II is made when $n_l < N'$ (bottom case in the figure). This means that although there are fewer distinct solutions than the population slots, the distinct solutions are multi-modal. However, the total number of multi-modal solutions of all distinct solutions (N_l) is more than the remaining population slots. Thus, we need to make a decision of choosing a few solutions. The purpose here is to have at least one copy of each distinct objective solution and as many multi-modal copies of them so as to fill up the population. Here, we choose a strategy in which every distinct objective solution is allowed to have a proportionate number of multi-modal solutions as they appear in $\mathcal{F}_l$. To avoid loosing any distinct objective solutions, we first allocate one copy of each distinct objective solution, thereby allocating n_l copies. Thereafter, the proportionate rule is applied to the remaining solutions ($N_l - n_l$) to find the accepted number of solutions for the i-th distinct objective solution as follows:

$$\alpha_i = \frac{N' - n_l}{N_l - n_l}(m_i - 1), \qquad (3)$$

where m_i is the number of multi-modal solutions of the i-th distinct objective solution in $\mathcal{F}_l$, such that $\sum_{i=1}^{n_l} m_i = N_l$. The final task is to choose $(\alpha_i + 1)$ random multi-modal solutions from m_i copies for the i-th distinct objective solution. Along with the duplicate-deletion strategy, the random acceptance of a specified number multi-modal solutions to each distinct objective solution ensures a good spread of solutions in both objective and decision variable space. In the rare occasions of having less than N non-duplicate solutions in R_t, new random solutions are used to fill up the population.

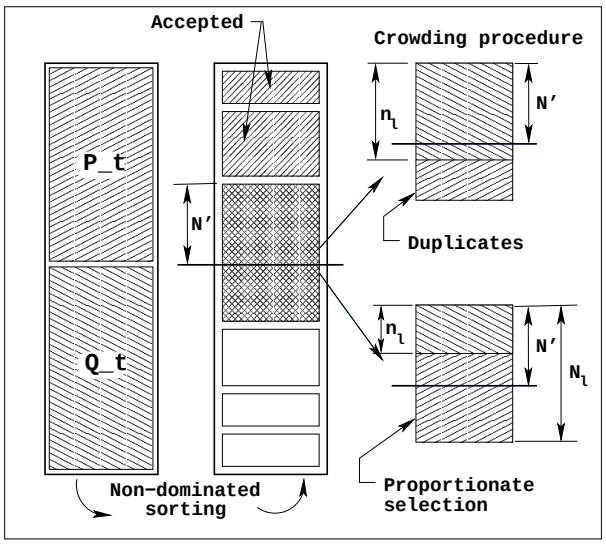

Figure 2: Schematic of the multi-modal NSGA-II procedure is shown.

SIMULATION RESULTS

In this section, we present results obtained by the modified NSGA-II on different cancer datasets: Leukemia, Lymphoma, Colon, GCM, and NCI60.

Leukemia Dataset

The Leukemia gene expression dataset (Golub et al., 1999) containing expression profiles of 72 leukemia

samples each in 7,129 gene was downloaded from `http://www.genome.wi.mit.edu/MPR`. The dataset was divided into two groups: an initial training set of 27 samples of acute lymphoblastic leukemia (ALL) and 11 samples of acute myeloblastic leukemia (AML), and an independent test set of 20 ALL and 14 AML samples. Here, the expression values are preprocessed by using a threshold of 20 units and a ceiling of 16,000 units, and then exclude genes violating $max(x_g) - min(x_g) > 500$ and $max(x_g)/min(x_g) > 5$ conditions from further consideration, leaving a total of 3,859 genes. Based on the suggestion in (Golub et al., 1999), the logarithm of the gene expression values (denoted as $\hat{x_g}$) are calculated and then normalized as follows: $\bar{x}_g = (\hat{x_g} - \mu)/\sigma$. Here, μ and σ are the mean and standard deviation of the $\hat{x_g}$ values in the training set only.

First, we apply the multi-modal NSGA-II on 50 genes which were used in another Leukemia study (Golub et al., 1999), considering minimization of above three objectives (f_1, f_2, and f_3). For NSGA-II, we choose a population of size 500 and run upto 500 generations with a single-point crossover with a probability of 0.8 and a bit-wise mutation with a probability of $p_m = 1/50$. In this case, we use $\theta = 30\%$ threshold on the prediction strength. We obtain eight distinct non-dominated solutions, as shown in Figure 3. It can be observed from the figure that there

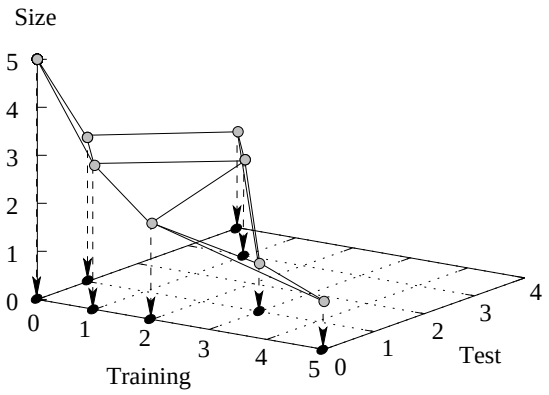

Figure 3: Eight non-dominated solutions found for the 50-gene Leukemia dataset.

exist a five-gene classifier with zero mismatches in all 72 leukemia samples and other classifiers with reducing number of genes but producing non-zero mismatches, meaning that any other classifier with less than five genes is not capable of providing 100% correct classification. Interestingly, the multi-modal NSGA-II has also discovered a number of multi-modal solutions corresponding to each of the Pareto-optimal solutions, which are all depicted in Figure 4. Here, we found 31 different five-gene classifiers which are all making no mismatches in all training and test samples. Each column in the figure represents a classifier. The corresponding mismatch in classifying all samples is also marked.

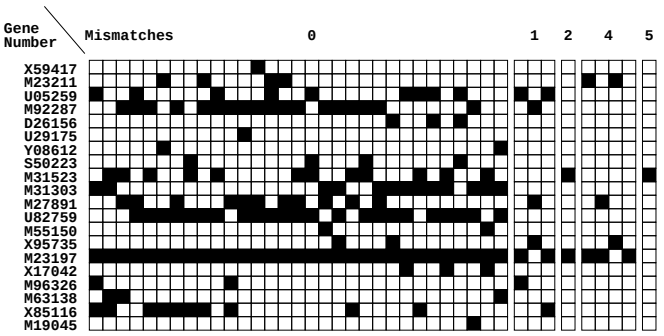

Figure 4: The multi-modal solutions in 50-gene Leukemia dataset with 30% threshold on prediction strength. Each column represents a classifier with marked genes as members of the classifier.

Figure 4 brings out an interesting aspect. Among 31 different five-gene combinations, three genes (accession numbers: M92287, U82759, and M23197) found to appear in more than 50% of the obtained high-performing classifiers. Moreover, the gene M23197 appears in all 31 classifiers. Such information about frequently appearing genes in high-performing classifiers is certainly useful to biologists. Hence, investigating these frequently appearing genes further from a biological point of view should provide intriguing information about the causes of different cancer diseases.

Since the above results unveiled the importance of using multi-modal NSGA-II in finding multiple distinct Pareto-optimal solutions, we apply the same multi-modal NSGA-II on the complete Leukemia dataset to investigate if there exist a smaller gene subset (less than five genes) with 100% correct classification. In the case of complete Leukemia dataset, we use a binary string of length 3,859, the total number of genes after preprocessing. Because of the large string length requirement, we have chosen 1,000 population size here and run NSGA-II for 1,000 iterations. Here, we use a mutation probability of 0.0005, so that on an average about two bits get mutated in the complete string. We consider a classification procedure with $\theta = 30\%$ threshold on the prediction strength as before. Strikingly, the multi-modal NSGA-II has found two different three-gene classifiers making 100% correct classification in all samples. The gene accession numbers and the corresponding correlation values against ALL versus AML (calculated based on 38 training samples and all 72 samples separately) of the genes found in above two different three-gene classifiers are shown in Table 1. Two genes having accession numbers M23197 and U05259 found in classifier 1 were also present in 50-gene Leukemia study discussed earlier, but the third gene U58046 appears as an important gene here, as with the presence of this gene the optimal classifier size is reduced to three, instead of five as obtained before. Unfortunately, the original 50-gene study (Golub et al., 1999) did not consider this gene. In case of classifier 2, the presence of X64364 gene (which was also

Table 1: The gene accession numbers and their corresponding correlation values between ALL and AML. The second column represents the correlation values calculated based on only training samples and the values in third row calculated based on all 72 samples.

Classifier 1		
M23197	U05259	U58046
-1.416	0.962	0.396
-1.441	1.002	0.220
Classifier 2		
M23197	X64364	M31523
-1.416	-0.744	1.292
-1.441	-0.601	1.427

not included in the original 50-gene study) yields another compact classifier producing 100% classification accuracy. Another observation is that the gene M23197 appears in all classifiers obtained by NSGA-II, thereby making it the single most important gene in the classification of ALL and AML in a Leukemia sample. This study shows how from original 7,129 genes the attention can be focused to three (Table 1) or even to one gene (M23197), achieving more than a thousand-fold advantage. It is worth mentioning that a recent study using an artificial immune system has also stressed the importance of finding multiple high-performing classifiers for Leukemia dataset (Ando and Iba, 2003). That study reported eight classifiers having four to 11 genes capable of making 100% correct classification.

In order to show the confidence in our classification, we show the minimum, maximum, and average prediction strength (ps_{min}, ps_{max}, and ps_{avg}) values observed in all 72 samples in Table 2. Since a threshold $\theta = 0.3$ is used here, the minimum expected prediction strength of 0.3 is ensured.

Table 2: The minimum, maximum, and average prediction strength values observed in all 72 Leukemia samples using classifiers 1 and 2.

Classifier	ps_{min}	ps_{max}	ps_{avg}
1	0.358	1.000	0.827
2	0.389	1.000	0.914

Effect of Prediction Strength

To investigate the effect of chosen threshold (θ) on the obtained classifiers, we apply the above multi-modal NSGA-II with identical parameters as used above on the complete Leukemia dataset for different values of θ. Table 3 shows the number of genes in the classifiers, number of mismatches in training and test sets, and the number of multi-modal solutions obtained for different values of θ. In general, as the classification prediction strength is increased, the optimal classifier size increases and the cardinality of high-performing classifiers reduces. Compared to an earlier study (Liu and Iba, 2002), our approach finds a better classifier with a smaller size and providing a 100% classification accuracy.

Effect of NSGA-II Parameters

It has been observed that among all other NSGA-II parameters (crowded tournament selection, crossover, and elitism), the mutation probability plays a vital role. To unveil the importance of using an optimum mutation probability, we rerun the above NSGA-II with different mutation probabilities on the complete Leukemia dataset. The minimum gene subset size obtained at the end of 10,000, 50,000, 250,000, 500,000, and 750,000 solution evaluations are recorded and plotted in Figure 5. It is clear that with the increase in the number of evaluations (or generations), NSGA-II reduces the minimum gene subset size for any mutation probability. However, after 500,000 evaluations, there is no change in the optimal gene subset size for most mutation probabilities, meaning that there is no need to continue running NSGA-II after these many evaluations. When an optimum mutation probability ($p_m \sim 1/\ell$) is chosen, about 250,000 evaluations are enough. The figure also indicates the total number of mismatched samples in each case. It is also clear that a 100% correct classification cannot be obtained with less than three genes.

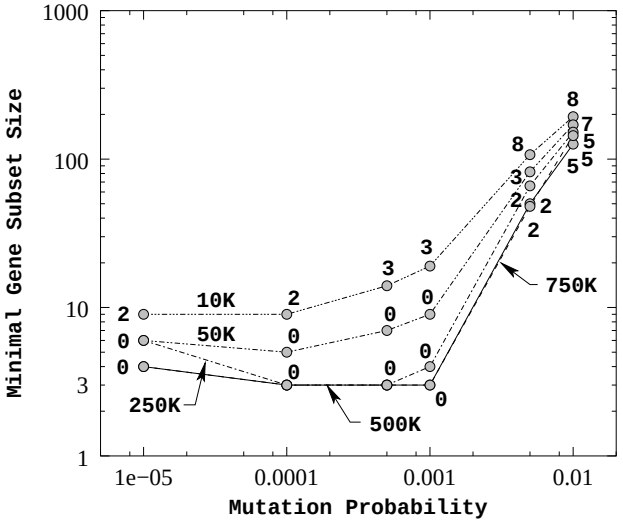

Figure 5: The effect of mutation probability on the obtained minimum gene subset size. The number near each point indicates the total number of mismatched samples obtained with the gene subset.

Diffuse Large B-Cell Lymphoma Dataset

The diffuse large B-cell lymphoma (DLBCL) dataset (Alizadeh et al., 2000) contains expression measurements of

Table 3: Multi-modal solutions for three disease samples corresponding to different values of θ. Parameters f_1, f_2, f_3, and α represent gene subset size, mismatches in training samples, mismatches in test samples, and the number of multi-modal solutions obtained with (f_1, f_2, f_3) values.

θ(%)	Leukemia samples					Lymphoma samples					Colon samples				
	f_1	f_2	f_3	% correct	α	f_1	f_2	f_3	% correct	α	f_1	f_2	f_3	% correct	α
0	3	0	0	100	352	5	0	0	100	121	6	0	0	100	13
Liu & Iba	16			97	1	18			94	1	14			90	1
10	3	0	0	100	113	6	0	0	100	258	7	0	1	100	8
20	3	0	0	100	18	5	1	0	99	155	7	1	2	95	16
30	3	0	0	100	2	8	0	1	99	1	6	2	2	94	3

96 normal and malignant lymphocyte samples each measured using a specialized cDNA microarray, containing 4,026 genes that are preferentially expressed in lymphoid cells or which are of known immunological or oncological importance. There are 42 DLBCL and 54 other cancer disease samples. The expression data was downloaded from `http://llmpp.nih.gov/lymphoma/data/figure1.cdt`. However, some arrays contain a number of genes with missing expression values. For correcting missing expression values, we have used k-nearest neighbor algorithm (Troyanskaya et al., 2001), in which k genes with similar expression profiles to the gene of interest to impute missing values are selected and the missing values of that particular gene are imputated by using a simple weighted average of k nearest genes. Thereafter, we randomly divide the dataset into a training set and a test set of equal sizes (each with 50% samples), and the expression values are normalized as the case before.

When we apply the multi-modal NSGA-II with the same NSGA-II parameters as in the 3,859-gene Leukemia case and with a threshold of $\theta = 30\%$, we obtain only one eight-gene classifier producing 100% correct classification in all training samples but one mismatch (out of 48) in test samples. Although the multi-modal NSGA-II fails to identify any classifier providing 100% correct classification in all 96 samples with a threshold of $\theta = 30\%$, we found 121 different five-gene classifiers with zero mismatches in both training and test samples with a poor confidence in classification (with $\theta \leq 10\%$) values of θ), as shown in Table 3. Compared to (Liu and Iba, 2002), our approach finds a better classifier, as shown in Table 3.

Colon Cancer Dataset

The Colon gene expression dataset (Alon et al., 1999) containing expression values of 62 colon biospy samples measured using high density oligonucleotide microarrays containing 2,000 genes is available at `http://microaaray.princeton.edu/oncology`. It contains 22 normal and 40 Colon cancer samples. The dataset is randomly partitioned into two groups, of which 50% of the samples (31 samples) for training and remaining 50% for testing purpose. The gene expression values are log-transformed and then normalized as in the case of Leukemia samples.

With identical NSGA-II parameters to those used in the Leukemia case except a mutation probability of 0.001, different classifiers are obtained corresponding to different θ, as shown in Table 3. It can be seen from the table that the multi-modal NSGA-II has found 13 different six-gene classifiers with zero mismatches in both training and test samples without any threshold on prediction strength. But, as we increase the value of θ, no classifier is found providing 100% correct classification. With 30% threshold, 94% samples are correctly classified.

This study shows that the outcome of the classification depends on the chosen prediction strength threshold. Keeping a higher threshold makes a more confident classification, but at the expense of some mismatches, while keeping a low threshold values may make a 100% classification, but the classification may have been performed with a poor confidence level. Also for a low threshold, there exist more classifiers providing 100% correct classification. Compared to (Liu and Iba, 2002), NSGA-II finds a better classifier, as shown in Table 3.

GCM Multi-Class Tumor Dataset

Next, we apply multi-modal NSGA-II on a more challenging and commonly-used multi-class GCM expression dataset (Ramaswamy et al., 2001), containing expression values of 198 primary tumor samples in 16,063 genes is available at `http://www-genome.wi.mit.edu/MPR`. The data contains 144 initial training samples and 54 independent test samples. Each set has 14 different common tumor types: 12 breast, 14 prostate, 12 lung, 12 colorectal, 22 lymphoma, 11 bladder, 10 melanoma, 10 uterus, 30 leukemia, 11 renal, 11 pancreas, 12 ovary, 11 mesothelioma, and 20 central nervous system (CNS) samples. We process the expression values with the same parameters as used in the case of Leukemia data. As a result, the number of significant genes in the dataset is reduced to 11,367 (from 16,063). The truncated expression values are log-transformed and then normalized such that the mean and the standard deviation of the expression values becomes zero and one respectively.

We use a NSGA-II population of size 1,000 and run upto 2,000 generations with a crossover probability of

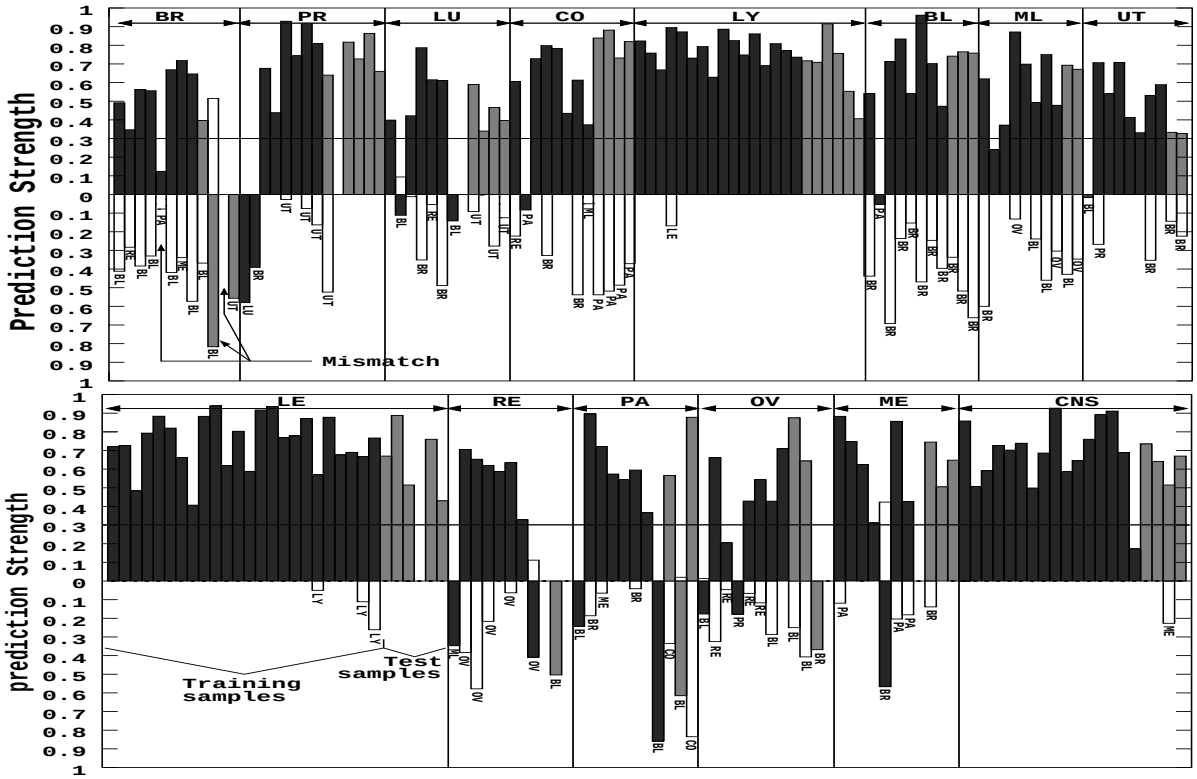

Figure 6: The first figure (top) represents the prediction strengths for first 103 (class 1 to class 8) samples. The second figure (bottom) represents the prediction strengths for remaining 95 (class 9 to class 14) samples. For each sample, the box directing upward represents the largest prediction strength and the box directing downward represents the next-best prediction strength. Any upward box having a value at least 0.3 indicates a correct match.

0.8 and with a mutation probability of 0.0001. We use $\theta = 30\%$ threshold on prediction strength. The multi-modal NSGA-II discovers an optimum gene cluster having 37 genes producing 86% and 80% classification accuracies in training and test samples, respectively. Moreover, the optimal 37-gene cluster also ensures that the minimum prediction strength observed in any of the perfectly classified sample is at least 30%, as shown in Figure 6. The class-wise performance of the 37-gene classifier is also shown in Table 4. The obtained classifier is capable of providing a better classification with an overall classification accuracy of 84.3% (167 correct samples out of 198) compared to two other studies in the past. A recent study (Ramaswamy et al., 2001) has used various ranking-based supervised and unsupervised learning methods to identify an optimal gene subset for the problem. The best classification accuracy is reported to be 80.3% (159 out of 198 samples) with 16,063-gene classifier obtained using the OVA/SVM (support-vector-machine) based method. Another study (Ooi and Tan, 2003) showed the applicability of GM (genetic algorithm/maximum likelihood) based classification methodology to the same dataset. Besides considering a truncated dataset containing only 1,000 genes (selected with high standard deviation values ranging from 0.299 to 3.089), the study focused in classifiers of size ranging from 1 to 60 only and obtained a 32-gene classifier having 81.3% (161 out of 198 samples) classification accuracy. Thus, in addition to being slightly more accurate than the GM-based classification (84.3% as compared to 81.3%), this study demonstrates that the multi-modal NSGA-II is applicable to a high-dimensional search space.

For each sample, we calculate the prediction strength corresponding to its belonging to all 14 classes by the procedure described earlier. Thereafter, we find the largest and the next-largest prediction strength values and plot them in Figure 6. These two values are differentiated with shaded-boxes and unshaded-boxes, respectively. For each sample, the box directing upward represents the prediction strength corresponding to its actual class and the box directing downward represents the prediction strength to its next probable class. A box with a dark shade represents the training sample and a box with a light shade represents the test sample in each class. If, for a sample, the shaded-box directing upward has a value more than 0.3, we consider the sample as a correctly-classified one, else the sample is considered as a misclassified one. Interestingly, there are no boxes (neither shaded nor unshaded) corresponding to sample 11, 22, 34, 123, 134, 136, and 174, which implies that the sign of the right term of the Equation 2 is negative for all j's (j = 1,2,...,14) and they are also considered to be misclassified samples. The important observations from this study are as follows:

Table 4: Best obtained results with multi-modal NSGA-II for the GCM multi-class cancer data. The available number of samples (S_j) and number of correctly classified (training, test) samples for a few methods are shown.

Class	S_j	NSGA-II	GM	SVM
Breast	(8,4)	(7,1)	(6,2)	(7,2)
Prostate	(8,6)	(6,5)	(6,5)	(6,4)
Lung	(8,4)	(5,4)	(4,4)	(4,2)
Colorectal	(8,4)	(7,4)	(5,4)	(6,4)
Lymphoma	(16,6)	(16,6)	(16,6)	(16,6)
Bladder	(8,3)	(7,3)	(7,3)	(5,2)
Melanoma	(8,2)	(7,2)	(5,1)	(5,1)
Uterus	(8,2)	(7,2)	(5,2)	(7,2)
Leukemia	(24,6)	(24,5)	(24,6)	(24,5)
Renal	(8,3)	(6,0)	(7,0)	(5,3)
Pancreas	(8,3)	(6,3)	(3,2)	(5,2)
Ovary	(8,4)	(5,2)	(5,4)	(3,2)
Mesothl.	(8,3)	(6,3)	(7,3)	(8,3)
CNS	(16,4)	(15,4)	(15,4)	(16,4)
Total	(144,54)	(124,43)	(115,46)	(117,42)
% Class. accuracy		**84.3**	**81.3**	**80.3**

1. If the available number of samples belonging to one particular class is more, then the performance of the obtained classifier is better. Since Lymphoma, Leukemia, and CNS tumors have more samples compared to other tumor types, more correct predictions are made in these cases.

2. Although BR and BL datasets have similar patterns, the obtained classifier makes an overall 88% correct classification. Moreover, in most cases the competing class for a BR sample is the BL class and vice versa.

NCI60 Multi-Class Tumor Dataset

Finally, we apply the multi-modal NSGA-II to the NCI60 expression dataset (Ross et al., 2000). In this case, the cDNA microarrays containing 9,703 spotted cDNA probes were used to measure the variation in gene expression values among 64 cancer cell lines. The experimental observations were derived from tumors with different sites of origin: 7 breast, 6 CNS, 7 colon, 6 leukemia, 8 melanoma, 9 non-small-cell-lung-carcinoma (NSCLC), 6 ovarian, 8 renal, 4 reproductive, 2 prostate, and 1 unknown cell line. In order to compare with past studies (Fridlyand et al., 2002; Ooi and Tan, 2003), two prostate and one unknown cell line observations are excluded from analysis, leaving a total of 61 samples. The dataset is available from `http://genome-www.stanford.edu/sutech/download/nci60/dross_arrays_nci60.tgz`. In this study, the gene expression measurements are preprocessed based on the guidelines given in (Ooi and Tan, 2003) and 6,167 genes are retained for 61 samples.

In this case, we have used a population of size 1,000 and run NSGA-II upto 1,000 generations with identical crossover and mutation probabilities as used in the GCM multi-class dataset. The results obtained with $\theta = 0\%$ and $\theta = 30\%$ are shown in Table 5. The multi-modal NSGA-II has found six different 14-gene classifiers making 90% (4 out of 41 and 2 out of 20 mismatches in training and test samples, respectively) classification accuracy without any threshold on prediction strength. These results are much better than any other previously reported results. The study (Ooi and Tan, 2003) used the same dataset and solved the problem with a GM-based classification approach and reported a different 14-gene classifier having 80% overall classification accuracy. However, that study also applied the same GM-based classification technique on a truncated dataset of 1,000 genes and found an optimum gene classifier having 13 genes producing 88.52% combined classification accuracy in training and test samples. An application of our multi-modal NSGA-II on the same 1,000 gene expression dataset and without any threshold finds a classifier having 12 genes producing 92.68% and 90% classification accuracies in training and test samples, respectively. However, with $\theta = 30\%$ threshold we found a 13-gene classifier making 85.36% and 90% classification in training and test samples, respectively. Table 5 summarizes these results. Since the

Table 5: The results obtained with multi-modal NSGA-II with $\theta = 0\%$ and $\theta = 30\%$ thresholds on a complete dataset of 6,167 genes and also on a truncated dataset of 1,000 genes. Parameters f_1, f_2, f_3, and α represents gene subset size, mismatches in training samples, mismatches in test samples, and the number of multi-modal solutions. Suc. refers to percentage classification success.

	6,167 genes					1,000 genes				
θ	f_1	f_2	f_3	α	Suc.	f_1	f_2	f_3	α	Suc.
0	14	4	2	6	90	12	3	2	1	91.8
30	11	6	4	8	83.6	13	6	2	1	86.9
GM	14			1	80	13			1	88.5

GM study did not use any prediction strength concept, it is fair to compare the result with our approach having a zero threshold. In this case, our 12-gene classifier achieves 91.8% accurate result than the 13-gene classifier reported in (Ooi and Tan, 2003) resulting in 88.5% accuracy.

Interestingly, our 1,000-gene study has found the following 12 genes in the classifier without any threshold: (5, 46, 63, 97, 125, 128, 170, 242, 408, 568, 664, and 755) and the 13-gene classifier with 30% threshold: (5, 17, 19, 46, 63, 97, 128, 216, 282, 284, 319, 568, and 653). The former classifier has one common gene (gene 97) with that obtained by GM and other studies. However, there are six genes in common between two classifiers obtained using nonzero and zero threshold values with NSGA-II, thereby indicating a reasonable confidence in pursuing them for further investigation. The study helps reduce the focus to six to 13 genes from original 6,167 genes involving the NCI60 dataset.

CONCLUSIONS

The identification of gene subsets responsible for classifying disease samples to fall in one category or another has been dealt in this paper. By treating the resulting optimization problem with three objectives, we have applied a multi-objective evolutionary algorithm (NSGA-II)to find optimal gene-subsets for five different available micro-array dataset. The following conclusions can be drawn from the study:

- Compared to all past studies involving similar classification of micro-array dataset, the modified NSGA-II finds smaller or similar-sized classifiers but resulting in more accurate classifications.
- A generic procedure is adopted for tasks involving two or more classes.
- Reliability in classification is ensured in obtained classifiers by using a prediction strength consideration in evaluating a classifier.
- The proposed approach is capable of finding *multiple* different classifiers (gene combinations) each of the same size and classification accuracy.

Due to the availability of only a few samples compared to the large number of gene expression values in disease samples, it is likely that in such classification problems there may exist multiple gene combinations producing near 100% classification accuracy. Since such results are obtained purely from a computing point of view, all such classifiers may not make much sense biologically. However, besides finding multiple such classifiers in one simulation run, the present computational approach is uniquely important from another point of view. The obtained high-performing multiple classifiers can be analyzed to discover any common set of genes. If found, such genes may provide useful insights to biologists for unveiling salient information about the cause and cure of the disease. In most cancer disease datasets used here, the study has unveiled such vital information for their further and immediate processing.

References

Alizadeh, A. A., Eisen, M. B., Davis, R. E., Ma, C., Losses, I. S., Rosenwald, A. and et al. (2000). Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature 403*, 503–511.

Allwein, E. L., Schapire, R. E. and Singer, Y. (2000). Reducing multiclass to binary: A unifying approach for margin classifiers. In *Proc. 17th International Conf. on Machine Learning*, pp. 9–16. Morgan Kaufmann, San Francisco, CA.

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D. and Levine, A. J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. In *Proceedings of National Academy of Science, Cell Biology*, Volume 96, pp. 6745–6750.

Ando, S. and Iba, H. (2003). Artificial immune system for classification of cancer. In *Proceedings of the Applications of Evolutionary Computing (LNCS 2611)*, pp. 1–10. Berlin, Germany: Springer.

Bose, R. C. and Ray-Chaudhari, D. K. (1960). On a class of error correcting binary group codes. *Information Control 2*, 68–79.

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. Chichester, UK: Wiley.

Deb, K., Agrawal, S., Pratap, A. and Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation 6*(2), 182–197.

Fridlyand, J., Dudoit, S. and Speed, T. P. (2002). Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association 97*, 77–87.

Goldberg, D. E. (1989). *Genetic Algorithms for Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P. and et al. (1999). Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science 286*, 531–537.

Kohavi, R. and John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence Journal, Special Issue on Relevance 97*, 234–271.

Liu, J. and Iba, H. (2002). Selecting informative genes using a multiobjective evolutionary algorithm. In *Proceedings of the World Congress on Computational Intelligence (WCCI-2002)*, pp. 297–302.

Liu, J., Iba, H. and Ishizuka, M. (2001). Selecting informative genes with parallel genetic algorithms in tissue classification. *Genome Informatics 12*, 14–23.

Ooi, C. H. and Tan, P. (2003). Genetic algorithms applied to multi-class prediction for the ananlysis of gene expression data. *Bioinformatics 19*(1), 37–44.

Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C. H., Angelo, M. and et al. (2001). Multiclass cancer diagnosis using tumor gene expression signatures. *Proceedings of the National Academy of Science 98*(26), 15149–15154.

Ross, D. T., U. Scherf, M. B. E., Perou, C. M., Rees, C., Spellman, P., Iyer, V., Jeffrey, S. S., de Rijn, M. V. and et al., M. W. (2000). Systematic variation in gene expression patterns in human cancer cell lines. *Nature Genetics 24*, 227–235.

Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D. and Altman, B. R. (2001). Missing value estimation methods for DNA microarrays. *Bioinformatics 17*(6), 520–525.