Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 9 (3): 459-468, 2009
ISSN 1812-5654

Classification and Diagnostic Prediction of Cancers Using Gene Microarray Data Analysis

Alireza Osareh and Bita Shadgar
Department of Computer Science, Faculty of Engineering, Shahid Chamran University, Ahvaz, Iran

Abstract: In this study, we aim to develop an automated system for robust and reliable cancer diagnoses based on gene microarray data. Amongst various utilized statistical classifiers, support vector machines outperform other popular classifiers, such as K nearest neighbours, naive Bayes, neural networks and decision tree, often to a remarkable degree. We choose a set of 9 publicly available benchmark microarray datasets that encompass both binary and multi-class cancer problems. Results of comparative studies are provided, demonstrating that effective feature selection is essential to the development of classifiers intended for use in gene-based cancer classification. In particular, amongst various systematic experiments carried out, best classification model is achieved using a subset of features chosen via information gain feature ranking for support vector machine classifier.

Key words: DNA, support vector machines, K nearest neighbours, gene selection, gene classification

INTRODUCTION

In recent years there has been an explosion in the rate of acquisition of biomedical data. Advances in molecular genetics technologies, such as DNA microarrays (Hedge, 2000) allow us to obtain a global view of the cell. For example, we can now routinely investigate the biological molecular state of a cell measuring the simultaneous expression of tens of thousands of genes using DNA microarrays.

An important emerging medical application domain for DNA microarray gene expression technology is clinical decision support. This support can be in the form of diagnosis of disease as well as the prediction of clinical outcomes in response to treatment. The two areas in medicine that currently attract the greatest attention in this respect are management and classification of cancer and infectious diseases (Ntzani, 2003).

Availability of gene expression data of tissue samples from different diagnostic classes led to the application of many well-established statistical learning algorithms to these profiles, in an attempt to provide more accurate and automatic diagnostic class (cancer/non-cancer) prediction. On the other hand, analysis of microarrays presents a number of unique challenges. Microarray data samples are typically available just from a small number of patients (often less than one hundred) mainly due to the expense of collecting microarray data. Indeed, the dimensionality of the decision problem is enormous (Mukherjee and Roberts, 2004) (usually thousands or tens of thousands of genes). The curse of dimensionality (Scott, 1992) and consequent issues such as the performance and generalization abilities of a classifier are the main reasons for restricting the data dimension.

In fact, there is a high redundancy in microarray data and many genes contain irrelevant information for accurate classification of diseases or phenotypes. Only a small number of genes may be important. Therefore, there is a pressing need for techniques capable of selecting the appropriate subset of genes relevant to a particular problem from among the entire set of microarray data.

The issue of gene selection/extraction has become a central challenge in the field of microarray data analysis and has been the subject of different studies (Guyon *et al*., 2002; Yeung *et al*., 2005). Gene selection is often used as a pre-processing step to find a reduced subset of features without significantly degrading the performance of a subsequent classifier. There are two general feature selection approaches; the filter and wrapper model (John and Kohavi, 1994). Filter methods rank genes according to certain pre-defined criterion; for example, statistical tests such as the t-statistics, information gain, relief algorithm and signal to noise ratio are some of the widely known filter approaches (Mohd *et al*., 2007). The wrapper approaches, in contrast, are more complex because they require a trained learning machine that can evaluate the relevance of a selected subset of genes. The wrapper approach finds a subset of genes and estimates its relevance using a machine like classifier.

Corresponding Author: Alireza Osareh, Department of Computer Science, Faculty of Engineering, Shahid Chamran University, Ahvaz, Iran Tel: +98-6113330010 Fax:+98-611-3336642

As an alternative to feature selection-based dimensionality reduction methods, feature extraction-based schemes such as principal component analysis has also been utilized to reduce the dimensionality of gene expression data sets using only few so-called principal features (Xiong *et al*., 2000).

A number of researchers have already been reported in the past for the cancer classification task using the microarray data. Here, a brief description of the most relevant research works is provided to highlight the techniques which have been used for gene selection and classification.

Golub *et al*. (1999) adopted gene selection criteria based on correlation of genes prior to the classification. The selected genes were utilized in weighted voting approach for cancer classification. Furey *et al*. (2000) applied similar technique as of Golub *et al*. (1999) for gene selection and demonstrated the use of Support Vector Machine (SVM) for cancer classification. Zhang *et al*. (2006) developed a type of regularization in SVM to identify important genes for cancer classification, compared the performance of different discrimination methods for classification of tumors. These methods included nearest neighbour classifier, linear discriminant analysis, diagonal discriminant analysis and classification trees. They considered bagging (Bylander, 2002) and boosting (Freund and Schapire, 1997) approaches to select relevant genes, which were used in classification.

Khan *et al*. (2001) employed PCA for dimension reduction before using neural networks for classification. Leng and Muller (2006) utilized functional logistic regression approach which was based on functional PCA for classification of gene expression data.

Yeung *et al*. (2001) and McLachlan *et al*. (2002) discussed the use of model-based gene expression profile clustering approaches for classification. They used Gaussian mixture models to cluster the samples and eventually tested the groupings in the samples against a priori known class information for the samples.

A major goal of an automated decision support system in cancer diagnosis such as ours is to develop diagnostic procedures based on the least number of possible genes needed to detect diseases. Thus, in this study, as an alternative to earlier methods, a simple yet powerful algorithm that can accurately predict gene expression class by using less number of genes is proposed. At the heart of this study lies the assumption that an automated system for use in clinical applications should benefit from high quality and robust classification models.

Therefore, in this study, we present a novel embedded approach composed of two main phases to the problem of cancer classification using gene expression data. Phase one includes gene selection. Selecting important predictive genes can really make the task easier because important genes determine a new reduced input space in which is the samples are more likely to be correctly classified. The second phase is to build powerful classifier models. Here, we touch upon both phases from an experimental viewpoint. For gene selection, we analyze 3 filter approaches, i.e., information gain, relief algorithm and t-statistics and a feature extraction method based on PCA. Having obtained a predictive reduced feature space comprising the most informative genes, 5 well-known classifiers i.e., support vector machine, K nearesr neighbour, naive Bayes, neural networks and decision tree are utilized to classify 9 famous publicly available gene expression datasets. As we show in the experimentation section, this approach allows us to obtain highly competitive results on all the data sets.

MATERIALS AND METHODS

Classification algorithms: In this study, 5 supervised learning models are adopted to build models to perform gene classification, namely, support vector machine, K-Nearest Neighbours (Duda and Hart, 1973), Naive Bayes (Duda and Hart, 1973; Devijver and Kittler, 1982), neural networks and Decision Tree (Michell, 1997). This is practical because class labels of the training examples are available for use in the search for separating genes. To be self-contained, this section gives a short overview of these models.

Classification performance matrices: As we are proposing a new gene selection scheme, we want to measure its performance and take a comparison. This scheme provides us with a set of informative genes that will be used for future classification. During classifier design, using the information extracted from the training samples, underlying parameters of the particular classifier are adjusted and the prediction accuracy is monitored by testing the classifier back on the training set and noting the resultant training (or resubstitution) error. This type of assessment of classifier performance, based on training error, is instrumental during the design phase; however, it may not be an accurate indicator of the final or overall performance of the classifier. As we are interested in employing present classifier in predicting diagnostic category of new or unseen samples, we also need to evaluate the generalization ability of the classifier.

If the training set contains too many samples with characteristics off the line with the population they represent (i.e., outliers) or excessive training is done so that the classifier learns or models even the inherent noise in the samples, the generalization performance of the classifier will be poor. Therefore, while evaluating prediction accuracy of classification methods, it is

important not to use the training error only (Braga-Neto and Dougherty, 2004). If there are plenty of training samples available, one can partition the overall training set into two sets and use one for training and the other one for testing.

However, the number of gene expression profile samples is generally too few to permit this. If we design the classifier based on a small training set, the generalization performance of the classifier will be poor again.

A common technique to assess classifier performance in such situations is to use m-fold cross validation. In this way, the overall set of n training samples is randomly divided into m approximately equal size and balanced set of subsets. Then each time one of these subsets is excluded from the overall training set and used as a test set (for the classifier that is trained based on the rest of the samples). This is repeated over the m sub-samples and the resultant test error rates are averaged to obtain the so-called m-fold cross validation error rate.

Support vector machines: Support vector machines (Vapnik, 1998) are becoming increasingly popular classifiers for microarray data. Advantages of a SVM in cancer diagnostic models where the number of feature variables (genes) p is so large relative to the sample size n, are that it is able to be fitted to all the genes and that its performance appears not to be too affected by using the full set of genes.

Formally, given a training set belonging to two classes, $\{X_i, y_i\}$ where, $\{X_i\}$ are the n training samples with their class labels y_i, a soft-margin linear SVM classifier aims at solving the following optimization problem:

$$\min_{w,b,\xi_i} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \quad (1)$$

subject to:

$$y_i(w.X_i + b) \geq 1 - \xi_i \quad \text{and} \quad \xi_i \geq 0, \quad i = 1,...,n \quad (2)$$

C is a given penalty term that controls the cost of misclassification errors. To solve the optimization problem, it is convenient to consider the following dual formulation:

$$\min_{\alpha_i} \frac{1}{2}\sum_{i=1}^{n}\sum_{l=1}^{n}\alpha_i\alpha_l y_i y_l X_i.X_l - \sum_{i=1}^{n}\alpha_i \quad (3)$$

subject to:

$$\sum_{i=1}^{n} y_i\alpha_i = 0 \quad \text{and} \quad 0 \leq \alpha_i \leq C \quad (4)$$

The decision function for the linear SVM classifier with input vector X is given by:

$$f(X) = w.X + b \quad \text{with} \quad w = \sum_{i=1}^{n}\alpha_i y_i X_i \quad \text{and} \quad b = y_i - w.X_i \quad (5)$$

The weight vector w is a linear combination of training samples. Most weights α_i are zero and the training samples with non-zero weights are support vectors.

In SVM, the complexity of the classifier is based on the number of support vectors rather than the dimensionality of the feature space and this makes the algorithm less prone to over-fitting. The samples in the original feature space are projected onto a higher dimensional feature space where they can be separated by a maximal margin hyper-plane.

K-nearest neighbour: K-Nearest Neighbour (KNN) is one of the simplest learning algorithms and has been successfully applied to a broad range of problems (Kuramochi and Karypis, 2005). To classify an unclassified vector X, the KNN algorithm ranks the neighbours of X amongst a given set of N data (X_i, c_i), i = 1,2,..., N and uses the class labels c_j (j = 1, 2, ..., K) of the K most similar neighbours to predict the class of the new vector X. In particular, the classes of these neighbours are weighted using the similarity between X and each of its neighbours, where similarity is typically measured by the Euclidean distance metric (though any other distance metric may also do). Then, X is assigned the class label with the greatest number of votes among the K nearest class labels.

Naive Bayes: A Naive Bayes (NB) classifier can achieve relatively good performance on classification tasks (Michell, 1997) based on the elementary Bayes' theorem. It greatly simplifies learning by assuming that features are independent given the class variable. Given a K-class classification problem, let x denote an observation in a d-dimensional feature space with a probability density of p(x) and k =1, 2, ..., K be the class index. If class prior probabilities p(k) and the class conditional probability densities p(x|k) are known, according to Bayes theory, the posterior probabilities are:

$$P(k|x) = p(x|k).\ p(k)/p(x) \quad (6)$$

The p(x) in the denominator of Eq. 6 is a scaling factor that ensures the posterior probabilities sum to unity, therefore it can safely be ignored while comparing posterior probabilities. The Bayesian or minimum-error-

rate classification decision rule simply assigns an observation to the class that has the highest posterior probability:

$$\text{Class of } x = \text{argmax}_k P(k|x) \quad (7)$$

Then, the following discriminant function which is based on the natural logarithm transform of posterior probabilities, can be maximized to assign samples to classes:

$$g_k(x) = \ln p(x|k).\ p(k) = \ln p(x|k) + \ln p(k) \quad (8)$$

We note that the prior probabilities bias the decisions in favor of the more likely classes. If the class conditional densities p(x|k) are multivariate normal, that is:

$$p(x|k) = \frac{1}{(2\pi)^{\frac{d}{2}} \det(\sum_k)^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(x-\mu_k)^T \sum_k^{-1}(x-\mu_k)\right\} \quad (9)$$

where, μ_k is the d×1 mean vector and Ó$_k$ is the d×d covariance matrix. Now the discriminant function can be shown as follows:

$$g_k(x) = -\frac{1}{2}[d\ln(2\pi) + \ln(\det(\Sigma_k)) + (x-\mu_k)^T \Sigma_k^{-1}(x-\mu_k)] + \ln p(k) \quad (10)$$

Further details of the Bayesian theory, including a discussion of the case where different classes have different covariance matrices, can be found in Duda and Hart (1973).

Neural networks: Back-propagation neural networks (NNs) are feed-forward neural networks with signals propagated only forward through the layers of units. These networks are comprised of (1) an input layer of units, which we feed with gene expression data; (2) hidden layer(s) of units and (3) an output layer of units, one for each diagnostic category, so-called 1-of-n encoding (Michell, 1997). The connections among units have weights and are adjusted during the training phase by back-propagation learning algorithm. This algorithm adjusts weights by propagating the error between network outputs and true diagnoses backward through the network and employs gradient descent optimization to minimize the error function. This process is repeated until we find a vector of weights that best fits the training data. When training of a neural network is complete, unseen data instances are fed to the input units, propagated forward through the network and the network outputs classifications.

Decision trees: Different methods exist to build Decision Trees (DT), which summarize given training data in a tree structure, with each branch representing an association between attribute values and a class label. The most famous and representative amongst these is, perhaps, the C4.5 algorithm (Quinlan, 1993). It works by recursively partitioning the training data set according to tests on the potential of attribute values in separating the classes.

Gene expression datasets: In this study, 9 publicly available microarray datasets, encompassing 5 binary and 4 multiclass classification problems are exploited. Details of the datasets describing the dimensionality of each study as well classification task are listed in Table 1.

Data preprocessing is an important step for handling gene expression data. This includes two steps: filling missing values and normalization. For both training and test dataset, missing values are filled using the average value of that gene. Normalization is then carried out so that every observed gene expression has mean equal to 0 and variance equal to 1. In summary, the 9 datasets had 2-26 distinct diagnostic categories, 22-308 samples and 2000-15154 genes after the data preparatory steps outlined above.

Gene selection algorithms: As it was already mentioned, in gene expression data the number of features (genes) is usually very high. Therefore, there is a pressing need for techniques capable of selecting and ranking the appropriate genes from among the entire set of microarray

Table 1: Cancer related gene expression datasets used in present study

Dataset	n	d	C	Diagnostic task
Lung_Cancer (Bhattacherjee *et al*., 2001)	203	12600	5	4 lung cancer types and normal tissues
Prostate_Cancer (Singh *et al*., 2002)	102	10509	2	Prostate cancer and normal tissues
Breast_Cancer (Lee *et al*., 2003)	22	3226	2	Breast cancer and normal tissues
Lukemia (Golub *et al*., 1999)	72	7129	2	Acute lympboblastic and acute Myelogenous
SRBCT (Yeung *et al*., 2001)	83	2308	4	Small, round blue cell tumors of childhood
Brain_Tumor (Pomeroy *et al*., 2002)	90	5920	5	5 human brain tumor types
Colon_Cancer (Alon *et al*., 1999)	62	2000	2	Colon cancer and normal tissues
Ovarian_Cancer (Shital *et al*., 2007)	253	15154	2	Ovarian cancer and normal tissues
14_Tumors (Ramaswamy *et al*., 2001)	308	15009	26	14 human tumor and 12 normal tissue types

n: No. of samples, d: No. of genes (features), C: No. of classes

data. Below, is a brief introduction to the feature selection via information gain-based ranking (Ryu and Cho, 2002), relief algorithm (Wang and Makedon, 2004), t-statistics and dimensionality reduction via principal component analysis (Devijver and Kittler, 1982), which are employed in present comparative studies.

Information gain: Suppose that a gene expression pattern is represented as g_i (for example i = 1-12600 in lung_cancer data set). Each g_i is a vector of gene expression levels from N samples, $g_i = (e_1, e_2, \ldots, e_n)$, while c_j represents a class sample j where j = 1-N (Ryu and Cho, 2002). If the number of genes excited ($P(g_i)$) or not excited ($P(\bar{g}_i)$) in class ($P(c_j)$) is counted, the coefficient of the of the information gain (IG) becomes:

$$IG(g_i, c_j) = P(g_i, c_j)\log\frac{P(g_i, c_j)}{P(c_j).P(g_i)} + P(\bar{g}_i, c_j)\log\frac{P(\bar{g}_i, c_j)}{P(c_j).P(\bar{g}_i)} \quad (11)$$

Relief algorithm: The basic idea of Relief Algorithm (RA) is to draw instances at random, compute their nearest neighbors and adjust a gene weighting vector to give more weight to genes that distinguish this set from neighbors of different classes. Specifically, it tries to find a good estimate of the following probability in order to assign a weight for each gene:

W_{gi} = P(different value of g_i | different class) -P(different value of g_i | same class) (12)

t-statistics: An often used feature selection method is based on t-statistics (TA) to measure the class predictability of genes for two-class problems. Here, we compute t-statistics based on distinguishing one class from the rest. This method starts with a data set S consisting of m expression vectors: $X^i = (x_1^i, \ldots, x_n^i)$, where, $1 \le i \le m$, m is the number of samples and n is the number of features measured. Each sample is labeled with $Y \in \{+1,-1\}$. Now, for each feature x_j, the mean μ_j^+ (μ_j^-) and the standard deviation σ_j^+(σ_j^-) using only the samples labeled +1 (-1) are calculated. Then a score $T(x_j)$ can be obtained as follows:

$$T(x_j) = \frac{|\mu_j^+ - \mu_j^-|}{\sqrt{\frac{(\sigma_j^+)^2}{n_+} + \frac{(\sigma_j^-)^2}{n_-}}} \quad (13)$$

where, n+ (n-) is the number of samples labeled as +1 (-1). When making selection, we take those features with the highest scores as the most discriminatory features.

Principal component analysis: Principal Component Analysis (PCA) is a standard statistical technique that can be used to reduce the dimensionality of a data set (Duda and Hart, 1973). This is done by projecting the data of a dimensionality N onto the eigenvectors of their covariance matrix with, usually, the largest M eigenvalues taken. Formally, each so-called principal component PC_i; i = 1, 2, …, M is obtained by linearly combining the original attributes (or features) such that:

$$PC_i = \sum_{j=1}^{M} e_{ij}x_j \quad (14)$$

where, x_j is the jth original feature and e_{ij} are the linear eignvectors which are chosen so as to make the variance of the corresponding PC_i as large as possible. The resulting eigenvectors can be ranked according to the amount of variation in the original data that they account for.

RESULTS AND DISCUSSION

In this study, various classifier models are used to accomplish classification by mapping gene feature vectors of a different dimensionality onto their underlying functional class types. For our evaluation of different classification models we utilized 3 types of SVM (linear, polynomial kernel with exponent 2 and RBF kernel), 3 types of KNN classifiers (K = 3, 5, 7, 9), a NB classifier, a back-propagation NN classifier with one hidden layer and a C4.5 decision tree. The SVMs were trained using the SMO algorithm (Platt, 1999) and employed the pairwise classification strategy for multiclass classification.

The classification performance was measured using 4-fold cross validation technique. That is the gene expression vectors are randomly partitioned into 4 equally-sized subsets and each subset is used as a test set for a classifier trained on the remaining 3 subsets. We experimented with 3 filter-based feature selection methods i.e., IG, RA, TA and a PCA based feature extraction algorithm. When using filter methods, the number of selected genes must be pre chosen. Thus, we chose a rather broad selection of feature sets with: 25, 50, 100, 250, 500 and 1000 top ranked features.

The optimum results of classification experiments without gene selection and in terms of classification accuracy as a performance metric are shown in Table 2. As can be seen, in 8 out of 9 datasets, SVMs perform cancer diagnoses with accuracies > 85%.

Overall, SVM classifiers outperform KNN, NB, NN and DT significantly in all cases. The superior performance of SVM-based methods compared to KNN, NB, NN and DT reflects that SVM classifiers are less

Table 2: Cancer classification accuracies (%) without gene selection using 4-fold cross validation (the last column reports average performance computed over all datasets)

	Dataset									
Classifier	Lung	Prostate	Breast	Lukemia	SRBCT	Brain	Colon	Ovarian	14 Tumors	Average
SVM	**85.1**	**93.0**	**89.4**	**90.6**	**96.3**	**86.5**	**98.5**	**91.8**	**81.7**	90.3
KNN	76.6	85.3	80.0	81.4	84.8	83.7	86.0	89.3	78.5	82.8
NB	71.4	80.4	72.4	75.6	84.3	75.0	82.4	83.2	70.2	72.2
NN	60.3	75.2	74.0	71.3	79.8	74.8	83.1	85.5	59.4	73.7
DT	55.2	76.4	63.2	73.4	65.6	59.8	67.3	67.8	73.3	66.8

Number in bold correspond to the best classification result for each dataset

sensitive to the curse of dimensionality and more robust to a small number of high-dimensional gene expression samples than other non-SVM techniques. The KNN classifiers are the second best classifiers while NB and NN classifiers represent almost comparable classification results. Finally, DT classifiers produced the worst overall classification accuracy.

To investigate the impact of feature selection on the classification performance, 3 feature selection methods and a feature extraction algorithm were applied to present 9 gene expression datasets defined in Table 1. Figure 1 shown the optimum classification performance using SVM, KNN, NB, NN and DT classifiers in conjunction with the use of the introduced feature selection/extraction schemes. In Fig. 1, each bar indicates the classification accuracy using a different classifier in addition to both the optimum number of original selected features and the best gene selection method. For the sake of comparison, Fig. 1 also shows our best classification results when no feature selection was applied (Table 1).

As expected, the performance of the classifiers differs among the datasets. Indeed, the overall performance on the multiclass datasets (e.g., 14_Tumor and Lung) is lower than the performance on the binary class (e.g., Colon and Prostate) datasets.

In Fig. 1, for example the left-most case in Lung dataset represent both SVM-based classification accuracy (85.1%) when no feature selection method was involved and an optimum SVM-based classification accuracy (92.5%) that was achieved using a IG-based feature selection method which chose 250 features from all the initial features. The results also show that gene selection significantly improves the classification performance of non-SVM classifiers. In particular, for some datasets, classification accuracy is improved by up to 26.5, 15.4, 12.4 and 8.5% for NN, DT, NB and KNN, respectively (Table 2).

Gene selection also improved the accuracy of SVMs up to 7.8%. It is very interesting to note that the classification ability using selected features could be better than that of using all feature set. In particular, for NN classifier, much better results can be obtained if careful selection of a subset of features is carried out. Importantly, this improvement of performance was obtained by structurally much simpler classifiers, as compared to a classifier that requires the full feature set.

Again, SVM models considerably outperform the other non-SVM classifiers over all datasets. KNN and NN classifiers perform close to each other when the gene selection is applied and tend to outperform the NB and DT classifiers.

Overall, DT classifier did the worst on average amongst the other experimented classifiers. This is probably due to the fact that the other types of classifier work directly on numerically-valued data without the need for discretisation, whilst DT learning requires partitioning the underlying domain into a set of symbolic values (even though they may be represented as real-valued intervals).

The variation among the filter methods is very low. IG has the lowest mean error rate of 0.1 against all datasets (Fig. 2). The other two filter methods have slightly higher mean error rates of 0.12 and 0.15. The PCA mean error rate was the highest and measured to 0.2.

The fundamental advantage of preserving feature meanings through the use of a strict feature selection approach like information gain-based ranking, over the use of a transformation-based approach such as PCA, has been pointed out earlier. The effects of using either approach are experimentally evaluated in terms of classification accuracy. As can be seen from Fig. 1 and 2, the classifiers using the features selected via feature selection approaches have a higher classification accuracy or equivalently less mean error rate (Fig. 2).

The impact of the size of the features set on the performance of the filter feature selection methods was also analyzed. The plot in Fig. 3 shows that the performance of the filter methods is strongly dependent on the size of the feature selection, while the choice of the method itself makes only marginal difference. The results were generally better for feature selections between 100 and 500 genes. For IG and TA methods, feature selections of 250 genes have the lowest mean error rate while for the RA and PCA the optimum number of genes was obtained equal to 100 and 500, respectively. On the other hand, for feature selections smaller than 100 or larger than 500 genes, the performance was generally reduced.

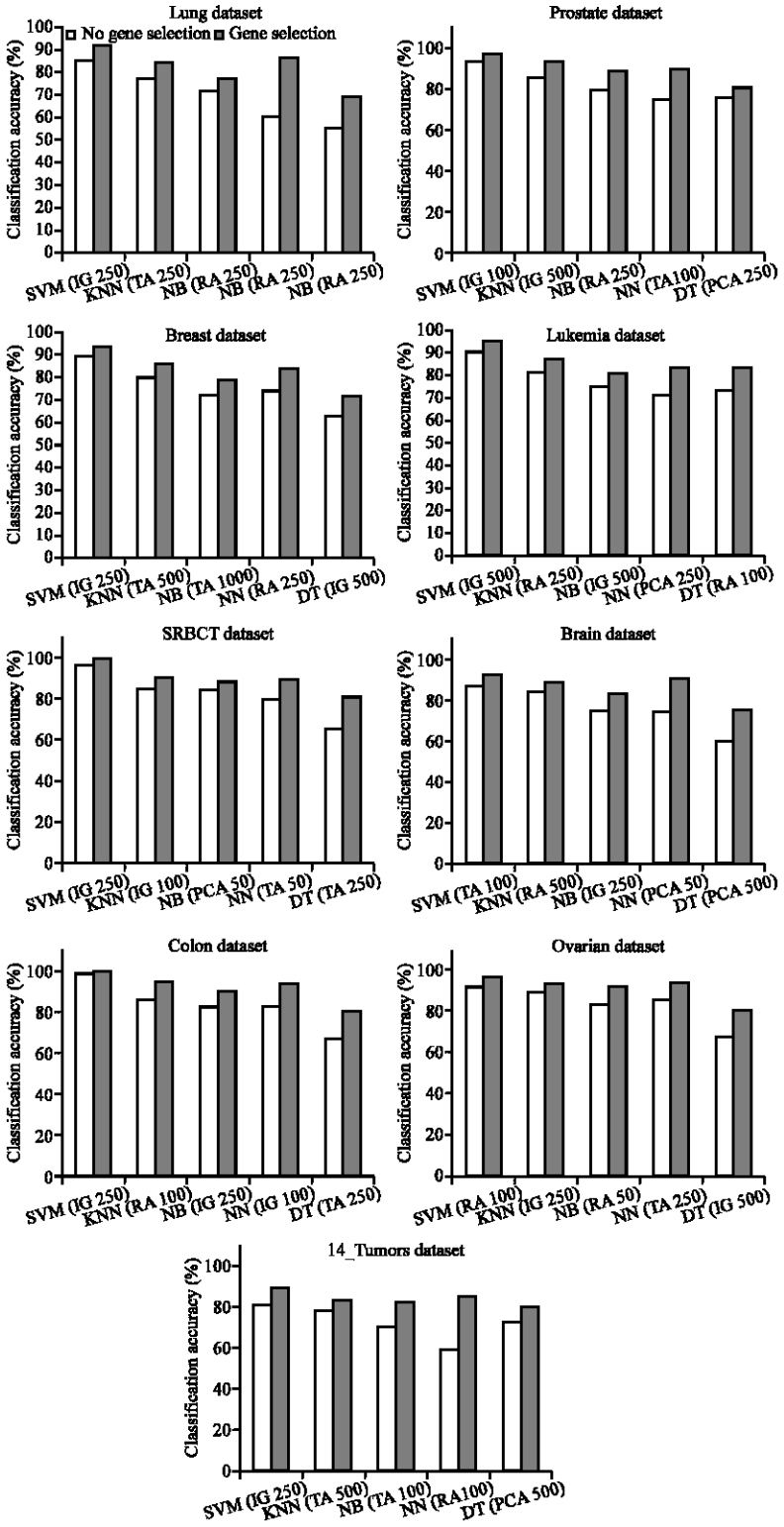

Fig. 1: Performance results (accuracies) of the classification experiments with gene selection obtained using 4-fold cross validation for 9 datasets introduced in Table 1. The white/grey bars correspond to the classification results without/with gene selection. The text below each bar indicates the optimal combination of gene selection method and the number of genes for a specific classifier

Table 3: Optimum cancer classification accuracies (%) with gene selection and using 4-fold cross (the last column reports average performance computed over all datasets)

	Dataset									
Classifier	Lung	Prostate	Breast	Lukemia	SRBCT	Brain	Colon	Ovarian	14 Tumors	Average
SVM	**92.5**	**97.1**	**93.4**	**95.2**	**99.5**	**92.1**	**99.9**	**96.8**	**89.5**	95.1
KNN	84.1	93.1	85.6	87.3	90.3	88.5	94.5	93.5	84.0	88.9
NB	77.0	88.9	79.1	80.4	88.2	83.4	90.0	92.1	82.6	84.6
NN	86.5	90.3	85.2	83.8	89.6	91.0	93.8	94.6	85.9	88.5
DT	69.0	81.0	72.0	83.6	81.0	75.5	80.6	81.0	80.2	78.2

Number in bold correspond to the best classification for each dataset

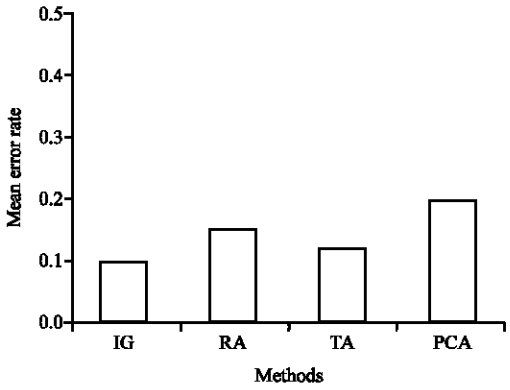

Fig. 2: Feature selection/extraction performance of 4 different methods i.e., information gain, relief algorithm, t-statistics and principal component analysis in terms of mean error rate over all 9 gene expression datasets

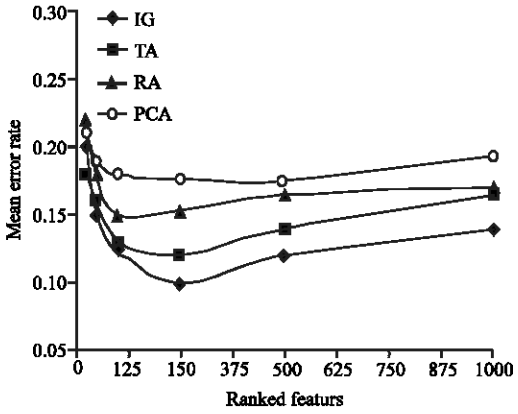

Fig. 3: Mean error rate as a function of feature selection/extraction method and target feature set size

We also found that KNN and NN methods are not statistically significant (based on paired t-test) from each other and NN at the 0.05 level while non-SVM algorithms have statistically significant poorer performance than the SVMs. As we have empirically noticed, the non-SVM methods NN, DT, NB and KNN benefit significantly more than SVMs from gene selection. A number of observations can explain this behavior: in high-dimensional spaces, KNN has high variance of the prediction since all the training points are located close to the edge of the sample.

Furthermore, many irrelevant variables in the data dominate distances between samples which present a significant problem for the prediction. NB encounters problems similar to KNN, in particular because they rely on Euclidean or Mahalanobis distance for density estimation that generally require exponential sample to the data dimensionality.

NNs are sensitive to high dimensionality for at least two reasons: first, note the larger the number of variables, the larger is the number of weights in this type of neural network. Because of this, (1) there may be more local minima in the error landscape and it is thus more probable for back-propagation to get trapped in one of them and (2) the model space becomes exponentially larger with the addition of each weight and therefore, it becomes harder to identify a model that generalizes. In comparison, SVMs seem relatively insensitive to the curse of dimensionality, possibly due to the specific regularization mechanism they employ.

Present best classification results reinforce the observation that using SVM in conjunction with information gain-based feature selection approach generally leads to a better classification for the underlying datasets. These results are indicative of the power of feature selection in helping to reduce redundant feature measures and also the noise associated with such measurement. This also shows that information loss can be minimized and even avoided in building the classifiers if feature selection process is carefully carried out.

CONCLUSION

In this study, we have conducted a comprehensive study of both classification methods as well as feature selection methods for classification of several cancer-related gene expression human datasets publicly available.

The study is itself novel as feature selection methods are for the first time employed in conjunction with the learning process of each classifier considered to address the difficulties in handling real problems represented by various gene expression datasets. It has shown that in general, feature selection/extraction is beneficial for improving the performance of these common learning algorithms. It has also revealed that, as with the learning algorithms, there is no single best approach for all situations involving dimensionality reduction or feature selection.

The research focused on identifying the best combination of classifier and feature selection strategy. In particular, this work has investigated the following 5 classification algorithms: support vector machine, K nearest neighbour, neural networks, naive Bayes and decision trees and the following 4 methods for making choice of features: information gain, relief algorithm, t-statistics and principal component analysis.

Comparative studies have been performed between the use of full feature sets and that of a subset; between the employment of different types of learning algorithm in building classifiers and between the utilization of dimensionality reduction techniques that choose features in the strict sense (i.e., not altering any form of the original features) or through transformation (i.e., changing the representation of original features). Finally, amongst a large number of experimental studies carried out, the best classification accuracy is achieved by using a subset of 250 features chosen by IG based method for SVM classifier. The KNN and NN classifiers do well and come next to SVMs. The NB classifiers achieved less classification accuracy than the aforementioned models and the DT classifiers are the worst model in the field.

REFERENCES

Alon, U., D. Barkai, A. Notterman and K. Gish, 1999. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proc. Natl. Acad. Sci., 96: 6745-6750.

Bhattacherjee, A., W. Richards, Staunton J. and C. Li, 2001. Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses. Proc. Natl. Acad. Sci., 98: 13790-13795.

Braga-Neto, U. and E. Dougherty, 2004. Is cross-validation valid for small sample microarray classification? Bioinformatics, 200: 374-380.

Bylander, T., 2002. Estimating generalization error on two-class datasets using out-of-bag estimates. Mach. Learn., 48: 287-297.

Devijver, P. and J. Kittler, 1982. Pattern recognition: A statistical approach. 1st Edn., Prentice Hall, London.

Duda, R.O. and P.E. Hart, 1973. Pattern classification and scene analysis. 2nd Edn., John Wiley and Sons, New York, ISBN: 0-471-22361-1, pp: 482.

Freund, Y. and R. Schapire, 1997. A decision theoretic generation of on-line learning and application to boosting. J. Comput. Syst. Sci., 55: 119-139.

Furey, T., N. Cristianini, N. Duffy and D. Bednarski, 2000. Support vector machine classification and validation of cancer tissue Samples using microarray expression data. Bioinformatics, 16: 906-914.

Golub, T., D. Slonim and C. Huard, 1999. Molecular classification of cancer: Class discovery and class Prediction by gene expression monitoring. J. Sci., 286: 531-537.

Guyon, I., J. Weston and S. Barnhill, 2002. Gene selection for cancer classification using support vector machines. J. Mach. Learn., 46: 389-422.

Hedge, P., 2000. A concise guide to cDNA microarray analysis. Biotechniques, 29: 548-550.

John, G. and R. Kohavi, 1994. Irrelevant features and the subset selection problem. 11th International Conference on Machine Learning, 1994, Morgan Kaufmann Publishers, pp: 121-129.

Khan, J., J. Wei, M. Ringner and L. Saal, 2001. Classification and diagnosis of cancers using gene expression and artificial neural networks. J. Nat. Med., 7: 673-679.

Kuramochi, M. and G. Karypis, 2005. Gene classification using expression profiles: A feasibility study. Int. J. Artif. Intell. Tools., 14: 641-660.

Lee, K., N., Sha, E. Dougherty and E. Vannucci, 2003. Gene classification: A Bayesian variable selection approach. Bioinformatics, 19: 90-97.

Leng, X. and H. Muller, 2006. Classification using functional data analysis for temporal gene expression data. Bioinformatics, 22: 68-76.

McLachlan, G., R., Bean and D. Peel, 2002. A mixture model based approach to the clustering of microarray expression data. Bioinformatics, 18: 413-422.

Michell, T., 1997. Machine Learning. 1st Edn., McGraw-Hill, ISBN: 0-07-115467-1.

Mohd, S.M., O. Sigeru and D. Safaai, 2007. A model for gene selection and classification of gene expression data. Artif. Life. Robot., 11: 219-222.

Mukherjee, S. and S. Roberts, 2004. A theoretical analysis of gene selection. Proceedings of the IEEE Computer Society Bioinformatics Conference. August 16-19, IEEE Computer Society Washington, pp: 131-141.

Ntzani, E., 2003. Predictive ability of DNA microarray for cancer outcomes and correlates: And empirical assessment. Lancet, 362: 1439-1444.

Platt, J., 1999. Fast Training of Support Vector Machines Using Sequential Minimal Optimization Advances in Kernel Methods, Support Vector Learning. 1st Edn., MIT Press, Cambridge, ISBN: 0-262-19416-3.

Pomeroy, S., P. Tamayo, M. Gaasenbeek and L. Sturla, 2002. Prediction of central nervous system embryonal tumor outcome based on gene expression. Nature, 415: 436-442.

Quinlan, J., 1993. Programs for Machine Learning. 1st Edn., Morgan Kaufmann, San Francisco, ISBN: 1-55860-238-0.

Ramaswamy, S., P. Tamayo, R. Rifkin and S. Mukherjee, 2001. Multiclass cancer diagnosis using tumor gene expression signature. Proc. Natl. Acad. Sci., 98: 15149-15154.

Ryu, J. and S. Cho, 2002. Towards optimal feature and classifier for gene expression classification of cancer. Lecture Notes Comput. Sci., 2275: 310-317.

Scott, D., 1992. Multivariate Density Estimation: Theory, Practice and Visualization. Wiley Series in Probability and Mathematical Statistics. 1st Edn., John Wiley and Sons, New York, ISBN: 978-0-471-54770-9.

Shital, S. and A. Kusiak, 2007. Cancer gene search with data mining and genetic algorithms. Comput. Biol. Med., 37: 251-261.

Singh, D., P. Febbo, D. Jackson, , J. Manola and C. Ladd, 2002. Gene expression correlates of clinical prostate cancer Behavior. Cancer Cell, 12: 203-209.

Vapnik, V.N., 1998. Statistical Learning Theory. 1st Edn. John Wiley and Sons, New York, ISBN: 0471030031.

Wang, Y. and F. Makedon, 2004. Application of relief feature filtering algorithm to selecting informative genes for cancer classification using microarray data. Proceedings of the IEEE Conference on Computational Systems Bioinformatics Computer Society Bioinformatics, (CSB'04), IEEE Computer Society, pp: 497-498.

Xiong, M., L. Jin and W. Li, 2000. Computational method for gene expression based tumor classification. Biotechniques, 29: 1264-1270.

Yeung, K., C. Fraley, A. Murua and A. Raftery, 2001. Model based clustering and data transformations for gene expression data. Bioinformatics, 17: 977-987.

Yeung, K., R. Bumgarner and A. Raftery, 2005. Bayesian model averaging: Development of an improved multi-class gene selection and classification tool for microarray data. Bioinformatics, 21: 2394-2402.

Zhang, H., J. Ahn and C. Park, 2006. Gene selection using support vector machine with non-convex penalty. Bioinformatics., 22: 88-95.