Paper: ASAT-14-173-CE

14th International Conference on
AEROSPACE SCIENCES & AVIATION TECHNOLOGY,
ASAT - 14 – May 24 - 26, 2011, Email: asat@mtc.edu.eg
Military Technical College, Kobry Elkobbah, Cairo, Egypt
Tel: +(202) 24025292 –24036138, Fax: +(202) 22621908

Gene Ranking Techniques via Attribute Evaluation Algorithms for DNA Microarray Analysis

M.B. Al Senousy*, H.M. El-Deeb†, Kh. Badran‡ and I.A. Al-Khlil§

Abstract: Due to the huge numbers of genes that produced from microarray technology versus genes that actually discriminate disease classes, gene selection methods for microarray data analysis are vital to identify the significant genes that distinguish disease classes and to use these selected genes as diagnostic biomarkers in clinical treatment decisions. In this study, we describe how to achieve reduction of microarray data dimensionality by two attribute selection methods (AS), namely information gain method (IG) and support vector machine method (SVM) which can greatly reduce the number of attributes used to discriminate microarray data. We employ both methods, to pre-process gene expression profiles achieved from DNA microarray experiments in three steps: (i) Ranking genes according to the highest dataset separation between diseased and normal classes, (ii) Choosing the smallest subset of ranked genes that assures the highest classification accuracy, (iii) Constructing the classification models to classify diseased versus normal samples using multiple algorithms based on the extracted subset in (ii). Evaluation of this approach was conducted by using ten different classification algorithms, with eight variant cancerous microarray dataset. Based on the obtained results, this pre-processing approach improved classification accuracy compared to using the whole original dataset. All the evaluated algorithms which used in our approach provided classification accuracy exceeds over (94%) with majority of datasets. By using a few numbers of top ranking genes, we obtained higher classification accuracy instead of using original dataset, the average values of enhancement were (1.31%, 3.01%, 4.06%, 3.54% and 3.59%) using (2, 5, 10, 20, 50) subset of ranking genes by information gain attribute selection respectively, and (0.19%, 4.33%, 5.05%, 5.54% and 5.63%) using (2, 5, 10, 20, 50) subset of ranking genes by SVM attribute selection. Experimental results shows that using SVM attributes selections method yields better results than using information gain attribute selection method as preprocessing stage of the classification task. Also, it can be shown that Artificial Neural Network (ANN) outperforms all classifiers when SVM attribute selection method used while Bayes Net outperforms all classifiers when information gain attribute selection is applied.

Keywords: Microarray gene expression data, supervised classifier, attributes selection, ranking genes, cancer.

* Prof., Sadat Academy for Management Science (SAMS), Department of Computer science, badr_senousy_arcoit@yahoo.com
† Dr., Modern University for Technology and Information (M.T.I), Department of Computer science, hmeldeeb14@yahoo.com
‡ Egyptian Armed Forces, khaledBadran@hotmail.com
§ Egyptian Armed Forces, ibrahim.alkhlil@gmail.com

1. Introduction

The microarray dataset that obtained from Microarray technology is fairly different from normal machine learning datasets. For instance normal machine learning datasets contain a small number of attributes and a large number of samples (tuples). In the other hand, gene expression microarray data typically contains a very large number of genes (attribute) vs. a small number of samples. With these large numbers of genes, it is preferable to have a large number of samples to build reliable microarray classification and prediction models [1]. But in reality most microarray experiments has a limited amount of samples due to the huge cost of producing such microarray data, in addition to the availability and privacy which added other factors of that limitation. For instance in cancerous microarray data, the number of samples is often less than 100 vs. 33000 genes.

The high dimensionality causes several problems in microarray data analysis, such as noise caused by human error, malfunctions and missing values, as well as irrelevant genes can significantly reduce the quality of microarray data classification. Processing the large number of genes causes great computational complexity in building classification models; also several classification algorithms don't deal with these numbers of attributes, due to memory consuming. In shortly, high dimensionality makes many classification algorithms not applicable for analyzing raw gene expression microarray dataset. Gene selection is a fundamental step when building predictors of disease state based on gene expression data; it's intended to reduce the risk of an overfitting problem and enhances the efficiency of the classification process, and increases comprehensibly of the result [2]. Gene selection is achieved by selecting a subset of genes from the original dataset which are mostly high predictive of classes category [3]. A lot of gene selection methods have been applied to microarray dataset classification in past decades [3, 4, 5].

The main objective of this paper is to investigate the effect of two gene selection methods on the performance of classification methods, by using subset of smallest number of ranking genes.

The rest of this paper is organized as the follows. In section 2 we present a mention on public gene selection methods. In section 3 we provide problem statement and motivations. In section 4 we exploit the attribute selection methodologies and related algorithms. In section 5 we explore the used classification techniques. In section 6 we describe the evaluation Datasets. In section 7 we present the experimental results. In section 8 we conclude the paper.

2. Gene selection Algorithms

Gene/attribute selection problem consists of making good predictions/classification with as few genes as possible [6]. Some genes among the selected genes may have similar expression levels among classes, and they are redundant since no additional information is gained for classification algorithms by keeping them all in the dataset. Based on the dependency on classification algorithms, genes/attribute selection methods can be approximately separated into filter and wrapper methods [7]. A filter method can performs as independently from a classification method. It is preprocesses a microarray dataset before building classification model. As shown in figure 1. Filter-base gene selection is categorized based on evaluation functions into four main types, first: the evaluation by distance functions for example Euclidean distance measure and Cosine (CO) ranking methods [8], second: the evaluation by information measure (entropy or information gain (IG), gain ratio etc.) [5, 9], third: the evaluation by dependency measure (correlation coefficient CC) [8], fourth: the evaluation by consistency (min-features bias). There are other filter-base attributes selection method such

as, Markov blanket-embedded genetic algorithm for gene selection [10], Chi-square, Relief-F symmetric uncertainty [3], Signal-to-Noise ratio (SNR) [8], t-statistics (TS) [4] One-gene-at-a-time filter methods, such as ranking [11], Wilcoxon rank sum test [12].
In the other hand, a wrapper method embeds a gene selection method within a classifier, as shown in figure 2, for instance of a wrapper method is SVMs [13, 14]. SVMs use the recursive attribute elimination (RFE) approach to eliminate the attributes iteratively in a greedy approach until the largest margin of partition is reached. Shortly In spite of existing 32 different attribute selection methods yet, no single gene selection method can generally improve the performance of classification algorithms in terms of efficiency and accuracy thus there are about 60 different gene selection procedures developed by combining the attribute selection methods [12].

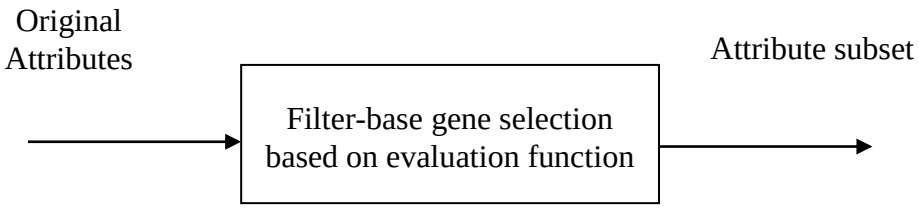

Fig. 1 Filter method for subset selection

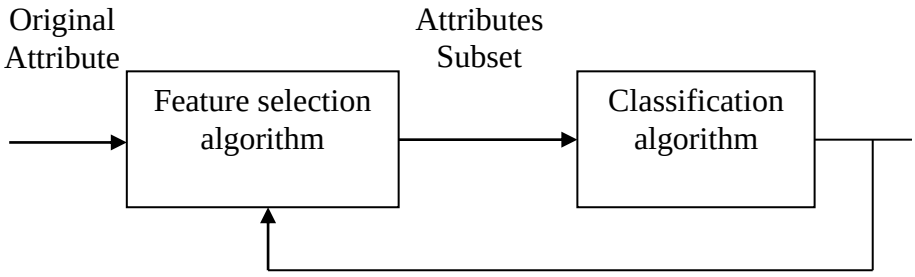

Fig. 2 Wrapper method for attribute subset selection

3. Problem Statement and Motivations

In data mining and machine learning, there are several possible statements of the problem of attribute selection. In this work, we place ourselves in the context of *supervised learning*, in which data samples or "patterns" are recorded as vectors **x** of dimension *N* and a target variable *y* should be predicted.

Let $\{x_i, y_i\}$; $i = 1, \ldots M$ pairs of training examples or training "patterns" are given in the form of a data matrix $X = [x_{ij}]$: $i=1, \ldots, M$, $j=1, \ldots, N$ of dimensions (M, N), with column vectors X_j, and a target matrix $Y = [y_i]$ of dimension $(M, 1)$. The main goal is to make good predictions of the target variable y for new patterns x with as few attributes as possible.
The attribute selection (AS) problem is selecting a set of attribute indices S of dimension N_s, corresponding to given columns of the data matrix, thus implicitly assuming that the same attributes are useful for making predictions *for any model*, in other words AS is a multi-objective problem, which has been formalized in several ways, including minimizing a risk functional subject to using a number of selected attributes N_s lower than a given threshold; or minimizing N_s subject to keeping the risk lower than a given threshold [13].

Gene classification as domain of research poses a new challenges due to its unique problem nature. The challenge comes from the unique nature of the available gene expression dataset; where most of these datasets has sample size below 100 vs. thousands to hundred thousands of genes presented in each tuple. Other challenge is only a few numbers of these (genes) presents relevant attributes to the investigated disease.

Due to the specification of microarray dataset some classification algorithms can deal with whole dataset, by using embedded gene selection, such as decision tree suite, which use split criterion as gene selection and SVM uses recursive feature eliminations (RFE) technique as dimension reduction. Other classification algorithm can deal with original microarray dataset without any gene selection such as KNN. But other classifier such as ANN, Naïve Bayesian and Bayesian Network are difficulty deal with such dataset, due to extremely consuming memory. Gene selection solves these challenges as we shown in the rest of this paper.

4. Methodologies and Related Algorithms

4.1 Attribute Selection Algorithms

From the range of attribute selection methods available, two established methods, including one from each of the main categories of methods: filter and wrapper witch used in our study. The salient issues of each method are briefly outlined next.

1) *Information Gain Attribute Selection* IGAS[5, 9]: is a filter-based method that achieved by calculating the expected information (entropy) for each attributes to sort these attributes in dataset *D*, according to higher information expected from splitting data by the attribute (genes) *A*.

- Calculate *Information Entropy* of the dataset *D as follow*.

$$Info(D) = \sum_{i=1}^{m} p_i log_2(p_i) \qquad (1)$$

Where, P_i is the probability that an arbitrary tuple in *D* belong to class C_i.

- Calculate the expected information required to classify a tuple from *D* based partitioning by attribute *A*.

$$Info_A(D) = \sum_{i=1}^{m} \frac{|D_j|}{|D|} \times Info(D_j) \qquad (2)$$

The term $\frac{|D_j|}{|D|}$ acts as the weight of the *j*th partition

- Calculate IG of attribute *A*.

$$Gain(A) = Info(D) - Info_A(D) \qquad (3)$$

- Arrange the value of information gain IG with descending order.

where attributes which own high IG values according to a specified threshold are considered as best dataset discrimination features. In our implementation, we apply attribute selection by selection of top *n* attributes where *n* is equal to 5, 10, 15, etc.

2) *SVM Recursive Feature Elimination* SVM-RFE: The SVM-RFE is a wrapper-based approach utilizing the SVM as base classifier [13, 14].

The backward elimination algorithms work by iteratively removing one "worst" gene at a time until the predefined size of the final gene subset is reached. In each loop, the remaining genes are ranked again, resulting in possibly modified genes ranking. Notice that correlation-based metrics cannot utilize back elimination because the ranking is never modified.

Recently, a backward elimination algorithm called the SVM-RFE achieved notable performance improvement.

In the SVM-RFE, the gene being removed should change the objective function *J* least.

$$J = \|w\|^2/2$$

In which *w* is calculated as $w = \sum_{i=1}^{N_s} \alpha_i y_i x_i$, margin width $= 2 / \| w \|$
where N_s is the number of support vectors, which are defined as the training samples with $0 < \alpha_i \leq C$ because a linear kernel is adopted.

The Optimal Brain Damage (OBD) algorithm approximates the change of *J* by removing the *ith* gene by expanding *J* in Taylor series to second order:

$$\Delta J(i) = \frac{\partial J}{\partial w_i} \Delta w_i + \frac{\partial^2 J}{\partial w_i^2} (\Delta w_i)^2$$

At the optimum of *J*, the first order is neglected and the second order becomes

$$\Delta J(i) = (\Delta w_i)^2$$

Because removing the *i*th gene means $\Delta w_i = w_i$ we can adopt w_i^2 as the ranking criterion. The gene with the smallest w_i^2 is removed due to its smallest effect on classification.

In practice, more than one gene could be removed at a single step.
Algorithm below describes the SVM-RFE algorithm in detail.

The parameter *f*, here named "filter-out" factor, decides how many genes are removed at one step. Notice if $0 < f < 1$, a fraction of f bottom-ranked genes are removed at each step; if $f = 0$, the least possible bottom-ranked genes are removed so that the number of remaining genes is the power of 2 at the first step and then half of genes are removed in the following steps; If $f = -1$, only one gene is removed; if $f = -2$, two genes are removed, and so on. In each step, a new linear SVM is trained in a smaller feature space and, thus, each remaining gene is assigned a new weight w_i^2 to be ranked again. This process is repeated until the predefined number of features remains. Obviously, the SVM-RFE with $f = -1$ is the most time-consuming because the maximum steps are needed in this case.

Suppose there are ***s*** tissue samples and ***d*** genes, an SVM can be modeled in $O(s^2 * d)$ time.

```
SVM-RFE Algorithm
SVM-RFE(T,F,f,s)
Initialize
    T:= {training dataset}
    F:= {all input features}
    f:= {filter_out_factor}
    s:= {the size of final informative gene subset}
 begin
     while (the size of F>s)
         Train linear SVM on T in feature space defined by F Rank the features of F  Rank the features of F by w_i^2 in the
             descending order
             If f<0
                 F2:=F-{ f bottom ranked featuers in F}
             elseif  F2:=F-{a number of features with largest ranks are removed so that
                             the size of F2 is closest smaller number of power of 2}
             eles
                  F2:=F-{ f *100% of features in F with largest rank}
             end
             F=F2
         end
         return F
 end
```

4.2 Classification Algorithms

C4.5: algorithm top-down decision tree base proposed by Quinlan [15]. The algorithm uses the greedy technique and is a successor of ID3 algorithm, which determines at each step the most predictive attribute, and splits a node based on this attribute. Every node represents a decision point over the value of some attribute. The split criterions based on information entropy see (1). C4.5 uses gain ratio for spilt dataset, see (2) (3),
The split information of attribute *A* Calculated as:

$$SplitInfo_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right) \qquad (4)$$

$$\text{Gain ratio: } GainRatio(A) = \frac{\text{Gain(A)}}{\text{SplitInfo}_A(D)} \qquad (5)$$

The attribute with the maximum gain ratio is selected as best splitting attribute.

Bagging: produced by Leo Breiman [16, 17] it uses a bootstrap technique to resample the training data sets *D*. To form a resampled data set D_i, each sample is independently drawn from *D* with *n* samples. Each sample in *D* has a probability of 1/n of being drawn in any trial. D_i contains the same number of samples as the original data set *D*. in D_i, some samples may appear more than once, and some samples do not appear at all training data. *The tree* T_i is built on a set of resampled D_i. This process returned for *K* times. The final prediction of a sample is determined by simple voting and each classifier has an equal weight of 1. The most often predicted class label will be the final classification result.

AdaBoost: developed by Freund and Schapire [18]. In this method initial classifier is constructed from the original data set where every sample has an equal distribution ratio of 1. In the next iterations D_i constructed according on prediction accuracy in the previous data set D_{i-1}. If a sample has a lower prediction accuracy rate in D_{i-1}, it will be given a higher weight in D_i and therefore get a higher possibility to be selected in D_i.

Naïve Bayesian: The algorithm is depending on Bayes theorem, its work as the follows:
Let D be a training set and their associated class labels, each tuple is represented by *n*-dimensional attribute vector, $\boldsymbol{X} = (x_1.x_2, ...,x_n)$, naïve Bayesian Classifier, predict that tuple $\boldsymbol{X}$ belongs to the class $\boldsymbol{C}_i$ if and only if
$P(C_i|X) > P(C_j|X)$ for $1 \le j \le m, j \ne i$ this mean maximize $P(C_i|X)$. By Bayes theorem

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \qquad (6)$$

where $P(C_i) = |C_{i,D}|/|D|$, $|C_{i,D}|$ is the number of training tuble of class C_i in *D*.

$$P(C_i|X) = \prod_{k=1}^{n} P(x_k|C_i) \qquad (7)$$

where x_k refers to the value attribute A_k .for microarray data, A_k is continues-value, attribute is assumed to have Gaussian distribution

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \qquad (8)$$

$$P(x_k|C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i}) \qquad (9)$$

where σ is standard deviation, μ is mean [19].

Bayesian Network: The naïve Bayesian classifier makes the assumption of variables conditional independent, but in practice dependencies can exist between variables. Bayesian networks specify joint conditional probability. They provide a graphical model of causal relationship.
Let **X**=$\{x_1,...,x_n\}$, $n \geq 1$ be a set of variables (attributes). *Bayesian Network B* over set of variable **X** is a *network structure* B_S, which is a directed acyclic graph (DAG) over **X** and a set of probability tables- $B_P = \{p(x|pa(x))|x \in X\}$ where *pa(x)* is the set of parents of *x* in B_S. A Bayesian network represents probability distributions $P(X) = \prod_{x \in X} p(x|pa(x))$. To use Bayesian network as classifier, one simply calculate $argmax_y$ *P(y*|x*)* using distribution *P(X)* representing by Bayesian network.

$$\begin{aligned} P(y|\mathrm{x}) &= P(X)/P(\mathbf{x}) \quad \approx P(X) \\ ArgMax_y \; P(y|\mathrm{x}) &= \prod_{x \in X} p(x|pa(x)) \end{aligned} \qquad (10)$$

And since all variables in **x** are known, we do not need complicated inference algorithms, but just calculate (10) for all class values [20, 21].

Neural networks (Multilayer Perceptron): ANN is predictive models loosely based on the action of biological neurons, Contains from several layers of neurons. In general an input layer takes the input and distributes to the hidden layers which do all the necessary computations and outputs. The standard algorithm used for classification is a multilayered ANN trained using back propagation and the delta rule [22]. The algorithm based on initial state and calculates the output for each node, and computes prediction errors for each node, and update the weight for next iteration, thus return these step until the condition threshold is satisfied.

Support Vector Machine SVM: performs classification by constructing an n-dimensional hyper plane that optimally separates the data into two categories [23]. To extend SVMs for discriminating more than two classes, several approaches have been introduced [24].

K-nearest-neighbor: The K-NN classifier constructs decision boundaries by just storing of the complete training data. An object $\vec{x}$ is classified by choosing the majority class among the k closest objects of the training dataset. Standard or weighted metric distance functions are used to determine the distance of the *k* nearest neighbors from $\vec{x}$, such as: Absolute distance measuring: $d_A(X,Y) = \sum_{i=1}^{n} |x_{i-}y_i|$ or Euclidean distance measuring: $d_E = \sum_{i=1}^{N} \sqrt{x_i^2 - y_i^2}$. In fact, *k* is determined experimentally [25].

PART: is a separate-and-conquer rule learner proposed by Eibe and Witten [26]. The algorithm producing sets of rules called "decision lists" which are ordered set of rules. A new data is compared to each rule in the list in turn, and the item is assigned the category of the first matching rule. PART builds a partial C4.5 decision tree in each iteration and makes the "best" leaf into a rule. The algorithm is a combination of C4.5 and RIPPER (Repeated incremental pruning to produce error reduction algorithm) [27, 28].

Decision Table: It summarizes the dataset with a "decision table" which contains the same number of attributes as the original dataset. Then, a new data item is assigned a category by finding the line in the decision table that matches the non-class values of the data item. It employs the wrapper method to find a good subset of attributes for inclusion in the table. By eliminating attributes that contribute small or nothing to a model of the dataset, the algorithm

reduces the likely-hood of over-fitting and creates a smaller and condensed decision table [38]. We summarizes category of classification methods which used in this study in Table 1.

Table 1. Summary of the main characteristics of the classification methods, C4.5, Bagging (Ba), AdaBoost (Ad), SVM, ANN,KNN, Naive Bayes (NB) , Bayesian Network (BN),PART, Decision Table (DT)[29]

Category	Paradigm		Knowledge Rep. and Inference Method
Decision Tree	C4.5	Decision-tree induction	Decision-tree. *Inference*: class given by the corresponding leaf.
Meta Classifier	Ba.	Statistical Learning Theory	*Population of rules with strength per class.* *Inference:* The output is the most voted class (weighted by the strength) among the matching classifiers
	Ad.		
Function	SVM	SVM (Poly Kernel)	Weights of the support vector machines. *Inference*: The class is determined by the decision function represented by the SVM.
	ANN	Multilayer Perceptron	Weights of the network *Inference*: The class is determined by the decision function represented by the NN
Lazy	KNN	10NN	*Inference*: Majority class matching in the training dataset.
Bayes	NB	Statistical Modeling	Probabilities of a Bayesian model. *Inference*: the output is the class with maximum probability.
	BN		
Rule	PART	Rule induction based on decision-tree	Ordered list of rules. *Inference:* the output is the class of the first matching rule in the ordered list
	DT		Summarizes the dataset with a 'decision table'. *Inference*: ordered list of rule

5. Classification Based on Ranking Genes

Our approach consists from three main steps:

First, gene ranking: A ranking method identifies one gene at a time with differentially expressed levels among predefined classes and puts all genes in decreasing order; we have applied two gene selection methods (IGAS and SVMAS) to ranking the genes that provides highest data distinction in a decreased order.

Second, Form subsets of ranking the attributes: $R(X_{r1}, Y) \leq R(X_{r2}, Y) \leq \ldots \leq R(X_{rN}, Y)$; and form nested subset attributes: S1 = {r_1}; S2 = {r_1, r_2},... , S_N = {r_1; r_2, ...,r_N}. We select a five subset of top ranking genes (2, 5, 10, 20 genes, and 50 genes) as classifier entry.

Third, evaluate classifiers accuracy for each subset of ranking genes among each dataset.

The accuracy measured as follows:

$$accurcy = \frac{numbers\ of\ correct\ classified\ samples}{total\ number\ of\ samples} \quad (11)$$

We selected *k*-fold Cross-validation technique to evaluate our approach. This technique is commonly used to evaluate classifier for micro-array datasets especially with small number of samples. This technique is described as follow:

- Partition the data randomly into mutually exclusive *k* subse--ts, each subset with approximately equal size.
- At each iteration, use one subset as test set and others is merged as training set
- Each split is called "fold" where notation *k-fold* is used to identify number of splits and value of 10 is the most popular selected value for *k*
- Finally, Average *CV* (*h*,*k*) over all folds to obtain *CV* (*h*) "the classifiers accuracy for each genes subset".

The classifiers described in this paper were performed using libraries from Weka 3.7.1 machine learning environment [30].

6. Datasets

In this study, we select eight microarray datasets these datasets are ***Breast cancer dataset*** [31]: which contains from 62 samples, divided on two classes (tumor/normal), each sample contains from 16383 genes. ***Colon tumor dataset*** [32]: Contains from 36 samples, divided on two classes (tumor/normal), each samples contains from 7458 gens. ***Leukemia dataset*** [33]: Contains from 72 samples divided on two classes (acute lymphoblastic leukemia (ALL)/ acute myeloid leukemia (AML)), each samples contains from 7130 genes. ***Lung1 cancer dataset*** [34]: Contains from 197 samples divided on four classes (AD adenocarcinomas-/SQ squamous cell carcinomas COID carcinoids/NL: normal lung) each sample contains from 10937 genes. ***Lung2 cancer***: Contains from 88 sample divided on two classes (tumor/normal), each sample contains from 16382 genes. ***Lymphoma dataset*** [35]: Contains from 40 sample divided on two classes (PA patient/ CO control). ***Prostate cancer dataset*** [36]: Contains from 146 sample divided on two classes (normal prostate tissue adjacent to tumor/ tumor tissue) each sample contains from 12626 genes. ***Prostate 2***: Contains from 108 sample divided on two classes (tumor/ normal), each sample contains from 12554 genes. The breast cancer, lung2 cancer, prostate cancer prostate2 cancer and lymphoma, datasets were collected from EBI, http://-www.ebi.ac.uk/arrayexpress, lung1 cancer, colon tumor, Leukemia, were collected from Gene Expression Omnibus (GEO) repository [37]. Table 2 provides briefly description of microarray dataset.

Table 2 Description of microarray datasets

Dataset	Sample NO.	Gene No.	Category			
			tumor		normal	
Breast	62	16383	43		19	
Colon	36	7458	18		18	
Lung2	88	16382	69		19	
Prostate	146	12626	65		81	
Prostate2	108	12554	92		16	
Lung1	197	10937	AD	NL	SQ	COID
			139	17	21	20
Leukemia	72	7130	ALL		AML	
			46		26	
Lymphoma	40	16381	PA		CO	
			40		20	

7. Results and Analysis

In this section, we exposed the results of our approach and relevant analysis. The tables below summarized these results, where each table contains the classification accuracy of the classification algorithms, on each subset of top ranking genes (2, 5, 10, 20 and 50 genes), as well as the classification accuracy of the classifiers on original data (without gene selection), in addition to the average accuracy of all subsets of top ranking genes by IGAS and SVMAS, also the average accuracy obtained from all classifiers. We highlight the highest classification accuracy obtained using IGAS and SVMAS by bold font for each subset, and we indicated to the highest accuracy on all subset by gray cell color.

By dealing with ***Breast*** dataset: we notice from Table 3 the following: Using subset of top ranking gene improves the classification accuracy on majority of the classification algorithms the improvements fluctuate between (1.61% ~ 8.06%). We obtain the highest classification accuracy on majority of the classification methods by using 2, 5and 10 of top ranking genes, with the tow gene selection methods; the average accuracy of the classifiers by using was outperforming the average accuracy obtained using IGAS.

Table 3. Classification accuracy and average accuracy %, using (2, 5, 10, 20, and 50) subset ranking genes by IGAS and SVMAS, and accuracy on original Breast cancer dataset.

	2 Genes		5 Genes		10 Genes		20 Genes		50 Genes		Average		Original DS
Methods	IGAS	SVM AS	IGAS	SVMAS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IG AS	SVM AS	
C4.5	**96.77**	**96.77**	**96.77**	**96.77**	**96.77**	**96.77**	90.32	**95.16**	88.71	**93.55**	93.55	**95.81**	88.71
Ad.(C4.5)	**96.77**	**96.77**	**96.77**	**96.77**	**100**	96.77	**100**	98.39	**100**	**100**	93.23	**97.74**	96.77
Ba.(C4.5)	98.39	98.39	96.77	96.77	96.77	96.77	**100**	98.39	98.39	95.16	92.83	**97.10**	98.39
BN	**100**	98.389	**100**	**100**	98.39	**100**	**100**	**100**	**100**	**100**	92.14	**99.68**	-
NB	95.16	**100**	**100**	96.77	**98.39**	96.77	**98.38**	96.77	**98.39**	95.16	91.01	**97.10**	-
ANN	70.97	**100**	98.39	**100**	**100**	**100**	**100**	**100**	**100**	**100**	90.32	**100**	-
SVM	88.71	96.77	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	93.55	**99.35**	98.39
KNN	91.94	**100**	96.77	**100**	**100**	**100**	**100**	**100**	**100**	**100**	94.52	**100**	95.16
DT	**98.39**	**98.39**	96.77	**98.39**	95.16	**98.39**	90.32	**96.77**	90.32	**95.16**	95.16	**97.42**	90.32
PART	**98.39**	96.77	95.16	**96.77**	91.94	**96.77**	88.71	**95.16**	88.71	**93.55**	94.09	**95.81**	88.71
Average	93.55	**98.23**	97.74	**98.23**	97.74	**98.23**	96.77	**98.06**	96.45	**97.26**	93.04	**98.00**	94.01

By dealing with ***Colon*** dataset**:** we notice from Table 4 the following: Using subset of top ranking gene improvers the classification accuracy of the majority of the classification algorithms the improvement fluctuates between (2.78% ~ 5.55%). We obtain the highest classification accuracy on majority of the classification methods using 2 of top ranking genes; by IGAS and 5 of top ranking genes by SVMAS. The average accuracy of the classifiers using SVMAS outperforming average accuracy obtained using IGAS on majority of classifiers.

By dealing with ***Leukemia*** dataset: we notice from Table 5 the following: Using subset of top ranking gene improvers the classification accuracy of all classification algorithms the improvements fluctuates between (1.39~16.66%). We obtain the highest classification accuracy on majority of the classification methods using 20 of top ranking genes, by SVMAS vs. 50 of top ranking genes by IGAS. The accuracy of the classifiers using SVMAS outperforms the accuracy using IGAS on majority of the classifiers.

Table 4. Classification accuracy and average accuracy %, using (2, 5, 10, 20, and 50) top ranking genes by IGAS and SVMAS, and accuracy on original Colon cancer dataset.

	2 Genes		5 Genes		10 Genes		20 Genes		50 Genes		Average		Original DS
Methods	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IG AS	SVM AS	
C4.5	**97.22**	**97.22**	**97.22**	91.67	**91.67**	**86.11**	**91.67**	88.89	**91.67**	88.89	**93.89**	90.56	91.67
Ad.(C4.5)	**97.22**	**97.22**	97.22	**100**	**97.22**	**97.22**	97.22	**100**	97.22	**100**	97.22	**98.89**	97.22
Ba.(C4.5)	**97.22**	**97.22**	**97.22**	91.67	**94.44**	91.67	**94.44**	**94.44**	**94.44**	**94.44**	**95.56**	93.89	94.44
BN	**100**	97.22	**100**	97.22	**100**	**100**	**100**	**100**	**100**	**100**	**100**	98.89	-
NB	97.22	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	99.44	**100**	-
ANN	**100**	**100**	**100**	**100**	97.22	**100**	97.22	**100**	**100**	**100**	98.89	**100**	-
SVM	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	97.22
KNN	**100**	**100**	**100**	**100**	97.22	**100**	**100**	**100**	**100**	**100**	99.44	**100**	97.22
DT	**97.22**	**97.22**	**97.22**	91.67	94.44	86.11	91.67	86.11	86.11	86.11	**93.33**	89.44	86.11
PART	**97.22**	**97.22**	**97.22**	91.67	**97.22**	86.11	**91.67**	88.89	**91.67**	88.89	**97.28**	90.56	91.67
Average	98.33	**98.33**	98.61	**96.39**	96.94	**94.72**	96.39	**95.83**	96.11	**95.83**	**97.28**	**96.22%**	94.05

Table 5. Classification accuracy and average accuracy %, using (2, 5, 10, 20, and 50) top ranking genes by IGAS and SVMAS, and accuracy on original Leukemia dataset.

	2 Genes		5 Genes		10 Genes		20 Genes		50 Genes		Average		Original DS
Methods	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IG AS	SVM AS	
C4.5	87.5	**88.89**	87.5	**88.89**	88.89	**88.89**	**86.11**	**86.11**	**86.11**	83.33	84.13	**87.22**	79.17
Ad.(C4.5)	**90.28**	84.72	87.5	**93.06**	91.67	**93.06**	94.44	**95.83**	90.28	**97.22**	89.68	**92.78**	86.11
Ba.(C4.5)	**90.28**	81.94	90.28	**93.06**	90.28	**91.67**	90.29	**91.67**	**90.29**	**90.29**	**89.88**	89.72	88.89
BN	**90.28**	79.17	93.056	**97.22**	97.22	**98.61**	97.22	**98.61**	97.22	97.22	**95.00**	94.17	-
NB	**93.06**	81.94	93.06	**97.22**	93.06	**97.22**	94.44	**97.22**	94.44	**97.22**	93.61	**94.17**	-
ANN	**93.06**	90.28	91.67	**95.83**	94.44	**97.22**	94.44	**98.61**	94.44	**98.61**	93.61	**96.11**	-
SVM	**91.67**	80.56	91.67	**95.83**	93.06	**98.61**	93.06	**98.61**	94.44	**98.61**	94.05	**94.44**	97.22
KNN	**91.67**	81.94	**91.67**	90.28	93.06	**95.83**	91.67	**97.22**	95.83	**97.22**	89.29	**92.50**	80.56
DT	84.72	**77.78**	84.72	**91.67**	**93.06**	**93.06**	91.67	**93.06**	**90.28**	87.5	87.50	**88.61**	87.5
PART	**87.5**	77.78	**87.5**	**87.5**	**88.89**	**88.89**	**88.89**	**88.89**	**87.5**	84.72	84.52	**85.56**	79.17
Average	90.00	82.5	89.86	93.06	92.36	94.31	92.22	**94.58**	92.08	93.19	90.13	**91.53**	85.52

By dealing with ***Lung1*** dataset: As we mention in section 7, Lung1 dataset consist of four classes, we notice from Table 6 the following: Using subset of top ranking gene improvers the classification accuracy of the classification algorithms in several subset of ranking genes.

By dealing with ***Lung2*** dataset: we notice from Table 7 the following: Using subset of top ranking gene improves the classification accuracy of all classification algorithms the improvements fluctuate between (1.13%~5.68%). we obtain highest accuracy using 10, 20, 50 of top ranking genes by SVMAS; but we obtain the highest accuracy on majority of the classification methods by using 5 top ranking genes by IGAS. The average accuracy of the classifiers by using SVMAS was outperforming IGAS.

By dealing with ***Lymphoma*** dataset**:** From Table 8 we notice the following: Using subset of top ranking gene improves classification accuracy the improvements fluctuate between (1.1% ~ 5%), and we obtain 100% accuracy using most subsets of ranking genes by the two methods of AS.

Table 6. Classification accuracy and average accuracy %, using (2, 5, 10, 20, and 50) top ranking genes by IGAS and SVMAS, and accuracy on original Lung1 dataset.

	2 Genes		5 Genes		10 Genes		20 Genes		50 Genes		Average		Original DS
Methods	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IG AS	SVM AS	
C4.5	83.25	**88.89**	86.29	**88.89**	**92.39**	88.89	**92.39**	86.11	86.80	83.33	**89.05**	87.92	90.86
Ad.(C4.5)	80.20	**84.72**	87.31	**93.05**	**95.43**	93.06	93.40	**95.83**	89.34	**97.22**	**90.36**	89.54	93.40
Ba.(C4.5)	**82.23**	81.94	87.82	**93.06**	89.84	**91.67**	89.34	**91.67**	88.32	**90.28**	**89.34**	**89.34**	93.40
BN	**82.74**	79.17	88.83	**97.22**	93.40	**98.61**	92.39	**98.61**	94.92	**97.22**	90.46	**90.56**	-
NB	**83.25**	81.94	88.83	**97.22**	91.37	**97.22**	92.89	**97.22**	93.91	**97.22**	90.05	**91.68**	-
ANN	85.28	**90.28**	88.32	**95.83**	92.89	**97.22**	91.37	**98.61**	92.39	**98.61**	90.05	**93.10**	-
SVM	79.60	**80.56**	80.71	**95.83**	87.82	**98.61**	93.40	**98.61**	94.42	**98.61**	89.56	**93.20**	95.43
KNN	**85.79**	81.95	89.85	**90.28**	92.89	**95.83**	94.42	**97.22**	95.43	**97.22**	**92.89**	91.68	95.94
DT	**81.73**	77.78	82.23	**91.67**	85.79	**93.06**	**87.82**	93.06	85.79	**87.5**	84.99	**87.01**	85.29
PART	**83.25**	77.78	84.77	**87.5**	**92.39**	88.89	88.32	**88.89**	**87.82**	84.72	**87.89**	87.82	89.85
Average	82.74	81.79	86.50	93.52	91.42	94.91	91.57	**95.52**	90.91	94.29	89.46	90.18	92.02

Table 7. Classification accuracy and average accuracy %, using (2, 5, 10, 20, and 50) top ranking genes by IGAS and SVMAS, and accuracy on original Lung2 dataset.

	2 Genes		5 Genes		10 Genes		20 Genes		50 Genes		Average		Original DS
Methods	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IG AS	SVM AS	
C4.5	94.32	**95.45**	**94.32**	**94.32**	**94.32**	93.18	92.05	**93.18**	92.05	**94.32**	93.02	**94.09**	92.05
Ad.(C4.5)	94.32	**98.86**	**94.32**	**94.32**	95.45	**97.73**	95.45	**97.73**	95.45	**97.73**	95.78	**97.27**	97.73
Ba.(C4.5)	**98.86**	95.45	**95.45**	**95.45**	**95.45**	94.32	93.18	**95.45**	92.05	**95.45**	94.48	**95.23**	93.18
BN	**96.59**	92.05	**98.86**	**98.86**	**98.86**	**98.86**	**100**	98.86	**100**	97.73	**98.86**	97.27	-
NB	**97.73**	96.59	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**99.55**	99.32	-
ANN	**98.86**	**98.86**	**100**	**100**	98.86	**100**	98.86	**100**	98.86	**100**	99.09	**99.77**	-
SVM	97.73	**98.86**	**100**	**100**	98.86	**100**	**100**	**100**	**100**	**100**	99.19	**99.77**	98.86
KNN	**97.73**	96.59	**100**	98.86	98.86	**100**	98.86	**100**	98.86	**100**	97.56	**99.09**	94.32
DT	**98.86**	94.32	**98.86**	92.05	97.73	94.32	97.73	94.32	97.73	**97.73**	**97.73**	94.55	97.73
PART	**94.32**	**95.45**	**94.32**	**94.32**	**94.32**	**94.32**	92.05	**95.45**	92.05	**94.32**	93.02	**94.77**	92.05
Average	96.93	96.25	97.61	96.82	97.27	97.27	96.82	97.50	96.70	**97.73**	96.83	**97.11**	95.13

Table 8. Classification accuracy and average accuracy %, using (2, 5, 10, 20, and 50) top ranking genes by IGAS and SVMAS, and accuracy on original *Lymphoma* dataset.

	2 Genes		5 Genes		10 Genes		20 Genes		50 Genes		Average		Original DS
Methods	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IG AS	SVM AS	
C4.5	**98.33**	**98.33**	**98.33**	**98.33**	**98.33**	96.67	**98.33**	96.67	**98.33**	**98.33**	**98.10**	97.67	96.67
Ad.(C4.5)	**98.33**	**98.33**	**98.33**	**98.33**	**98.33**	96.67	**100**	96.67	**98.33**	**98.33**	**98.57**	97.67	96.86
Ba.(C4.5)	**98.33**	**98.33**	**100**	**98.33**	**98.33**	96.67	**98.33**	96.67	**98.33**	**98.33**	**98.81**	97.67	97.33
BN	98.33	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	99.67	**100**	-
NB	**100**	**100**	**100**	**100**	**100**	98.33	**100**	**100**	**100**	**100**	**100**	99.67	-
ANN	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	-
SVM	93.33	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	99.05	**100**	98.5
KNN	98.33	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	99.76	**100**	98.9
DT	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**96.66**	**100**	98.57	**100**	95
PART	**98.33**	**98.33**	**98.33**	**98.33**	**98.33**	**98.33**	**98.33**	**98.33**	**98.33**	**98.33**	98.10	**98.33**	96.67
Average	98.33	99.33	**99.50**	99.33	99.33	98.67	99.33	98.83	99.00	99.33	99.06	99.10	97.33

By dealing with ***Prostate*** dataset**:** as we mention in section 5 the normal tissues was adjacent to the tumor tissues, this special microarray dataset is not intended to evaluate the classifiers performance, that is justifies the low accuracy of the classifiers, with and without gene selections, instead of Using subset of top ranking gene significantly improve classification

accuracy the improvements fluctuate between (6.85% ~ 28.08%), and by using 50 top ranking genes ANN and SVM gave 100% accuracy as we show in Table 9.

By dealing with ***Prostate2*** dataset: we notice from Table 10 the following: Using subset of top ranking gene improves the classification accuracy on all the classifiers. The improvements fluctuate between (1.14% ~14.61%). We obtain the highest classification accuracy on majority of the classification methods using 20 top ranking genes by SVMAS. The average accuracy of the classifiers using SVMAS was outperforming using IGAS.

Table 9. Classification accuracy and average accuracy %, using (2, 5, 10, 20, and 50) top ranking genes by IGAS and SVMAS, and accuracy on original Prostate dataset.

	2 Genes		5 Genes		10 Genes		20 Genes		50 Genes		Average		Original DS
	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IG AS	SVM AS	
C4.5	67.12	**70.55**	71.23	**71.92**	69.86	**75.34**	71.23	**74.66**	69.86	65.75	67.71	**71.64**	61.64
Ad.(C4.5)	68.49	**70.55**	73.29	**78.77**	78.77	**80.14**	76.71	**78.78**	79.45	**80.82**	72.41	**77.81**	65.07
Ba.(C4.5)	67.81	68.49	71.98	**80.14**	73.97	76.71	**76.03**	79.45	**76.03**	**80.14**	70.65	**76.99**	63.70
BN	**71.92**	62.33	**75.34**	73.29	**79.45**	71.92	**80.82**	71.92	78.77	**80.14**	**77.26**	71.92	-
NB	**72.60**	70.55	69.86	**80.82**	73.29	**83.56**	76.71	**87.67**	77.40	**90.41**	73.97	**82.60**	-
ANN	**70.55**	74.66	73.29	**85.62**	71.92	**89.73**	67.81	**96.58**	75.34	**100**	71.78	**89.32**	-
SVM	73.97	**74.66**	73.97	**82.88**	74.66	**91.78**	75.34	**95.89**	75.34	**100**	73.87	**89.04**	71.92
KNN	**74.66**	73.97	73.97	**80.82**	72.60	**86.99**	72.60	**87.67**	74.66	**91.78**	72.11	**84.25**	67.81
DT	**77.40**	63.01	**77.38**	71.23	**78.08**	69.18	**77.40**	67.12	**72.60**	65.07	**71.72**	**67.12**	62.33
PART	**73.29**	69.18	73.29	**73.97**	70.55	**74.65**	**70.55**	**70.55**	**78.77**	71.92	69.96	**72.05**	61.64
Average	67.12	69.79	71.23	77.95	74.32	80.00	74.52	81.03	75.82	82.60	72.14	78.27	64.87

Table 10. Classification accuracy and average accuracy %, using (2, 5, 10, 20, and 50) top ranking genes by IGAS and SVMAS, and accuracy on original **Prostate2** dataset.

	2 Genes		5 Genes		10 Genes		20 Genes		50 Genes		Average		Original DS
	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IGAS	SVM AS	IG AS	SVM AS	
C4.5	92.59	90.74	91.67	**93.52**	**95.37**	92.59	87.96	87.96	88.89	86.11	89.68	**90.19**	85.19
Ad.(C4.5)	**91.67**	90.74	94.44	**95.37**	**95.37**	94.44	92.59	91.67	91.67	92.59	92.06	**92.96**	91.67
Ba.(C4.5)	**90.74**	89.81	**91.67**	**91.67**	**96.30**	92.59	**91.67**	89.81	**92.59**	91.67	**92.20**	91.11	91.67
BN	**92.59**	90.74	**97.22**	92.59	**96.30**	94.44	**98.15**	97.22	**98.19**	98.15	**96.48**	94.63	-
NB	89.81	**92.59**	**97.22**	96.30	94.44	**100**	95.37	**100**	97.22	**100**	94.81	**97.78**	-
ANN	**89.81**	**89.81**	**97.22**	96.30	93.52	**100**	92.59	**100**	97.22	**100**	94.07	**97.22**	-
SVM	85.19	**89.81**	85.19	**97.22**	89.81	**99.07**	92.59	**100**	96.30	**100**	90.87	**97.22**	93.59
KNN	88.89	**91.67**	93.52	**94.44**	**95.37**	93.59	94.44	**99.07**	92.59	**100**	91.27	**95.74**	87.04
DT	**90.74**	87.96	**93.52**	89.82	**93.52**	88.89	89.81	**91.66**	**87.96**	87.037	**89.15**	89.07	82.41
PART	**92.59**	88.89	91.67	**93.52**	**97.22**	90.74	87.96	**94.44**	89.81	**92.59**	89.42	**92.04**	82.41
Average	90.46	90.28	93.33%	94.07	94.72	94.63	92.31	**95.19**	93.24	94.81	92.00	93.80	87.70

In this discussion we do not pay attentions on the best classifier, where we focus only on classification accuracy improvement for each classifiers individually, but we briefly summarizing, the results by calculate the average classification accuracy of each classifiers on all dataset, as we show in Fig. 4 we notice C4.5 and rule base classifiers gave the lower average classification accuracy with IGAS and SVMAS, but the accuracy of C4.5 has been enhanced by ensample methods, (bagging and boosting) for instance the average improvement by bagging(C4.5) (2%) using SVMAS, (1.84%) using IGAS and the average improvement by boosting(C4.5) (3.69%) using SVMAS, (2.71%) using IGAS. Bayes Net classifier outperforms other classifiers with IGAS. ANN and SVM classifiers outperform other classifiers with SVMAS. Also we summaries accuracy as function of subset of ranking genes by calculate the average accuracy of all classifiers at all dataset as we show at Fig. 5,

from Fig. 5 we notice SVMAS better than IGAS as preprocessing stage of the classification task and each of the AS enhanced classifiers accuracy, the average values of enhancement was (1.31%, 3.01%, 4.06%, 3.54% and 3.59%) using (2, 5, 10, 20, 50) subset of ranking genes by IGAS respectively and (-0.19%, 4.33%, 5.05%, 5.54% and 5.63%) using (2, 5, 10, 20, 50) subset of ranking genes by SVMAS as we show in Fig. 6.

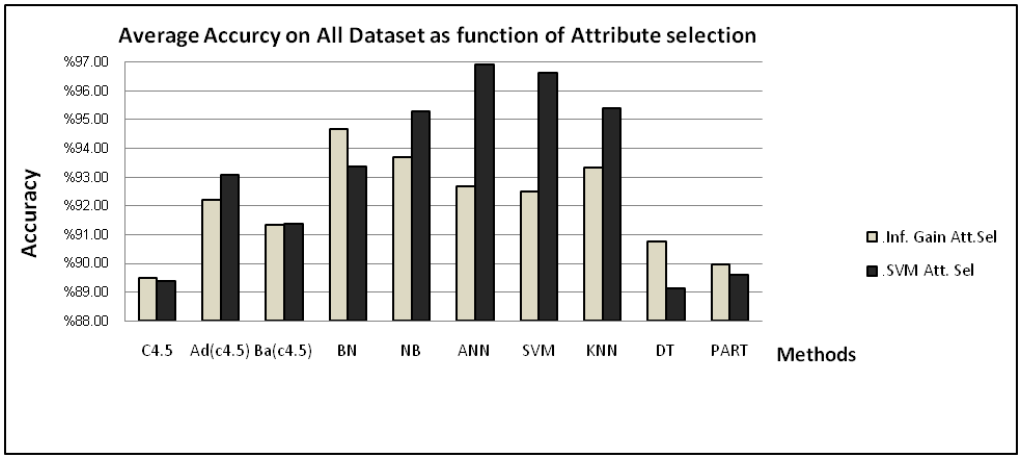

Fig. 4 Average accuracy of the classifiers as function of AS on all datasets

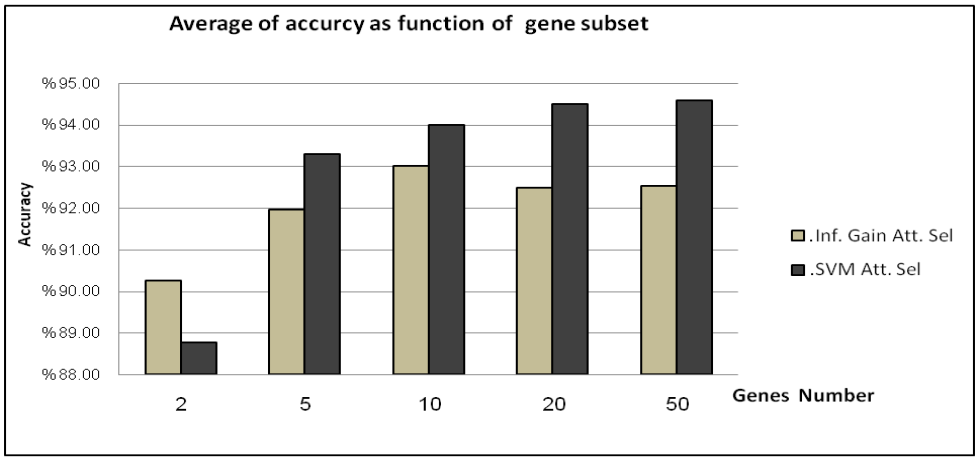

Fig. 5 Average accuracy of the classifiers as function of subset of ranking genes on all datasets

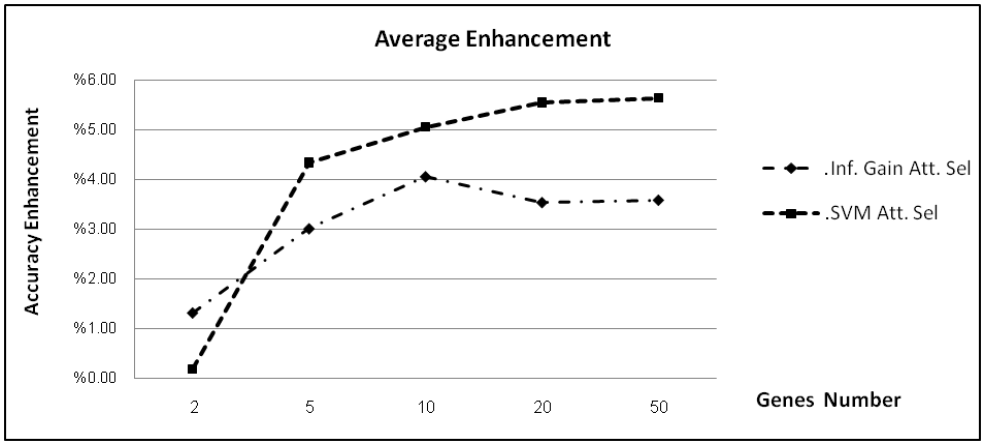

Fig. 6 Average enhancement of the classifier as function of subset of ranking genes on all datasets

We compare our approach with previous researcher such as Xiaosheng Wang et al (2010) [3], which use four method of attribute selection (chi-square, information gain, Relief-F and symmetric uncertainty), and Peter J. Tan et al (2007) which use Partial Least Square PLS as dimension reduction. From there results as we show in table 11 and our experiment we fiend our approach which select sub set of top ranking genes outperforms there results which apply classification via entire genes remaining attribute selection or dimension reduction.

Table 11. Previous works on cancer classification. where AdaBoos (Ad), Bagging(Ba)

Authors	Dataset	Att. Sel.	Classifier accuracy [%]			
Xiaosheng Wang et al. [3]	Colon		Naiv. Bay.	C4.5	SVM	k-nn
		Chi-Squ.	88.71	90.32	87.1	87.1
		Inf.	85.48	85.48	87.1	87.1
		RF	87.1	85.48	87.1	87.1
		SU	87.1	91.94	87.1	88.71
Peter J. Tan et al. [9]	Dataset		single C4.5	Ran. Forest	Ad C5.0	MML Oblique Forest
	Leukemia	PLS dimensionality	94.3	96.2	95.7	96.7
	Breast		65.2	71.2	67.9	69.2
	Central nervous		61.2	64.5	63.2	65.9
	Colon		80.9	84.7	82.7	88.8
	Lung		98	96.2	98.2	99.4
	Prostate		83	90.6	88.1	91.3
	Prostate		65	69.3	51.5	53.2

8. Conclusions

In the present study, we discuss inference of attribute selection (AS) methods upon the classification methods. By two diverse attribute selection methods, information gain (IG) as filter base AS and SVM (REF) as wrapper base AS method as preprocessing stage of classification task on microarray dataset. We evaluated classification accuracy of ten classifier belong to several classifier categories, where eight cancerous microarray dataset belong to several types of cancer were used.

From this wild experiment we conclude the following facts: AS methods significantly enhanced classification accuracy, and it can facilitate the classification task by using subset of few ranking genes instead of using original dataset. In additions it can facilitate the Comprehensibility of classifiers results due to eliminates irrelevant genes (attributes) by SVMAS or using highest relevancy genes which provides higher expected information from dataset by using IGAS. In addition to reduces computational time and memory consuming, by using few numbers of top ranking genes instead of using original dataset that contains several thousands of genes which produced from raw microarray dataset. For example it's extremely difficult to training Bayesian Networks, Naïve Bayesian and ANN Classifier on original microarray dataset.

Genes ranking by SVMAS provides better genes ranking by IGAS, due to the complexity of the optimization process, but a major disadvantage of the wrapper approach is that its computation requirement is formidable, for instance SVMAS spent more than two days vs.

several minutes spent by IGAS. Statistical base classifiers such as Bayesian Networks gave highest classification accuracy with ranking genes by IGAS but function base classifiers such as SVM and ANN gave the highest accuracy by ranking genes by SVMAS.

In spite of function base, lazy base and bayes base classifiers gave higher accuracy comparison with decision tree (C4.5) and rules base classifiers, but rules base are still attractive classifiers due to easy to results Comprehensibility and we can enhancing its performance by ensemble methods. Finally; Gene selection is a multi-faceted problem, which has evolved over the past few years from a collection of typically heuristic methods to a theoretically grounded methodology, finding and using a proper gene selection procedure specific to a given microarray dataset is necessary and useful.

9. References

[1] Manaswini Pradhan, Sabyasachi Pattnaik and Bhabatosh Mittra, "Effective Classification Technique by Blending of PPCA and EP-Enhanced Supervised Classifier: Classifies Microarray Gene Expression Data" American Journal of Scientific Research, ISSN 1450-223X Issue 11, pp.60-71, 2010.

[2] Y. Saeys, I. Inza, and P. Larra, "A review of feature selection techniques in bioinformatics" Bioinformatics, 23(19):2507–2517, 2007.

[3] Xiaosheng Wang and Osamu gotoh, "A Robust Gene selection Method for Microarray-based cancer Classifcation", Cancer Informatics, 15–30 2010:9

[4] Yukyee Leung and Yeungsam Hung, "A Multiple-Filter-Multiple-Wrapper Approach to Gene Selection and Microarray Data Classification", IEEE/ACM Transactions on computational biology and bioinformatics vol. 7, no. 1, 2010

[5] X. Liu, A. Krishnan, and A. Mondry, "An entropy-based gene selection method for cancer classification using microarray data", BMC Bioinformatics, 6:76, 2005.

[6] J. Bi, K. Bennett, M. Embrechts, C. Breneman, and M. Song, "Dimensionality reduction via sparse support vector machines", JMLR, 3:1229–1243, 2003.

[7] R. Kohavi and G. H. John, "Wrappers for feature subset selection", Artificial Intelligence, 1-2:273–324, 1997.

[8] S.-B. Cho and H.-H. Won, "Machine learning in DNA microarray analysis for cancer classification". In CRPITS '19: Proceedings of the First Asia Pacific bioinformatics conference on Bioinformatics, pp. 189–198, 2003.

[9] J. Tan Peter, L. Dowe David and I. Dix Trevor, "Building classification models from microarray data with tree-based classification algorithms", International Conference on Machine Learning, 2007.

[10] Z. Zhu, Y.-S. Ong, and M. Dash, "Markov blankete-mbedded genetic algorithm for gene selection. Pattern Recognition, 40(11):3236–3248, 2007.

[11] S. Mukkamala, Q. Liu, R. Veeraghattam, and A. H. Sung, "Feature selection and ranking of key genes for tumor classification: Using microarray gene expression data", ICAISC, volume 4029 of Lecture Notes in Computer Science, pages 951–961, Springer, 2006.

[12] Taeho Hwang, Choong-Hyun Sun, Taegyun Yun and Gwan-Su Yi, 1,2 " a filter-based gene selection workbench for microarray data", BMC Bioinformatics 2010.

[13] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines", Machine Learning, 46(1-3):389–422, 2002.

[14] Mahesh Pal and Giles M, "Feature Selection for Classification of Hyperspectral Data

by SVM ", IEEE Transactions on geosciences and remote segnaling vol. 48, no. 5, 2010.

[15] J. R. Quinlan, "C4.5: Programs for Machine Learning", Morgan Kaufmann, San Mateo California, 1993.

[16] L. Breiman, "Bagging predictors", Machine Learning, 24(2):123–140, 1996.

[17] M. B. Senousy, H. M. El-Deeb, K. Badran and I. A. Al-Khlil, " Suite of Decision Tree-Based Classification Algorithms on Cancer Gene Expression Data", Egyptian Informatics Journal, vol. 12, Issue 2, 2011

[18] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm", In International Conference on Machine Learning, pp. 148–156, 1996.

[19] Shyam Visweswaran, An-Kwok Ian Wong, M. Michael Barmada," A Bayesian Method for Identi-fying Genetic Interactions", AMIA 2009.

[20] Remco R. Bouckaert, "Bayesian Network Classiers in Weka" September 1, 2004

[21] N. Friedman, M. Nachman, and D. Pe'er., "Using baysian networks to analyze expression data", In Proc. of the 4th Ann. Int. Conf. on Comp. Molecule Biology, 2000.

[22] C. M. Bishop, "Neural networks for pattern recognition", Oxford University Press, 1995.

[23] B. Boser, I. Guyon, and V. Vapnik, " A training algorithm for optimal margin classifiers", In Proc.of 5th Annual ACM Workshop on Computational Learning Theory, pp. 144–152. ACM PRESS, 1992.

[24] N. H. Sweilam, A. A. Tharwat and N. K. Abdel Moniem, "Support vector machine for diagnosis cancer diseases: Acomparative study", Egyptian Informatics Journal, vol. 11, Issue 2, no 81-91, 2010.

[25] Li Baoli1, Yu Shiwen1, and Lu Qin2"An Improved k-Nearest Neighbor Algorithm for Text Categoriza-tion", International Conference on Computer Proce-ssing of Oriental Languages, Shenyang, China, 2003.

[26] Eibe Frank, Ian H. Witten "Generating Accurate Rule Sets Without Global Optimization", In: Fifteenth International Conference on Machine Learning, 144-151, 1998.

[27] Fadi Thabtah, Peter Cowling, "Mining the data from a hyperheuristic approach using associative classification", Expert Systems with Applications vol 34, pp 1093–1101, 2008.

[28] M. M MAZID, A B M SHAWKAT ALI, KEVIN S TICKLE, " Input space reduction for Rule Based Classification", Information Sconce and Applications, Issue 6, Vol. 7, 2010.

[29] Albert Orriols-Puig, Jorge Casillas, and Ester Bernad´o-Mansilla, "A Comparative Study of Several Genetic-Based Supervised Learning Systems", Journal of Machine Learning Research, 7:1–30, 2006

[30] Remco R. Bouckaert, Eibe Frank, Mark Hall, Richard Kirkby, Peter Reutemann, Alex Seewald, David Scuse, "WEKA Manual for Version 3-7-1", University of Waikato, Hamilton, New Zealand 2010.

[31] L. V. Veer, H. Dai, M. V. de Vijver, and et.al. Gene expression profiling predicts clinical outcome of breast cancer. Nature, 415:530–536, 2002.

[32] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, A. J. Levine, "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays", Proc Natl Acad Sci U S A. June 8; 96(12): 6575–6576 1999.

[33] T.R.Golub, D.K.Slonim, P. Tamayo, and et.al. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science,

286:531–537, 1999.

[34] Stefano Monti, Pablo Tamayo, Jill Mesirov and Todd Golub, "A resampling-based method for class discovery and visualization of gene expression microarray data", Kluwer Academic Publishers. Printed in the Netherlands 2003.

[35] A. Alizadeh, M. Eishen, E. Davis, and C. M. et. al. Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling. Nature, 403:503–511, 2000.

[36] D. S. et al. Gene expression correlates of clinical prostate cancer behavior. Cancer Cell, 1:203–209, 2002.

[37] I.F., Soboleva, A., Tomashevsky, M., Edgar, R.: NCBI GEO: mining tens of millions of expression profiles-database and tools update. Nucleic Acids Res., 35(Database issue), D760-5 (2007)

[38] Ron Kohavi, "The Power of Decision Tables", In: 8th European Conference on Machine Learning, 174-189, 1995