European Journal of Academic Essays, Special Issue (1): 18-26, 2014
ISSN: 2183-1904
www.euroessays.org

Recent Trends in Analysis of Algorithms and Complexity Theory

Analysis Two Stage Feature Selection for Classification of Microarray Databases

Dash, Rasmita[1], Misra, Bijan Bihhari[2]

[1]Department of Computer Sc. & Information TechnologyInstitute of Technical Education and Research,Siksha ,O' anusandhan Universuty,Khandagiri Square, Bhubaneswar
rasmitadash@soauniversity.ac.in

[2]Department of Computer Sc. & Engineering,Silicon Institute of Technology, Bhubaneswar-751024, Odisha, India
misrabijan@gmail.com

Abstract: Classification of microarray data with high dimension and small sample size is a complex task. This work explores the optimal search space appropriate for classification. Here the crush of dimensionality is handled with a two stages feature selection technique. At the first stage, statistical measures are used to remove genes that do not contribute for classification. In the second stage, attributes are arranged in a non-increasing order of signal to noise ratio (SNR). The number of attributes to be considered from this stage is a challenging task. Selection of very few numbers of genes may not help representing the properties of each class properly, while selection of a large numbers of genes also leads to diminishing classifier performance. A set of four classifiers such as artificial neural network, naïve Bayes, k-nearest neighbor, multiple linear regression classifier and ten microarray databases are considered for this experimentation. For each microarray data and for each classifier, selection for different set of genes improves the performance. However, the analysis shows that a favorable set of genes can be suggested to improve performance and another set of genes can be suggested to avoid which degrades the performance.

Keywords: Microarray, Classification, feature selection.

1. Introduction

Classification of microarray data is one of the important tasks of microarray data analysis Given a set of previously classified examples, (for example, different types of cancer classes such as AML and ALL), a classifier finds a rule that allows to assign new samples to one of the above classes [1]. To design a classifier model, sufficient number of sample is required to train the model. But in microarray data the number of samples are much less (within hundred) in comparison to the number of attributes (in the range of tens of thousands) [2]. The analytical precision of such high dimensional data gets influenced by its dimensions. It is therefore highly essential to reduce the dimension/genes of the dataset in such a way that the remaining genes can contribute significantly in designing the classifier. In fact prior to developing any type of model for any category of task of microarray data analysis, dimensionality reduction is almost mandatory. Feature selection (variable elimination) helps in understanding data, reducing computation requirement, reducing the effect of curse of dimensionality and improving the predictor performance and produce a list of genes whose expression is known as differentially expressed genes. So identification of differential gene expression is the first task for microarray analysis [3].

Several feature selection techniques are developed to address the problem of reducing genes but identifying an appropriate feature reduction technique is challenging in microarray data analysis. This is due to the presence of enormous number of genes compared to the number of samples. Some of the genes have strong relevance (always represents the optimal subset), some other genes have weak relevance (sometimes represents the optimal subset) and rest of the gens are irrelevant that are not at all required. Therefore, it is significant to extract the informative genes from the original data. Gene expression data may contain thousands of genes that are highly correlated, one feature out of the several correlated feature is good enough to represent the data. The dependent variables provide no extra information about the classes and thus serve as noise for the predictor. This means that the total information content can be obtained from fewer unique features which contain maximum discrimination information about the classes. Hence by eliminating the dependent variables, the amount of data can be reduced. If the original data is experimented with the classifier, then its performance degrades drastically [4], [2]. In some applications, variables which have no correlation to the classes serve as pure noise might introduce bias in the predictor and are not encouraged to allow to the classifier. This can happen when there is a lack of information about the process being studied. By applying feature selection techniques we can gain some insight into the process and can improve the computation requirement and prediction accuracy. So before classification

Corresponding Author: Bijan Bihari Misra, Department of Computer Sc. & Engineering,Silicon Institute of Technology, Bhubaneswar-751024, Odisha, India, *misrabijan@gmail.com* Tel-+919437400307

identification of relevant or significant genes is important. Therefore an ensemble feature reduction and classification model can be used for efficient processing of microarray data. Further, in case of unsupervised feature reduction, it is very difficult to decide the optimal size of the features which should be preserved for designing efficient models for classification.

In this paper a two stage feature selection technique for microarray data classification is used [5] with an objective to determine suitable set of features that maximizes the efficiency of the classifier.

Rest of the paper is organized as follows. Recent work on microarray data analysis is described in Section 2. Two stage feature selection approach is explained at Section 3. Basics of different classifiers used are presented at Section 4. Simulation results and analysis is presented in Section 5. The paper is concluded with Section 6.

2. Literature Study

Several attempts were made by researchers to improve the effectiveness and efficiency of the classifier for microarray data analysis. In many literatures it is observed that the classification accuracy falls due to the high dimension of microarray data. Here the number of available training sample is too small as compared to the number of genes and processing of such data leads to high computational cost and memory usage. It is pointed out at [6] that reduction of dimension (or genes) is prerequisite for classification in microarray datasets. The microarray data often contains noisy information and the sources of noise are multiple. A hybridized SNR/SVM approach is proposed in [7] for leukemia data classification. Here SNR is used to eliminate the noisy data and shows accuracy with 94.1%. Different methods were proposed to extract the gene expression module to identify the functionally related genes for microarray data [8-10]. One common method, principal component analysis (PCA), provides a representation of microarray data in terms of a set of linearly uncorrelated axes [11, 12]. While PCA can identify interesting biological information, its linear transformation involves only second order statistics, PCA may miss more complex relationships between genes [13]. For tumor classification D. V. Nguyen, and D. M Rocke, [14] show the efficiency of Partial least square over PCA for high dimensional data reduction. Kim et al. [15] proposed a novel method based on an evolutionary algorithm (EA) to assemble optimal classifiers and improve feature selection. Here EA is improved with a better encoding scheme for chromosome and produced a competitive result as compared to other feature selection-classifier pair. Patil & Kumaraswamy [16] proposed an artificial neural network based approach for heart attack prediction. They applied K-Means algorithm to extract the informative data for prediction. Resul & Abdulkadir [17] proposed a heart disease diagnosis method using neural network and produces 89.01% accuracy and sensitivity and specificity value 80.95% and 95.91% respectively. In [18] a sequential feature selection method (a multistage regression technique) is used and is applied to the Naïve Bayesian network to classify different microarray data. Other similar works for microarray data classification are an evolutionary approach [19] using Genetic Algorithm (GA) and k-NN gene reduction and classifier combination for colon data analysis, gene markers identification through Neural Networks [20], chronological feature extraction approach through Naive Bayes [21].

3. Two Stages Feature Selection Scheme

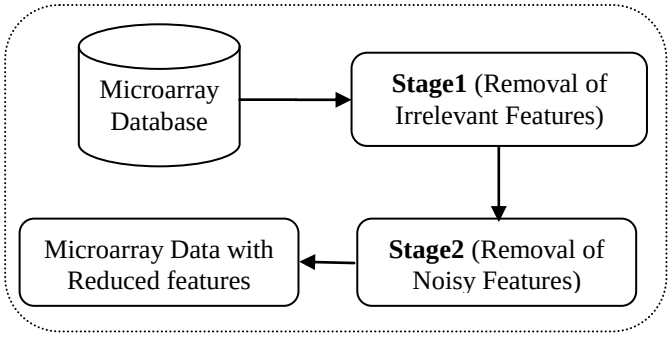


Figure 1: Two stage feature selection model

In two dimension feature selection technique, the number of genes in microarray data is reduced in two different stages.

Stage1:

In the 1st stage on analysis of the datasets, it is observed that the values of the genes in some of the attribute are highly similar irrespective of the class labels. The dissimilarity in the feature value belonging to different class contributes to design any classifier model. Therefore less dissimilar genes can be considered less important for the task of classifier design and may be excluded. In this stage the standard deviation of the attribute is evaluated. The attributes with high dissimilarity are allowed to remain in the database and other attributes are removed. Comparing the standard deviation values of different attributes, a threshold value δ is fixed. All the genes having standard deviation value less than that of threshold value are eliminated from the microarray dataset. The remaining database is referred as dimensionally reduced microarray database1 (DRMD1).

Stage2:

After removal of irrelevant genes at stage1, it is observed that DRMD1 contains significant number of attributes which can be considered as noise for developing a robust classifier model. So in the 2nd stage signal to noise ratio (SNR) technique is used to measure the level of desired signal to the level of background noise. The SNR score identifies the expression patterns with a maximal difference in mean expression between two groups and minimal variation of expression within each group [22]. In this method genes are first ranked according to their expression levels using SNR test statistic. The SNR is defined as follows:

$$SNR(j) = \frac{(\mu_-^j - \mu_+^j)}{(\sigma_-^j - \sigma_+^j)}$$

Where SNR(j) is the signal to noise ratio of j^{th} attribute, μ_-^j and μ_+^j is the mean value of jth attribute that belongs to negative class and positive class respectively. σ_-^j and σ_+^j are the standard deviations of jth

attribute for the respective classes. The higher SNR value indicates that more valuable signal than that of the noise. A predefined number of attribute with higher SNR value are selected and the reduced data is termed as dimensionally reduced microarray database2 (DRMD2).

4. Classification Techniques

Many machine learning methods have been introduced into microarray classification to attempt to learn the gene expression data pattern that can distinguish between different classes of samples in the recent years. This methodology has two phase (i) dimensionality reduction using 3 stage feature reduction techniques and (ii) evaluation of effectiveness of the feature reduction method using different classification algorithm. The 1st phase is already discussed in section 3.In the 2nd phase we applied four different classification algorithm ANN, MLR, Naïve Bayesian Network and k-NN and obtained result with different classification measures. The classification result is used to compare all feature reduction and classifier pair.

4.1 Multiple Linear Regression (MLR)

Multiple linear regression play a pivotal role in the literatures of adaptive control, adaptive signal processing, regression and statistics.

Given a data set $\{y_i, x_{i1}, \cdots, x_{ip}\}, 1 \le i \le n$ statistical units, a linear regression model assumes that the relationship between the dependent variable y_i and the p-vector of regressors xi is linear. This relationship is modeled through a disturbance term or error variable ε an unobserved random variable that adds noise to the linear relationship between the dependent variable and regressors. Thus the model takes the form between the dependent variable and regressor [23]. Thus the model takes the form.

$$y_i = \beta_1 x_{i1} + \cdots + \beta_p p + \varepsilon_i = x_i^T \beta + \varepsilon_i, i = 1, \cdots, n$$

Where T denotes the transpose, so that $x_i^T \beta$ is the inner product between vectors x_i and β. Often these n equations are stacked together and written in vector form as

$$y = X\beta + \varepsilon,$$

where

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, X = \begin{pmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_n^T \end{pmatrix} = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ x_{21} & \cdots & x_{2p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix}, \beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix}, \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

Multiple linear regressions are the simplest and thus most common estimator. It is conceptually simple and computationally straightforward. It estimates are commonly used to analyze both experimental and observational data. This method minimizes the sum of squared residuals, and leads to a closed form expression for the estimated value of the unknown parameter β

$$\beta = \left(X^T X\right)^{-1} X^T Y = \left(\frac{1}{n}\sum_{i=1}^{n} x_i x_i^T\right)^{-1} \left(\frac{1}{n}\sum_{i=1}^{n} x_i y_i\right)$$

The estimator is unbiased and consistent if the errors have finite variance and are uncorrelated with the regressor.

4.2 Artificial Neural Network

Back Propagation Artificial Neural Network is one of the most widely used neural networks. It is a single-direction multilayer neural network that contained of input layer nodes, output layer nodes and one or more layers of hidden nodes. The information transfer from nodes to nodes of different layers and the degrees of the connections are controlled by the connection weights. The network is trained using back propagation algorithm and optimizes connection weights. It always uses the sigmoid activation function in the hidden layers and output layer. The error in the network is calculated as

$$e = \sum_{i=1}^{n} (y_a - y_o)^2$$

Where e is the error in the network, y_a and y_o are actual output and observed output respectively. In classification using ANN, the network is trained with the following steps.

Step1: Initialize of the connection weights of the network;
Step2: Select a sample from the training set data as the input of the network;
Step3: Calculate the output value or output vector of the network;
Step4: Calculate the errors of the network;
Step5: Adjust the connection weights from the output layer back to the input layer;
Step6: Repeat steps3, 4, 5 until the errors are acceptable;
Step7: Select another sample of the training set data and repeat the upper steps until convergence.

4.3 Naïve Bayesian Network

Let $X = (x_1, x_2, \cdots, x_n)$ be an instance for classification, where $(x_1, x_2, \cdots, x_n)$ are the values of attributes $A_1, A_2, \ldots, A_n$ respectively. Naive Bayesian classifiers as sum that attributes are independent when class value C is given, and this is called the conditional independence assumption. The probability of predicting class value C for instance X can thus be calculated as

$$p(X \mid C) = \frac{P(X \mid C) \times P(C)}{P(X)} = P(C) \prod_{j=1}^{n} P(x_j \mid C)$$

The class value C that has the largest classification probability $P(X \mid C)$ will be the predicted class of instance X [24]. The conditional independence assumption is thus essential to the operation of the naive Bayesian classifier [25, 27].

4.4 k-Nearest Neighbor algorithm

k-Nearest Neighbor(kNN) algorithm was first introduced in [37]. For its simplicity yet efficient performance, kNN has been adopted as a classifier in many problems such as data mining, pattern recognition, statistics, and machine learning. The concept of kNN can be adapted to supervised learning algorithms. In the first step, kNN only stores the information of the training data that include data features and their categories. Next, the accuracy of a classifier will be evaluated by using the testing data. A classification is made by measuring the distances from the test instance to all training instances, most commonly using the Euclidean

distance. Finally, the majority class among the k nearest instances is assigned to the test instance.

The algorithm on how to compute the K-nearest neighbors is as follows:

Step1. Determine the parameter K (number of nearest neighbors beforehand).

Step2. Calculate the distance between the query-instance and all the training samples (depending on K value). Any distance algorithm can be used

Step3. Sort the distances for all the training samples and determine the nearest neighbor based on the K^{th} minimum distance.

Step4. Since this is supervised learning, get all the categories of training data for the sorted value which fall under K.

Step5. Use the majority of nearest neighbors as the prediction value of the query instance.

The value of k must be properly determined before the classification process because it has an influence on the classification accuracy. Smaller values of k produce larger variance in classification. Using a large value of k can reduce the effect of the variance, but requires more expensive computation.

5. Experimental Results

For analysis of performance of two stage feature selection, experimental studies are made with 10 gene expression databases. The details of the microarray databases are presented in Table.1.

Table1: Description of datasets used

Dataset	*Genes*	*Samples*		
		In Class1	*In class2*	*Total*
AllAml [32]	7129	27	11	38
Colon Tumor [33]	2000	40	22	62
DLBCL Harvard Outcome [36]	7129	32	26	58
DLBCL Harvard Tumor [36]	7129	58	19	77
DLBCL Stanford [35]	4026	24	23	47
Lung Cancer Michigan[29]	7129	86	10	96
Lung Cancer Ontario[34]	2880	24	15	39
Prostate Tumor [36]	12600	52	50	102
Adcalung[28]	12533	15	134	149
CNS[30]	7129	39	21	60

In microarray data the number of sample is too small in comparison to the number of features in the database. From classification point of view, it cannot be ignored that such huge dimensions are free from redundancy or noise. To remove the less significant features, standard deviation of each feature is calculated and maximum value of standard deviation is determined. In the first stage of feature selection, if standard deviation of a feature is less than or equal to 95% of the maximum standard deviation value then it is removed.

In the second stage, signal to noise ratio is evaluated, with an objective to remove more noisy data. Here features are rearranged and stored in such a way that the first feature possesses maximum signal and minimal noise. Subsequent features arranged in non-increasing order of signal to noise ratio.

Features ranging from one to ten from the reduced and rearranged database in order are considered for this experimentation.

For each feature size of each database, 10-fold cross validation approach is adopted to evaluate the classifier performance. The average performance of 10-fold cross validation is taken as the performance for the specific feature size of the database.

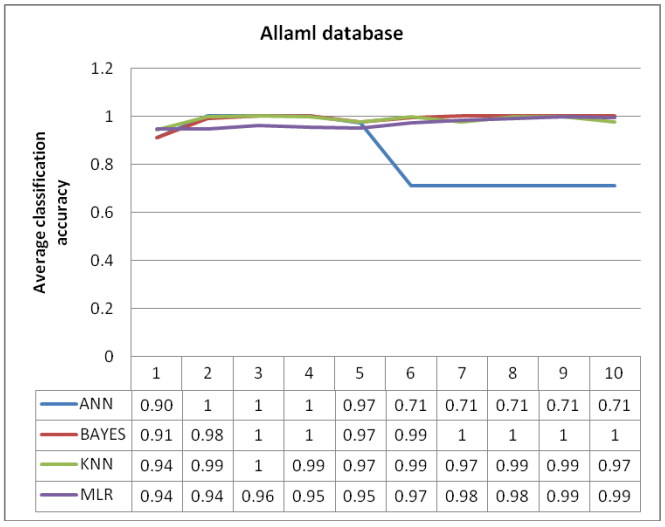


Figure 2: Classification performance of Allaml database with different classifiers for second stage feature selection.

From Figure 2, it is revealed that performance of ANN is best when 2-4 features are selected, but as the number of features increases, the performance falls significantly. Performance of Bayes is the best when 3-4 or 7-10 features are selected, otherwise remains slightly low. Though in all the selection catagories the performance of KNN is good, but attains the best only with combination of first three features.

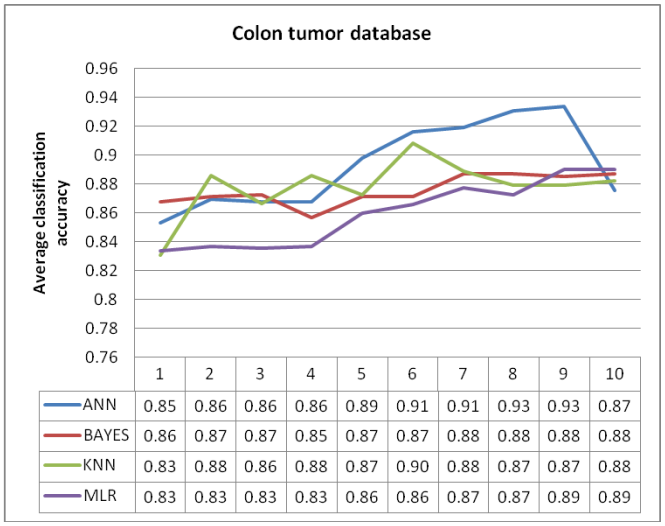
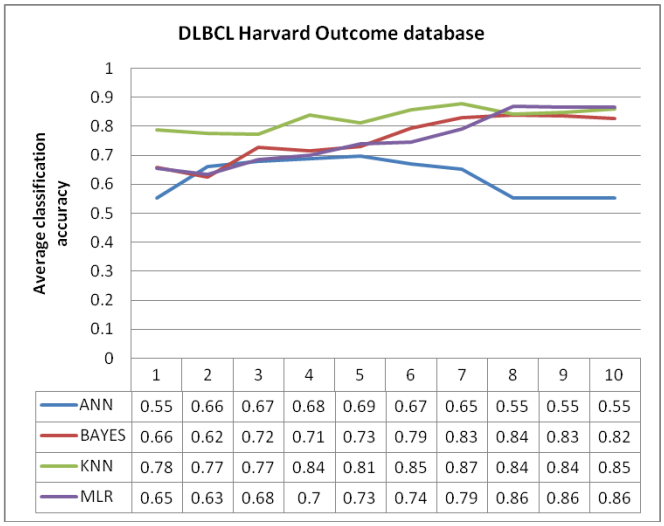


Figure 3: Classification performance of Colon tumor database with different classifiers for second stage feature selection.

It can be seen from Figure 3 that among the 4 classifiers ANN yeilds the best result with 9 features. Performance of

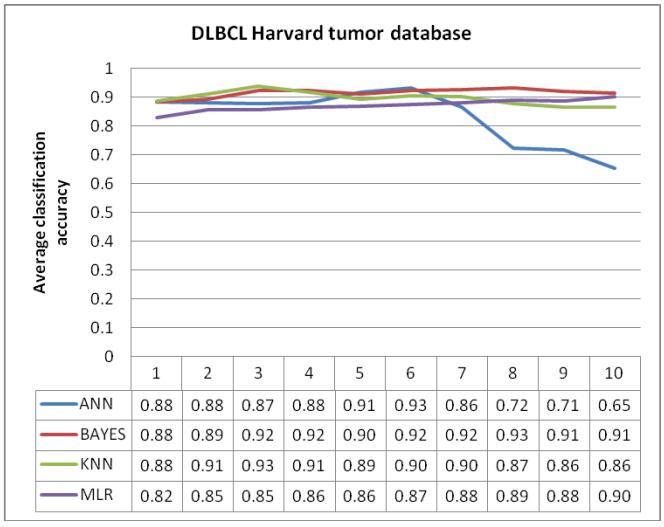
Bayes varies from 0.856 to 0.887 with standard deviation of 0.010, KNN varies from 0.830 to 0.908 with standard deviation of 0.019, and MLR varies from 0.833 to 0.890 with standard deviation of 0.022.



	1	2	3	4	5	6	7	8	9	10
ANN	0.55	0.66	0.67	0.68	0.69	0.67	0.65	0.55	0.55	0.55
BAYES	0.66	0.62	0.72	0.71	0.73	0.79	0.83	0.84	0.83	0.82
KNN	0.78	0.77	0.77	0.84	0.81	0.85	0.87	0.84	0.84	0.85
MLR	0.65	0.63	0.68	0.7	0.73	0.74	0.79	0.86	0.86	0.86

Figure 4: Classification performance of DLBCL Harvard outcome database with different classifiers for second stage feature selection.

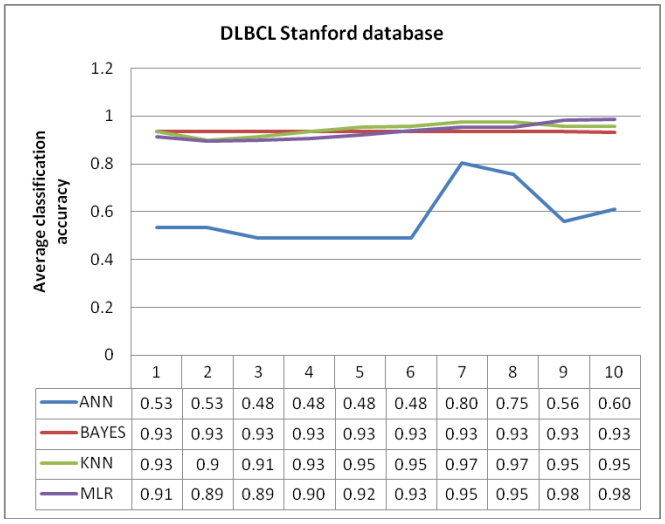
In DLBCL Harvard outcome database, KNN yields the best result of 0.878 with 7 features, followed by MLR with 0.869 at 8 features. Result of ANN is (0.551-0.696)±0.064, of Bayes is (0.625-0.839) ±0.078, of KNN is (0.772-0.878) ±0.037, and of MLR is (0.632-0.869) ±0.089. Details are presented in Figure 4.



	1	2	3	4	5	6	7	8	9	10
ANN	0.88	0.88	0.87	0.88	0.91	0.93	0.86	0.72	0.71	0.65
BAYES	0.88	0.89	0.92	0.92	0.90	0.92	0.92	0.93	0.91	0.91
KNN	0.88	0.91	0.93	0.91	0.89	0.90	0.90	0.87	0.86	0.86
MLR	0.82	0.85	0.85	0.86	0.86	0.87	0.88	0.89	0.88	0.90

Figure 5: Classification performance of DLBCL Harvard tumor database with different classifiers for second stage feature selection.

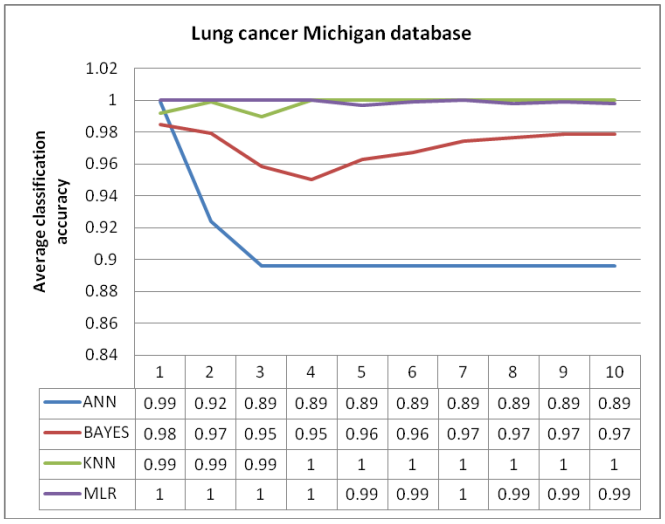
In DLBCL Harvard tumor database, result of ANN fall steadily with more features. KNN yields the best of 0.938 at 3, followed by ANN with 0.932 at 6 and Bayes with 0.9331 at 8. Here again the performance of ANN falls with more features. Results are presented in Figure 5.



	1	2	3	4	5	6	7	8	9	10
ANN	0.53	0.53	0.48	0.48	0.48	0.48	0.80	0.75	0.56	0.60
BAYES	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93	0.93
KNN	0.93	0.9	0.91	0.93	0.95	0.95	0.97	0.97	0.95	0.95
MLR	0.91	0.89	0.89	0.90	0.92	0.93	0.95	0.95	0.98	0.98

Figure 6: Classification performance of DLBCL Stanford database with different classifiers for second stage feature selection.

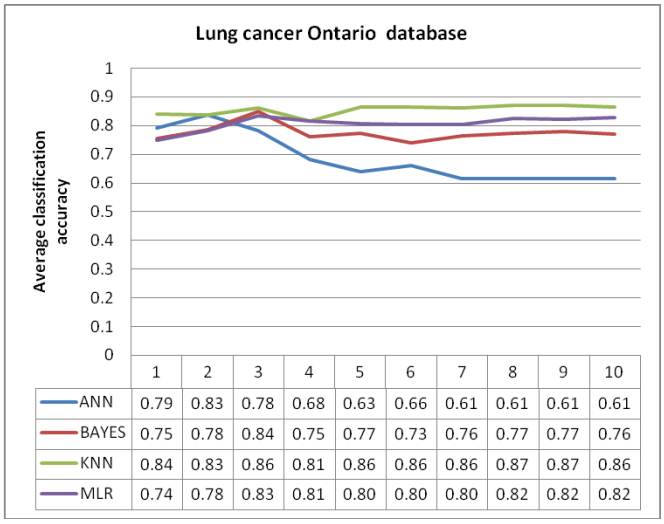
In DLBCL Stanford database, performance of ANN is much below the performance of other classifiers. Performance all other classifiers are competitive. Figure 6 shows the performance of the classifiers.



	1	2	3	4	5	6	7	8	9	10
ANN	0.99	0.92	0.89	0.89	0.89	0.89	0.89	0.89	0.89	0.89
BAYES	0.98	0.97	0.95	0.95	0.96	0.96	0.97	0.97	0.97	0.97
KNN	0.99	0.99	0.99	1	1	1	1	1	1	1
MLR	1	1	1	1	0.99	0.99	1	0.99	0.99	0.99

Figure 7: Classification performance of Lung cancer Michigan database with different classifiers for second stage feature selection.

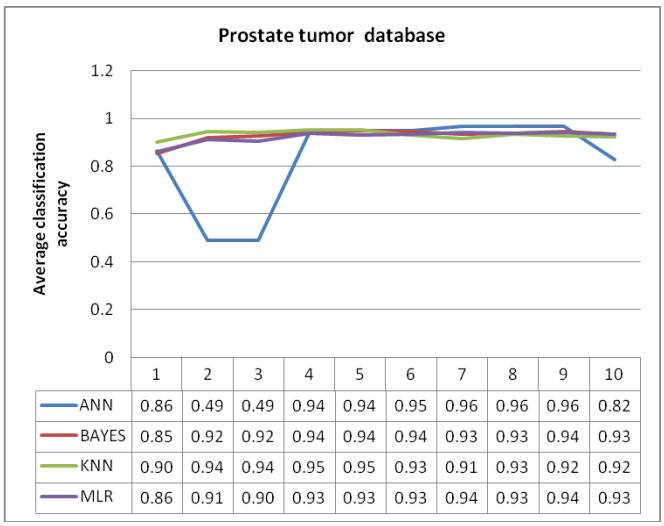
Figure 7 shows the performance of classifiers for Lung cancer Michigan database. The results of KNN and MLR are competitive. ANN and Bayes yield best result with single feature.



	1	2	3	4	5	6	7	8	9	10
ANN	0.79	0.83	0.78	0.68	0.63	0.66	0.61	0.61	0.61	0.61
BAYES	0.75	0.78	0.84	0.75	0.77	0.73	0.76	0.77	0.77	0.76
KNN	0.84	0.83	0.86	0.81	0.86	0.86	0.86	0.87	0.87	0.86
MLR	0.74	0.78	0.83	0.81	0.80	0.80	0.80	0.82	0.82	0.82

Figure 8: Classification performance of Lung cancer Ontario database with different classifiers for second stage feature selection.

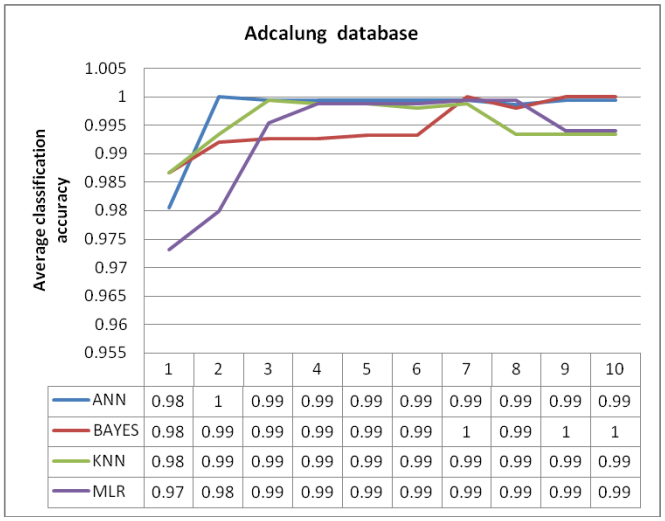
In Lung cancer Ontario database at Figure 8, KNN yields best result of 0.871 at 8 and 9, followed by Bayes with 0.848 at 3 and ANN with 0.838 at 2. However with higher number of features, order of performance is KNN, MLR, Bayes, ANN.



	1	2	3	4	5	6	7	8	9	10
ANN	0.86	0.49	0.49	0.94	0.94	0.95	0.96	0.96	0.96	0.82
BAYES	0.85	0.92	0.92	0.94	0.94	0.94	0.93	0.93	0.94	0.93
KNN	0.90	0.94	0.94	0.95	0.95	0.93	0.91	0.93	0.92	0.92
MLR	0.86	0.91	0.90	0.93	0.93	0.93	0.94	0.93	0.94	0.93

Figure 9: Classification performance of Prostate tumor database with different classifiers for second stage feature selection.

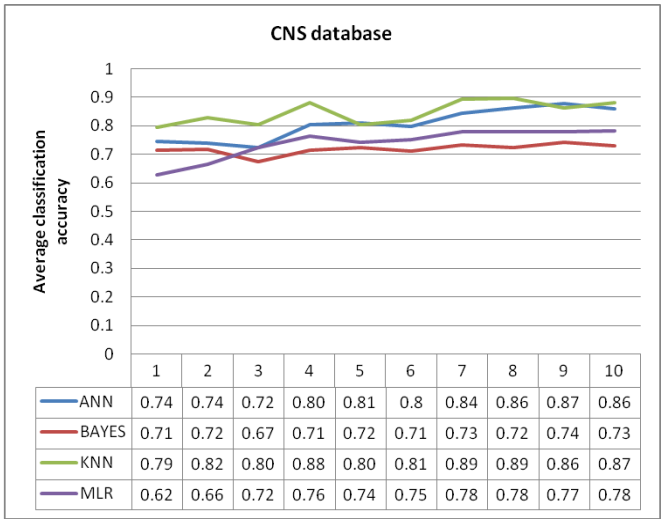
In Prostate tumor database at Figure 9, ANN yields best result of 0.966 at 8 and 9, followed by KNN with 0.951 at 5, and Bayes with 0.948 at 5. ANN fails to classify at 2 and 3, otherwise performance of all the classifiers remain competitive.



	1	2	3	4	5	6	7	8	9	10
ANN	0.98	1	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
BAYES	0.98	0.99	0.99	0.99	0.99	0.99	1	0.99	1	1
KNN	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
MLR	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99

Figure 10: Classification performance of Adaclung database with different classifiers for second stage feature selection.

In Adaclung database, the performance of all the classifiers are competitive. Figure 10 shows the performance of the classifiers for Adaclung database.



	1	2	3	4	5	6	7	8	9	10
ANN	0.74	0.74	0.72	0.80	0.81	0.8	0.84	0.86	0.87	0.86
BAYES	0.71	0.72	0.67	0.71	0.72	0.71	0.73	0.72	0.74	0.73
KNN	0.79	0.82	0.80	0.88	0.80	0.81	0.89	0.89	0.86	0.87
MLR	0.62	0.66	0.72	0.76	0.74	0.75	0.78	0.78	0.77	0.78

Figure 11: Classification performance of central nervous system database with different classifiers for second stage feature selection.

Figure 11 shows the performance of CNS databse for the four classifiers. KNN yeilds best results of 0.895 at 8, followed by ANN with 0.878 at 9. Best performance of other two classifier remain much below this.

Further to analyse the performance of different groups of features, the following observations are made. Higest perfromance of each classifier for each database is obtained, say H. Performance 5% less than Higest is treated as H'. The group of features perform between [H',H] are considered as the high performing feature groups. Similarly lowest performnace is taken as L and 5% more than the lowest performance is treated as L'. The feature groups perform between [L, L'] are treated as low performing feature groups. Based on these considerations the stastistics obtained are presented in Table 2 and Table 3 for high performing and low performing feature groups for each classifier.

Table2: High performing feature groups for classifiers

Databases	*High performing feature groups for*			
	ANN	*BAYES*	*KNN*	*MLR*
AllAml	2- 4	3,4,7-10	2, 3	9, 10
Colon Tumor	8, 9	7-10	6	9, 10
DLBCL Harvard Outcome	5	8, 9	7	8- 10
DLBCL Harvard Tumor	6	8	3	10
DLBCL Stanford	7	1-10	7, 8	9, 10
Lung Cancer Michigan	1	1	2, 4-10	1-10
Lung Cancer Ontario	2	3	8, 9	3
Prostate Tumor	7-9	5, 6, 9	4, 5	4, 7-9
Adcalung	2-10	7-10	3-7	3-8
CNC	9	9	7, 8	7- 10

The integer values say n in Table 2 and 3 represent the number of features starting from 1 to n in the decreasing order of signal to noise ratio i.e. the feature group n.

Table3: Low performing feature groups for classifiers

Databases	*Low performing feature groups for*			
	ANN	*BAYES*	*KNN*	*MLR*
AllAml	6-10	1	1	1, 2
Colon Tumor	1	4	1	1-4
DLBCL Harvard Outcome	1, 8-10	2	2, 3	2
DLBCL Harvard Tumor	10	1	9, 10	1
DLBCL Stanford	3-6	-	2	2-3
Lung Cancer Michigan	3-10	4	1, 3	-
Lung Cancer Ontario	7-10	6	4	1
Prostate Tumor	2, 3	1	1	1
Adcalung	1	1	1	1
CNC	3	3	1	1

Frequency of the feature groups in each database is calculated for high performance and low performance. Table 4 and 5 represents the frequencies of the feature groups in high performance and low performance, respectively.

Table4: Frequency of feature groups in high performance of the four classifiers

	Frequency of high performing feature groups									
Feature Size / Databases	1	2	3	4	5	6	7	8	9	10
AllAml	0	2	3	2	0	0	1	1	2	2
Colon Tumor	0	0	0	0	0	1	1	2	3	2
DLBCL Harvard Outcome	0	0	0	0	1	0	1	2	2	1
DLBCL Harvard Tumor	0	0	1	0	0	1	0	1	0	1
DLBCL Stanford	1	1	1	1	1	1	3	2	2	2
Lung Cancer Michigan	3	2	1	2	2	2	2	2	2	2
Lung Cancer Ontario	0	1	2	0	0	0	0	1	1	0
Prostate Tumor	0	0	0	2	2	1	2	2	3	0
Adcalung	0	1	3	3	3	3	4	3	2	2
CNC	0	0	0	0	0	0	2	2	3	1
Total	4	7	11	10	9	9	16	18	20	13

Table5: Frequency of feature groups in low performance of the four classifiers

	Frequency of low performing feature groups									
Feature Size / Databases	1	2	3	4	5	6	7	8	9	10
AllAml	3	1	0	0	0	1	1	1	1	1
Colon Tumor	3	1	1	2	0	0	0	0	0	0
DLBCL Harvard Outcome	1	3	1	0	0	0	0	1	1	1
DLBCL Harvard Tumor	2	0	0	0	0	0	0	0	1	2
DLBCL Stanford	0	2	2	1	1	1	0	0	0	0
Lung Cancer Michigan	1	0	2	2	1	1	1	1	1	1
Lung Cancer Ontario	1	0	0	1	0	1	1	1	1	1
Prostate Tumor	3	1	1	0	0	0	0	0	0	0
Adcalung	4	0	0	0	0	0	0	0	0	0
CNC	2	0	2	0	0	0	0	0	0	0
Total	20	8	9	6	2	4	3	4	5	6

Depending on the classifier model and database, different feature group perform differently and a single group cannot be treated as the best and any group cannot be treated as the worst. On analysis, it can be revealed that feature groups 7, 8 and 9 have got the high frequencies in Table 4 and low frequency in Table 5. Therefore feature group from 7-9 can be considered for simulation with minimal risk. In contrast, feature groups 1-3, has for high frequency in Table 5 and low frequency in Table 4, hence may not be considered suitable for simulation.

6. Conclusion

This paper has experimented on the feature selection strategies and has tried to find out the search space in each problem that favors classifier designing and also finds out the search space for which the designer should remain careful. For this experimentation, a two stage reduction approach is considered. In the first stage, the less relevant genes are dropped. In the second stage, attributes are ordered in high signal to noise ration basis. These ordered attributes are selected in different groups starting from one and experimented. Four different classifiers such as artificial neural network, naïve Bayesian, k-nearest neighbor, and multiple linear regression classifiers are employed for this experimentation. Objective of this experimentation is to find out suitable group of features which favors classifier design.

From the analysis of the results it is observed that a group of features from 7-9 favors high performing classifier design and from 1-3 does not favors for better performance. These observations are based on ten different microarray databases considered for this experimentation.

References

[1] J. Quackenbush, "Computational analysis of microarray data," Nat Rev Genet, 2(6), pp. 418-427, 2001.

[2] X. Zhou, D.P. Tuck, "MSVM-RFE extensions of SVM-REF for multiclass gene selection on DNA microarray data," Bioinformatics, 23 (9), 1106–1114, 2007.

[3] D. M. Mutch, A. Berger, R. Mansourian, A. Rytz, and M. A. Roberts, "Microarray data analysis: a practical approach for selecting differentially expressed genes," Genome Biol., 2(12), 2001.

[4] C -P Lee, Y. Leu, "A novel hybrid feature selection method for microarray data analysis," Applied Soft Computing, 11, pp. 208–213, 2011.

[5] R. Dash, B. B. Misra, S. Dehuri, and S.-b. Cho, "Efficient Microarray Data Classification with Three Stage Dimensionality Reduction," International Conference on Intelligent Computing, Communication & Devices (ICCD-2014), 18th and 19th April 2014, ITER, Bhubaneswar, 2014.

[6] M. B. Eisen, P. T. Spellman, P. O. Brown, D. Botstein, "Cluster analysis and display of genome-wide expression patterns," Proc. Natl. Acad. Sci., USA, 95(25), pp. 14863–14868, 1998.

[7] T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler, "Support vector machine classification and validation of cancer tissue samples using microarray expression data," Bioinformatics, 16(10), pp. 906-914, 2000.

[8] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, and M. B. Eisen, "Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization," Mol Biol Cell, 9(12), pp. 3273–97, 1998.

[9] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, and D. Koller, "Module networks identifying regulatory modules and their condition-specific regulators from gene expression data," Nat Genet, 34(2), pp. 166–76, 2003.

[10] Z. Bar-Joseph, G. K. Gerber, T. I. Lee, N. J. Rinaldi, J. Y. Yoo, and F. Robert, Computational discovery of gene modules and regulatory networks," Nat Biotechnol, 21(11), pp. 1337–42, 2003.

[11] S. Raychaudhuri, J. M. Stuart, R. B. Altman, "Principal components analysis to summarize microarray experiments: application to sporulation time series," Pac Symp. Biocomput., pp.455–66, 2000.

[12] O. Alter, P. O. Brown, D. Botstein, "Singular value decomposition for genome-wide expression data processing and modeling," Proc. Natl. Acad. Sci., USA. 97(18), pp. 10101–10106, 2000.

[13] S. Lee, S. Batzoglou, "Application of independent component analysis to microarrays," Genome Biol., 4(11), R76, 2003.

[14] D. V. Nguyen, and D. M. Rocke, "Tumor classification by partial least squares using microarray gene expression data," Bioinformatics, 18(1), pp. 39-50, 2002.

[15] K. J. Kim, S. B. Cho, "An evolutionary algorithm approach to optimal ensemble classifiers for DNA microarray data analysis," IEEE Transactions on Evolutionary Computation, 12, pp. 377–388, 2008.

[16] S. B. Patil, and Y. S. Kumaraswamy, "Intelligent and effective heart attack prediction system using data mining and artificial neural network," European Journal of Scientific Research, 31, pp. 642–656, 2009.

[17] D. Resul, T. Ibrahim, and S. Abdulkadir, "Effective diagnosis of heart disease through neural networks ensembles," Expert Systems with Applications, 36, pp. 7675–7680, 2009.

[18] D. J. Lockhart, E. A. Winzeler, "Genomics, gene expression and DNA arrays," Nature, 405(6788), pp. 827-836, 2000.

[19] L. Li, C. R. Weinberg, T. A. Darden, and L. G. Pedersen, "Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method," Bioinformatics, 17(12), pp. 1131-1142, 2001.

[20] A. J. Minn, G. P. Gupta, P. M. Siegel, P. D. Bos, W. Shu, and D. D. Girl, "Genes that mediate breast cancer metastasis to lung," Nature, 436, pp. 518–524, 2005.

[21] L. Fan, K.-L. Poh, and P. Zhou, "A sequential feature extraction approach for naïve bayes classification of microarray data," Expert Systems with Applications, 36, pp. 9919–9923, 2009.

[22] Y. Jun, Z. Benyu, L. Ning, Y. Shuicheng, C. Qiansheng, F. Weiguo, Y. Qiang, X. Wensi, and C. Zheng, "Effective and efficient dimensionality reduction for large-scale and streaming data preprocessing," IEEE transactions on Knowledge and Data Engineering,18(3), pp. 320-333, 2006.

[23] T. C. Hsia, System identification: least squares methods, D. C. Heath and Company, 1997.

[24] H. Lu, R. Setiono, H. Liu, "Effect data mining using neural networks," IEEE Trans. Knowl. Data Eng., 8, pp. 957– 961, 1996.

[25] P. Domingos, M. Plazzani, "On the optimality of the simple Bayesian classifier under zero one loss," Machine Learning 29, pp. 103–130, 1997.

[26] B. Cestnik, I. Bratko, "On Estimating Probabilities in Tree Pruning, Machine Learning – EWSL-91," European Working Session on Learning, Springer-Verlag, Berlin, Germany, pp. 138–150, 1991.

[27] T.M. Mitchell, Machine Learning, McGraw-Hill, 1997.

[28] G. J. Gordon, et. al."Translation of Microarray Data into Clinically Relevant Cancer Diagnostic Tests Using Gege Expression Ratios in Lung Cancer And Mesothelioma," Cancer Research, 62, pp. 4963-4967, 2002.

[29] D. G. Beer, et al., "Gene-expression Profiles Predict Survival of Patients with Lung Adenocarcinoma," Nature Medicine, 8(8), pp. 816-823, August 2002

[30] S. L. Pomeroy, et al., "Prediction of Central Nervous System Embryonal Tumour Outcome Based on Gene Expression," Nature, 415, pp. 436-442, January 2002.

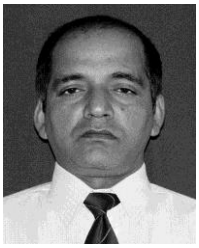
[31] D. Singh, et al., "Gene Expression Correlates of Clinical Prostate Cancer Behavior," Cancer Cell, 1, pp. 203-209, March, 2002.

[32] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. Caligiuri, C. D. Bloomfield and E. S. Lander, "Molecular classification of cancer: class discovery and class prediction by gene expression

monitoring," Science, New York, 286 (5439), pp. 531-7. ISSN 0036-8075, 1999.
[33] U. Alon, N. Barkai, D. A. Notterman, and K. Gish, S. Ybarra, D. Mack, and A. J. Levine, "Broad patterns of gene expression revealed by clustering analysis of tumor and normalcolon tissues probed by oligonucleotide arrays," Proc. Natl. Acad. Sci. USA, 96(12), pp. 6745–6750, 1999.
[34] A. D. Wigle, et al., "Molecular Profiling of Non-Small Cell Lung Cancer and Correlation with Disease-free Survival," Cancer Research, 62 pp.3005-3008, June 2002.
[35] A. A. Alizadeh, et al., "Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling," Nature, 403, pp. 503-511, 2000.
[36] Kent Ridge Bio-medical Dataset, retrieved Aug 25, 2013, from http://datam.i2r.a-star.edu.sg/datasets/krbd/
[37] T. Cover, P. Hart, "Nearest neighbor pattern classification," Proc. IEEE Trans. Inf. Theory, pp. 21–27, 1967.

Author Profile

Mrs. Rasmita Dash received her B.Tech. degree from Utkal University, Bhubaneswar, India in 2002, M.Tech degree in 2007 from the Biju Patnaik University of Technology, Rourkela, India and currently pursuing her Ph. D. in Computer Science and Engineering under Siksha 'O' Anusandhan University, Bhubaneswar, India. She is currently working as Asst. Professor in the Department of Computer Science and Information Technology at Institute of Technical Education and Research (I.T.E.R.), Faculty of Engineering under Siksha 'O' Anusandhan University, Bhubaneswar, India. She is having teaching experience of more than 10 years. Her major research Interests includes data mining, soft computing and bioinformatics.



B. B. Misra has completed his B. Text. degree in 1984 from Kanpur University, M. Tech. (Computer Science) in 2002 from Utkal University, and Ph.D. (Engineering) in 2011 from Biju Pattanaik University of Technology. He has done his Post Doctoral Research during 2013-14, at AJOU University, South Korea under the Technology Research Program for Brain Science of Yonsei University. His areas of interests include data mining, soft computing, wireless sensor network, bioinformatics. He has published two books, three book chapters, and more than 65 papers in different journals and conferences of National and International repute. Presently he is continuing as Dean Research at Silicon Institute of Technology, Bhubaneswar.