# Ensample Learning Based on Ranking Attribute Value (ELBRAV) for imbalanced Biomedical Data Classification

**M. B. Senousy**
Computer Science & Engineering Dept., the Sadat Academy for Management Science, Cairo, Egypt

**H. M. El-Deeb**
Computer Science & Engineering Dept., the Modern University for Technology and Information, Cairo, Egypt

**K. Badran**
The Military Technical College, Department of Computer Science, Cairo, Egypt

**Ibrahim Ali Al-Khlil**
The Military Technical Collage, Department of Computer Science, Cairo, Egypt

## Abstract

The imbalanced data problem occurs frequently and causes low prediction performance for discretization classes. Microarray technology provides an ideal example of imbalanced data problem. Microarray technology provides quantitative information about the complete transcription profile of cells and monitors the expression levels of thousands of genes at one time. This facilitates drug development, disease diagnosis, and understanding the basic cell biology capability. The imbalanced data problem occurs due to the fact that number of samples in each class not equal also the number of attributes is extremely greater than number of samples.

In this paper, a new ensemble method based on informative attributes namely Ensample Learning Based on Ranking Attribute Value (ELBRAV) was proposed, to resolve the imbalanced microarray data problem. The proposed method contains from three main steps. First: an active attributes evaluation algorithm to ranks the attributes according to its informative. By default is information gain ratio attributes evaluation was used. Second: building several classification models from subset of higher informative attributes, that chosen according to several strategies. Decision tree (C4.5) classifier was used as models generator. Third: two voting techniques for predictive unseen class label. ELBRAV integrate the attribute evaluation with classifier that generate the models. ELBRAV was evaluated on seven real cancerous DNA microarray datasets, and its performance evaluated with the most five popular classification method (C45, Bagging(C4.5), AdaBoost(C4.5), random forest and Support Vector Machine SVM ). The proposed method ELBRAV outperforms C4.5 and the traditional ensample methods Bagging(C4.5), AdaBoost(C4.5) and random forest, between (9.67%-4.16%) on average. And outperforms SVM about 1.39% on average. In the proposed method obtain high accuracy and meaningful rules; this make ELBRAV algorithm is very useful for biologists.

**Keywords:** *Decision Tree Modeling, Pattern Recognition, Classification, Ensemble Modeling, Cancerous DNA Microarray.*

## Introduction

Biomedical data is one of the popular domains of data mining applications. DNA Microarray technology provides capability to observe the expression levels of thousands of genes at one time. Microarray data analysis offers the potential for discovering the causes of diseases, and identifying the marker genes which may be the signature of certain diseases. Microarray is an imbalanced data [1], due to microarray data is often obtained via expensive experiments, hard to collect and the number of samples belong to each class are extremely different, as well as microarray dataset contains huge number of attributes vs. few number of samples. The gene expression data obtained from high throughput technologies, such as Affymetrix microarray or Oligonucleotide chips usually organized in a data matrix of *n* rows and *m* columns is known as a gene expression profile [2].

The rows represent genes, and the columns represent the samples. One can carry out two straightforward studies by comparing the genes (*n* rows) or comparing the samples (*m* columns) of the matrix as shown in figure (1). If we find that two rows are similar, we can hypothesise that the two genes are co-regulated and maybe functionally related. These analyses may facilitate our comprehensibility of metabolic and signaling pathways, gene regulation, the genetic mechanisms of disease, and the response to drug treatments [3].

Gene expression Value

*m* sample

*n* genes

| Genes | Condition1 | Condition 2 | ..... | Condition *m* |
|---|---|---|---|---|
| Gene 1 | 103.02 | 58.89 | ..... | 101.54 |
| Gene 2 | 40.55 | 1246.87 | ..... | 1432.12 |
| ..... | ..... | ..... | ..... | ..... |
| Gene *n* | 78.13 | 66.25 | ..... | 823.03 |

**Figure (1): A typical gene expression matrix where rows represent genes and the columns represent the samples**

Considering the amount of the gene expression data, it is impossible for a professional to compute and compare the ***n*×*m*** gene expression matrix manually, where ***n*** is usually greater than 33000 and ***m*** is less than 200 [4]. Thus, machine learning and data mining techniques have been widely used to classify gene expression data, for instance, support vector machines (SVMs) [5, 6], k-nearest neighbor classifier [7], ensemble methods including Bagging and Boosting [4, 8] etc. were applied on microarray dataset. Others researchers have focused their efforts to compare performance of these methods on microarray data analysis such as [9, 10] etc. Others applied gene selection as preprocessing stage of classification task, for instance [11, 12] etc. From our previous study [3, 13], we notice that no classifiers was superior other. Some classifiers gave high classification accuracy such SVM and artificial neural network ANN but difficult to be interpreted and memory consuming. Other classifiers such as rule base and decision tree family gave unsatisfying classification accuracy, but it's still attractive methods in the classification domain due to easy results interpretation, and providing informative attributes. For instance, informative genes are the genes whose expression pattern is strongly correlated with the class distinction [13].

Thus in this work, the performance of decision tree (C4.5) was boosted by propose an ensample classifier based on ensample ranking attribute value (ELBRAV) algorithm, the proposed classifier is focus on building multi-decision tree models via several subset of top ranked attribute and voting among these models.

The rest of paper is organized as follows: Section 2, describes the methodology. Section 3, introduce the proposed algorithm ELBRAV. Section 4, contains description of the dataset. Section 5, shows the experimental result and discussions. Section 6, conclude the paper and future work.

# 1. Methodology

C4.5, Bagging, Boosting, Random forests and SVM are most popular classification methods in the machine learning and data-mining domain [14].

## 1.1 C4.5 Algorithm

Decision tree C4.5 proposed by Quinlan [15]. The algorithm is a successor of ID3, which determines at each step the most predictive attribute, and splits data in the node based on this attribute. Every node represents a decision point over the value of some attribute.
The split criterion calculated as the follows:

- Calculate the expected information needed to classify a tuple in D (dataset )

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

Where, *m* refer to number of classes and $p_i$ be the probability that an arbitrary tuple in D belongs to class $C_i$, estimated by $|C_{i,D}|/|D|$

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

$$\mathrm{Info}_A(D) = \sum_{i=1}^{v} \frac{|D_j|}{|D|} \times \mathrm{Info}(D_j) \qquad (2)$$

Where, v refer to the number of categories related to attribute A, $|D_j|$ refer to number of tuples related to category j, $|D|$ refer to total number of tupels in the dataset and $\frac{|D_j|}{|D|}$ is the probability of tuple belonging to class Cj, its acts as the weight of the jth partition.

- Calculate information gain of attribute A.

$$\mathrm{Gain}(A) = \mathrm{Info}(D) - \mathrm{Info}_A(D) \qquad (3)$$

- Calculate split information of attribute A

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

- Calculate gain ratio:

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

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

C4.5 use supervised discretization continuous values by information gain ratio. This mean no discretization pre-process is required for this algorithm. Figure (2) shows pseudo code of discretization algorithm that imbedded in C4.5.

```
Input (class label, attribute_i)// dataset D;
calculate expected information needed to classify a tuple in D;  // as shown in equation (1)
for each attribute_i,value
   {
      sort the value of attribute_i increasingly;
      calculate Median between each value as candidate split point;
      for each median_i
         {
          Calculate information gain ratio (upper and lower of the median); // equation (2)(3)(4)(5)
         }
      out best cut point  for attribute_i (maximum gain ratio);
   }
Out best split attribute and best cut point related to this attribute;
```

**Figure (2): Supervised discretization algorithm that imbedded in C4.5**

## 1.2 Bagging Algorithm

Bagging produced by Leo Breiman [16], it aims to manipulate the training data by randomly replacing the original *T* training data by *N* items. The replacement training sets are known as bootstrap replicates in which some instances may not appear while others appear more than once. The final classifier $C^*(x)$ is constructed by aggregating $C_i(x)$ where every $C_i(x)$ has an equal vote. Bagging algorithm is shown in figure (3).

**Input:** Training examples <*x*, *y*>, Data Mining Algorithm *DM* (default decision tree ), Integer *j* (number of iteration)

For each iteration $i = 1 \ldots j$

{

Select a subset *t* of size *N* from the original training examples *T*

The size of *t* is the same with the *T* where some instances may not appear in it while others appear more than once (re-sampling)

Generates a classifier $C_i(x)$ from the *t*

}

The final classifier $C^*(x)$ is formed by aggregating the *j* classifiers

To classify an instance *x*, a vote for class *y* is recorded by every classifier $C_i(x) = y$

$C^*(x)$ is the class with the most votes. (Ties being resolved arbitrarily.)

**Output:** $C^*(x)$

**Figure (3): Bagging algorithm [16]**

## 1.3 AdaBoost Algorithm

AdaBoost proposed by Freund and Schapire [17] is an alternative method to influence the training data. Initially, the algorithm assigns every instance $x_i$ with an equal weight. In each iteration *i*, the learning algorithm tries to minimize the weighted error on the training set and returns a classifier $C_i(x)$. The weighted error of $C_i(x)$ is computed and applied to update the weights on the training instances $x_i$. The weight of $x_i$ increases according to its influences on the classifier's performance that assigns a high weight for a misclassified $x_i$ and a low weight for a correctly classified $x_i$. The final classifier $C^*(x)$ is constructed by a weighted vote of the

individual $C_i(x)$ according to its accuracy based on the weighted training set. Figure (4) describes AdaBoost.

**Input:** Training examples <*x*, *y*>, Data Mining Algorithm *DM* (default decision tree ), Integer *j* (number of iteration)
Assigns an equal weight for instance $x_i$
**For** each iteration $i = 1 \ldots j$
{
Generates a classifier $C_i(x)$ with minimize the weighted error over the instances *x*
Update the weight of $x_i$
}
The final classifier $C^*(x)$ is formed by a weighted vote of the individual $C_i(x)$ according to its accuracy on the weighted training set
**Output**: $C^*(x)$

**Figure (4): AdaBoost algorithm [17]**

## 1.4 Random Forest Algorithm

Random forest developed by Leo Breiman [18], specifically designed for decision tree classifiers that uses multi binary decision trees that has roots in CART. Each of the classification trees is built using a sample of data and at each node, a randomly chosen set of variables is considered for the best split. Random forest combines bagging and random feature selection methods to generate multiple classifiers. First, bootstrap is adopted to form a resampled training data set $D_i$ from which $T_i$ will be constructed. During the $T_i$ constructing stage, at each node a fixed number of features is selected randomly for splitting on. Two features are tried among the selected set of features and the one with the higher information gain ratio is selected to split the training data set. Random forest combines bagging and random feature selection methods to generate multi classifiers. Random Forests described in figure (5).

Initially select the number of *K* of trees to be generated; samples *S* in *training* set *D*; Tree T
**for** i=1 to K **do**
$D_i$ = bootstrap sample from *D* (sample with replacement) *A* Vector $\theta$ is generated (random selected genes for each node) Construct Tree $T_i = (s, \theta)$ using any decisions tree algorithm
**end**
Each Tree casts 1 vote for the most popular class at *S*
$C^*(s) = \arg\max \sum_{i:C_{i(s)=c}} 1$ (The class at is predicted by selecting the class with max Votes)
Output classifier $C^*$

**Figure (5): Random Forests algorithm [18]**

**Hong Hu**. et al [19], proposed new methods, which construct ensample classifier based on C4.5, hence they build first models from original data, in the next iteration they remove attributes which shared to build the previous model, and build the next model from the rest attributes in the dataset , so on. In the classification stage they voting among these models to predictive unseen class label.

Bagging, Boosting and Random forest were used to classify microarray data or to compare other method with it in several studies such as, [4, 10, 12, 19, 20, 21].

## 2. Proposed Algorithm (ELBRAV)

To improve the accuracy and reliability of decision tree (C4.5 classifier) for microarray classification, we propose a new algorithm Ensample Learning Based on Ranking Attribute Value (ELBRAV).

The proposed algorithm intended to benefit from the nature of imbalanced dataset (microarray dataset). Thus, the main objective of ELBRAV is to constructs multi classification models, which can correctly classify the cancerous tissues vs. normal tissues from the gene expression profiles and returns meaningful information for biologists. In ELBRAV, we intend to make hybridization between attribute evaluation and classification algorithm by extract the most important attributes according to information gain ratio criterion (the attribute that has maximum information gain ratio value is the most important attributes than the other gains) and avoid using irrelevant attributes (that has low information gain ratio value). At the next stage, multi-classification models are built from top ranked subset of attributes.

**Definition1**: *jointed/disjointed decision tree models*

*Let the attribute which constructed the model* $\boldsymbol{M_i}$ *is* $Att_{M_i}$ *and the attributes which constructed the model* $\boldsymbol{M_j}$ *is* $Att_{M_j}$ *where* $i \neq j$, $i, j = 1 \ldots$*.total number of attributes-1*

*if* $Att_{M_i} \cap Att_{M_j} \neq \varnothing$ *then the model* $\boldsymbol{M_i}$ *and* $\boldsymbol{M_j}$ *are jointed*

*else if* $Att_{M_i} \cap Att_{M_j} = \varnothing$ *then the model* $\boldsymbol{M_i}$ *and* $\boldsymbol{M_j}$ *are disjointed*

**Definition2**: *Fixed Interval/Incremental Interval between the top ranking attributes*

*Let the dataset contains* ***n*** *number of attributes that arranged according to its values witch obtained by information gain ratio criterion.*

***If*** *the model* $\boldsymbol{M_1}$ *constructed from* $\boldsymbol{Att_1}$ *to* $\boldsymbol{Att_k}$ *and model* $\boldsymbol{M_2}$ *constructed from* $\boldsymbol{Att_{k+1}}$ *to* $\boldsymbol{Att_{2*k}}$ *and the model* $\boldsymbol{M_3}$ *constructed from* $\boldsymbol{Att_{2*k+1}}$ *to* $\boldsymbol{Att_{3*k}}$ *etc. in this case k is fixed interval between the top ranking attributes. And the models are constructed from fixed interval.*

***If*** *the model* $\boldsymbol{M_1}$ *constructed from* $\boldsymbol{Att_1}$ *to* $\boldsymbol{Att_k}$ *and model* $\boldsymbol{M_2}$ *constructed from* $Att_{k+1}$ *to* $\boldsymbol{Att_{2(2*k)}}$ *and the model* $\boldsymbol{M_3}$ *constructed from* $\boldsymbol{Att_{2(2*k)+1}}$ *to* $\boldsymbol{Att_{3(3*k)}}$ *etc. in this case the models are from incremental interval.*

**Definition3***: simple voting / Weighted voting,*

*In simple voting the final decision, commits of the ensample classifier dos not take into account the individual accuracy of the models. But in* ***Weighted voting*** *the final decision, commits of the ensample classifier take into account the individual accuracy of the models.*

**The ELBRAV** algorithm consists of three main stages:

**(1) Ranking attributes (genes):** is a preprocessing stage**,** to extract the high informative attributes and reordering these attribute increasingly according to its value which obtained by IGRAE. Due to ELBRAV dealing with continues attribute, a supervised discretization of the continuous attribute is used to evaluate candidate cut points according to information gain ratio criterion [22] as shown in figure (2).

**(2) Multi models Construction**: The aim of this step is to construct multi models by resampling top ranked attributes. In ELBRAV C4.5 classifier was used as models generator to builds the multi-classification models. In this stage, several options were provided to construct jointed/disjointed models as input parameters. For instance (1) numbers of models to be constructed, (2) the interval between the subset of top ranking attributes which sharing to builds each models, (3) chosen fixed or incremental between the attributes, (4) remove or keep the attributes that used to constructing previous models. These options provided to obtain maxima diversified among models. The benefits of this option is one gene containing noise or missing values only affects on one model but not on all models.

**(3) Classification stage**: in the previous stages kth model generated with different accuracy. To avoid this problem, the final predicted class of incoming unseen sample is determined by two options. First one simple voting, this mean all models have an equal weight in a decision commit, the second option weighted voting this mean we calculate the accuracy of each model and give this the accuracy as weight of these models, and the final decision of ELBRAV comes from voting among all models.

In ELBRAV all models ware built on equal samples number of the original dataset, this avoids unreliability of voting strategy. We evaluate ELBRAV based on 10-fold cross validation method. Figure (6) shows the ELBRAV block diagram.

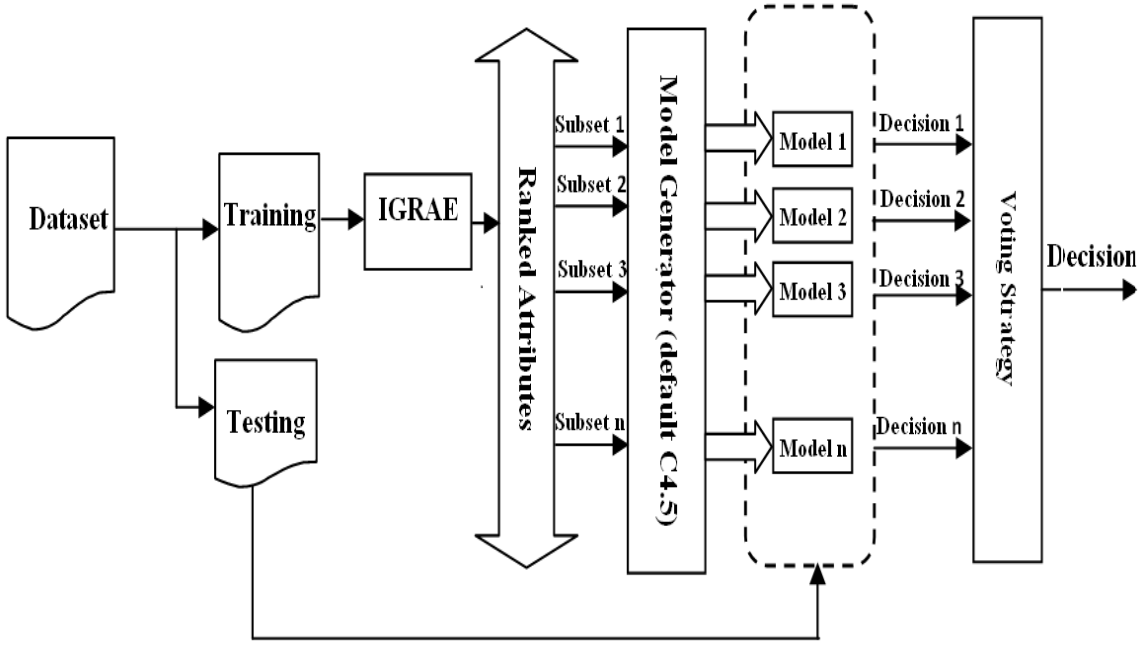


**Figure (6): Ensample Learning Based on Ranking Attribute Value (ELBRAV) block diagram**

The complete list of notation and options which used in ELBRAV is given in table (1) and pseudo code of ELBRAV is shown in figure (7).

**Table (1) Notations and options which used in ELBRAV**

| Symbol | Notations |
|---|---|
| D | The Microarray Dataset |
| D' | Ranked Microarray Dataset via IGRAE criterion |
| MG | models generator (default C4.5 ) |
| n | Number of Generated models |
| RM | Ranking Method (default IGRAE) |
| c | classes in the dataset |
| x | Testing Microarray sample |

**Table (1) Notations and options which used in ELBRAV (con.)**

| Symbol | Options |
|---|---|
| I | Interval between the top ranking attributes option: the Attributes to be construct the model $i$ ($M_i$) is chosen as<br>• $Att_{M_i} = Att_{M_{i-1}} \cup Att_{i+I}$ for Fixed Interval option<br>• $Att_{M_i} = Att_{M_{i-1}} \cup Att_{i*I}$ for Incremental Interval option |
| R | Keep/ Remove : attributes used to construct $M_{i+1}$ option (R=1 then remove else Keep) |
| W | simple voting / Weighted voting option (W=0 then simple voting else Weighted voting ) |

| | |
|---|---|
| **1-** | **Ranking (D)**<br>**Input** (D,RM)<br>Use Attribute evaluation criterion (default IGRAE)<br>**Output D'** //Ranked dataset according to IGRAE criterion. (the attribute has maximum IGRAE value at the first) |
| **2-** | **Model Constructing (D', MG, Options)**<br>**Input** A ranked microarray dataset D', MG, Options;<br>let M= ⌀;<br>**for** i=1 to n; //number of models<br>apply options;<br>**Call** MG to generate model $M_i$; // (default MG is C4.5 classifier)<br>M =M ∪ $M_i$;<br>**end for;**<br>**Output** n number of jointed/disjointed models M; |
| **3-** | **CLASSIFY**$_{ELBRAV}$<br>**Input**: **n** number of models **M** , testing samples x, W (simple voting / Weighted voting options);<br>Read test sample x, **n** models;<br>**for** i=1 to n<br>Classify (x);<br>Calculate accuracy $\mathbf{M_i}$ ;<br>**end for**<br>**apply** voting option<br>**if** simple voting **then**<br>$C(x) = arg\ max \sum_{i=1}^{n} C_i(x)$; //the most often predicted class label C<br>**else** Weighted voting<br>$C(x) = arg\ max \sum_{i=1}^{n} C_i(x) * accuracy_{\mathbf{M}_i}$;<br>Calculate confusion matrix of ELBRAV<br>**Output** class label of x ($C(x)$) , confusion matrix$_{ELBRAV}$ |

**Figure (7): pseudo code of Ensample Learning Based on Ranking Attribute Value (ELBRAV)**

## 2.1 ELBRAV Implementation

ELBRAV was used C4.5 algorithm as models generator, due to the advantages of decision tree modeling over other pattern recognition methods and it classified with top 10 algorithms in data mining [14], which able to generate understandable knowledge structures and set of

meaningful rules. ELBRAV contains several imbedded algorithms such as supervised discretization algorithm, C4.5 algorithm and ELBRAV options were implemented from the beginning; the code of ELBRAV written using C# 2010 programming language.

## 3. Datasets

To evaluate the performance of ELBRAV seven datasets were collected from European Bioinformatics Institute (EBI) available online (http://-www.ebi.ac.uk/arrayexpress), and Gene Expression Omnibus (GEO) [23]. Datasets had been generated by Affymetrix Gene Chip technology [24, 25]. Table (2) provides briefly description of microarray datasets continents.

**Table (2): Description of microarray datasets, where AML: acute myeloid leukemia, ALL: acute lymphoblastic leukemia, AD: adenocarcinomas, SQ: squamous cell carcinomas, COID carcinoids, and NL: normal lung. BR: Breast, PR: prostate, LN: lung, CO: colon, PA: patient, CO: control**

| Dataset | Samples NO. | Genes No. | Category | | | |
|---|---|---|---|---|---|---|
| | | | tumor | | normal | |
| Breast [26] | 62 | 16383 | 43 | | 19 | |
| Colon [27] | 36 | 7458 | 18 | | 18 | |
| Lung2 [28] | 88 | 16382 | 69 | | 19 | |
| Prostate2 [29] | 108 | 12554 | 92 | | 16 | |
| Lung1 [26] | 197 | 10937 | AD | NL | SQ | COID |
| | | | 139 | 17 | 21 | 20 |
| Leukemia [30] | 72 | 7130 | ALL | | AML | |
| | | | 47 | | 25 | |
| Lymphoma [31] | 40 | 16381 | PA | | CO | |
| | | | 40 | | 20 | |

## 4. Results and Discussions

ELBRAV performance is evaluated with five well known single and ensemble decision tree algorithms, namely J48, Random Forests, AdaBoost(J48), Bagging(J48), and SVM. These algorithms are existent in Weka 3.7 package [32]. The algorithms have been used without change in the default options in weka. We are aware that the accuracy of some algorithms on some datasets can be improved, when options are changed, but it is difficult to find a uniform good setting for all datasets. Thus, the default settings were not changed, since the default settings produced high accuracy on average. The main default setting for AdaBoost(J48) and Bagging(J48) is number of performed iterations which equal to 10, for Random forest main default setting is number of trees to be constructed where equal to 10 trees, for SVM the main default setting is the kernel function where Linear Kernel function is used.

The difference between our implementation of C4.5 and J48 (C4.5 weka implementation) came from the mechanism of assigning the best cut point that related to each attribute. In weka the best cut point assigned at the same value of the attribute. In our implementation we take the median between two values of attribute. The evaluating of C4.5 and J48 was based on **ten-fold cross-validation evaluation** method, its notice from table (3), C4.5 gave classification accuracy better than J48 in majority of the datasets.

Due to wide options available in ELBRAV, it is useful to extract the best number of models and best number of intervals that gives higher classification accuracy. Thus, ELBRAV has applied many times with different options: where ELBRAV at matrix of options (3, 5, 7, 9, 11, 15 and 21) models vs. (3, 5, 10, 15, 20, 30, 40 and 50) Intervals, 10-fold cross validation

technique has been used. Figure (8) shows the average accuracy of ELBRAV as function of number models and figure (9) shows the average accuracy as function of number of interval. From figure (8) its notes nine-models is the best number of models on majority of the datasets. And from figure (9) the fifteen- intervals is the best number of intervals between subset of top ranked genes on majority of the datasets. The average accuracy can be predicted as follows|:

$$Accuracy = 0.081 * (Number\ of\ Models) + 0.0084 * (Number\ of\ Intervals) + 96.6864$$

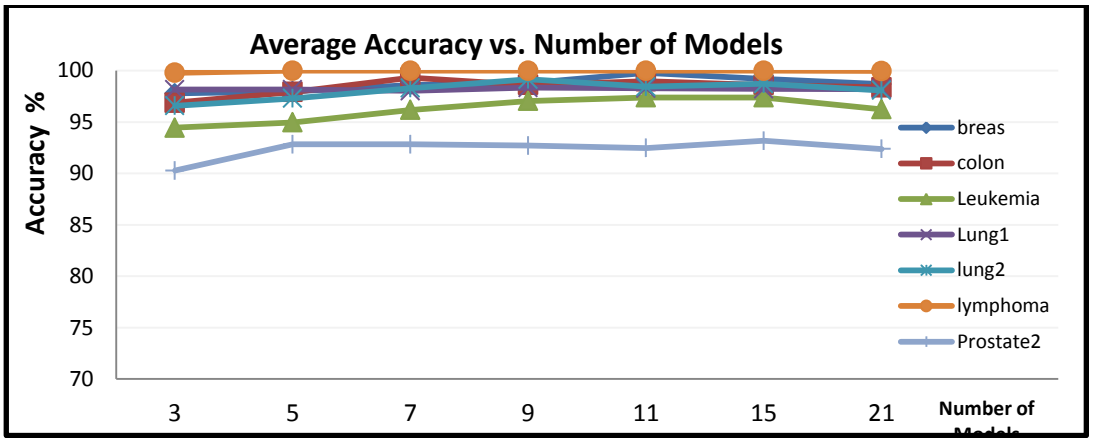


**Figure (8): The average accuracy of ELBRAV as function of number of models**

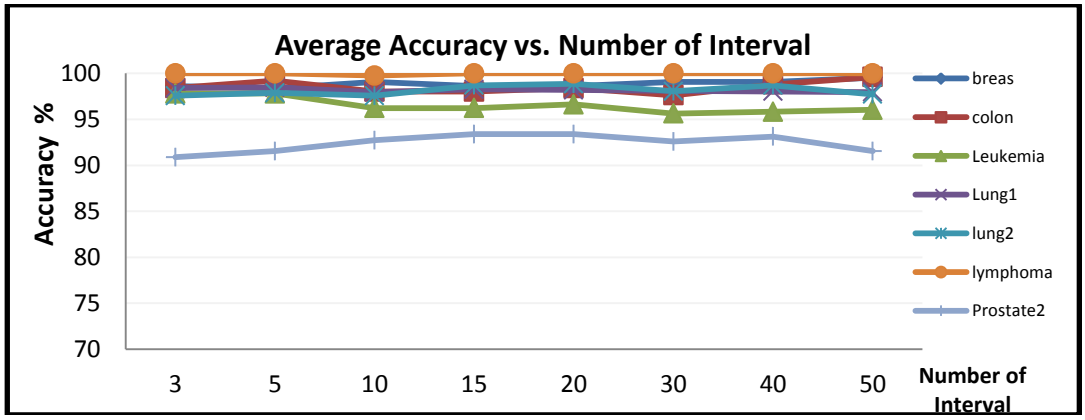


**Figure (9): The average accuracy of ELBRAV as function of number of intervals**

The average accuracy and standard deviations with all previous options were calculated as its shows in table (3).

**Table (3): the average accuracy obtained by ELBRAV by(3, 5, 7, 9, 11, 15 and 21 models) vs. (3, 5, 10, 15, 20, 30, 40 and 50 Intervals) and standard deviation**

| Dataset | Average Accuracy of ELBRAV % | STDEV |
|---|---|---|
| Breast | 98.82 | 1.5 |
| Colon | 98.54 | 1.82 |
| Leukemia | 96.53 | 1.98 |
| Lung1 | 98.18 | 0.49 |
| Lung2 | 98.13 | 1.36 |
| Lymphoma | 99.97 | 0.22 |
| Prostate2 | 92.36 | 1.75 |

As mentioned, ELBRAV is compared with most five popular algorithms that can be found it in weka package. on the seven different datasets. Thus by fixed interval between the top ranked genes to 15 genes and set the number of generated models to 9 models. The results that obtained is organized in table (4). Table (4) shows individual and average accuracy of the five Weka methods and ELBRAV. The evaluation was based on 10-fold cross-validation method for all methods. Its notice ELBRAV (Weighted voting) is outperforms J48, Bagging (J48), Adaboost (J48), Random forest and SVM on average (9.67%. 4.58%, 4.16% 5.34%, 1.39%) respectively. ELBRAV(simple voting) is outperforms J48, Bagging (J48), Adaboost (J48) and Random forest on average (7.45%, 2.36%, 1.94% and 3.12%) respectively. but SVM outperform ELBRAV(simple voting) 0.83% on average. In ELBRAV we can simply obtain 100% accuracy on all dataset by training ELBRAV with different options and chose the models which give 100% accuracy, but we fixed the condition to be fairly in our evaluation.

**Table (4): Individual and average accuracy results of the five Weka methods as well as C4.5 5 core of ELBRAV and ELBRAV (simple voting / Weighted voting)**

| | C4.5 core of ELBRAV | J48 | ELBRAV 9 Model and 15 fixed Interval. | | Bagging (J48) | Adaboost (J48) | Random forest | SVM |
|---|---|---|---|---|---|---|---|---|
| | | | Weighted voting | simple voting | | | | |
| Breast | 90.32 | 88.71 | **100** | **100** | 98.38 | 96.77 | 98.39 | **100** |
| Colon | 88.89 | 91.66 | **100** | 94.44 | 94.44 | 97.22 | 94.44 | 97.22 |
| Leukemia | 87.5 | 79.17 | **98.61** | 94.4444 | 88.89 | 86.11 | 86.11 | 97.22 |
| Lung1 | 97.97 | 90.86 | **98.98** | 97.4619 | 93.4 | 93.4 | 93.91 | 95.43 |
| Lung2 | 95.45 | 92.05 | **100** | 96.59 | 93.18 | 97.73 | 96.59 | 98.86 |
| Lymphoma | 95 | 96.67 | **100** | **100** | **100** | **100** | **100** | **100** |
| Prostate2 | 85.19 | 85.19 | **94.44** | 93.52 | 91.67 | 91.67 | 85.19 | 93.52 |
| Average | 91.47 | 89.19 | **98.86** | 96.64 | 94.28 | 94.7 | 93.52 | 97.46 |

## 5. Conclusion and Future Work

In this paper, we studied imbalanced biomedical data such as microarray datasets. a new proposed and implemented ensample algorithm, namely Ensample Learning Based on Ranking Attribute Value (ELBRAV) is presented. Its performance accuracy evaluated with the most five popular classification methods C45, Bagging(C4.5), AdaBoost(C4.5), Random Forest and Support Vector Machine SVM. The evaluation is performed using seven real cancerous DNA microarray datasets related to various diseases.

The power of ELBRAV came from containing a lot of options, to assure the constructed decision tree models are highly accurate. For instance; by constructing *disjointed* models this makes ELBRAV robust against noise, in case some genes have missing values the classifier committee is not affected due to ELBRAV allows constructing the disjointed models by removing the genes which used in a previous models. Other important ELBRAV options is voting strategy, ELBRAV contains two techniques for final decision committee (simple voting / Weighted voting) this makes ELBRAV more accurate, due to the models which gave high classification accuracy shared with the final decision more than inaccurate models.

ELBRAV is different from a traditional ensample algorithms such as Bagging and Adaboost, due to in ELBRAV ensample attributes instead of resample the tuples. Also ELBRAV differs from Random Forest that resample the attributes randomly and used CART (Classification and regression tree) as models generator, but ELBRAV ranked attributes at the

first stage by information gain ratio attribute evaluation IGRAE criterion and use C4.5 as models generator in addition to voting strategy.

The proposed method ELBRAV outperforms traditional ensample methods Bagging(C4.5), AdaBoost(C4.5) and random forest, the values of enhancement fluctuate between (9.67%-4.16%) on average and outperforms SVM about 1.39% on average. by ELBRAV can be obtains 100% accuracy by choosing the appropriate options for individual dataset or by using the highest accuracy models only. ELBRAV is suitable to classifying microarray data and faster than ensample algorithms, due to at each iteration ensample algorithms must deal with all attributes, while ELBRAV ranked attributes one-time and deal with few of top ranked attributes for each models. In addition to ELBRAV resolve disadvantage of single decision tree, due to each instance can be covered by more than one rule according to the number of generated models. As will as C4.5 must deal with all attributes minus one for each dataset splitting.

Information gain ratio attribute evaluation IGRAE was used for ranking genes this make ELBRAV is useful for biologists. Other criteria can be used to find top ranking genes too such as Chi-square ($\chi 2$) Attributes Evaluate, Partial Least-Squares (PLS) etc. and be using other models generators which makes ELBRAV general ensample algorithm.

## References

[1] Sangyoon Oh, Min Su Lee and Byoung-Tak Zhang "Ensemble Learning with Active Example Selection for Imbalanced Biomedical Data Classification", IEEE International Conference on Bioinformatics and Biomedicine, pp 1-10, 2010.

[2] Maureen R. Gwinna and Ainsley Westonb, "Application of Oligonucleotide Microarray Technology to Toxic Occupational Exposures", Journal of Toxicology and Environmental Health, Part A, vol. 71, Issue 5,pp. 315-324, 2008.

[3] 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, pp. 73-82, 2011.

[4] Tan, A. C. & Gibert, D., "Ensemble machine learning on gene expression data for cancer classification", Applied Bioinformatics , vol. 2(3), pp. 75–83. 2003.

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

[6] Brown, M., Grundy, W., Lin, D., Cristianini, N., Sugnet, C., Furey, T., Jr, M. & Haussler, D., "Knowledge-based analysis of microarray gene expression data by using support vector machines", in 'Proc. Natl. Acad. Sci.', vol. 97, pp. 262–267, 2000.

[7] Shiliang Sun and Rongqing Huang, "An adaptive k-nearest neighbor algorithm", IEEE, Seventh International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), pp. 91 – 94, 2010.

[8] Dietterich, T. G., "An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization", Machine learning vol. 40, pp.139–157, 2000.

[9] Hong Chai and Carlotta Domeniconi, "An Evaluation of Gene Selection Methods for Multi-class Microarray Data Classication", Data Mining and Text Mining in Bioinformatics, pp7-15, 2004.

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

[11] Taeho Hwang, Choong-Hyun Sun, Taegyun Yun and Gwan-Su Yi, 1,2 "FiGS: a filter-

based gene selection workbench for microarray data", BMC Bioinformatics, pp. 1-6, 2010.

[12] 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, pp. 108–117, 2010.

[13] M. B. Senousy, H. M. El-Deeb, K. Badran and I. A. Al-Khlil, "Gene Ranking Techniques via Attribute Evaluation Algorithms for DNA Microarray Analysis", ASAT international conference on Aerospace Sciences & aviation technology, pp. 1-14, 2011.

[14] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J.Hand, and D. Steinberg, "Top 10 algorithms in data mining", Knowl. Inf. Syst, vol. 14(1), pp.1–37, 2008.

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

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

[17] Freund, Y. & Schapire, R. E., Experiments with a new boosting algorithm, in 'International Conference on Machine Learning', pp. 148–156, 1996.

[18] L. Breiman, "Random forests–random features", Technical Report 567, University of California, Berkley, 1999.

[19] Hong Hu, Jiuyong Li, Hua Wang, Grant Daggard and Mingren Shi "A Maximally Diversified Multi Decision Tree Algorithm for Microarray Data Classification", Conferences in Research and Practice in Information Technology (CRPIT), vol. 73, pp. 1-6, 2006.

[20] Shital Shah, and rew Kusiak, "Cancer gene search with data-mining and genetic algorithms" Computers in Biology and Medicine vol. 37, pp. 251–261, 2007.

[21] AnthonyJ.Myles,Robert N.Feudale, Yang Liu, Nathaniel A. Woody and Steven D. Brown, "An introduction to decision tree modeling" JOURNAL OF Chemometrics, vol. 18, pp. 275–285, 2004.

[22] Sotiris Kotsiantis, Dimitris Kanellopoulos, "Discretization Techniques: A recent survey", GESTS International Transactions on Computer Science and Engineering, vol.32 (1), pp. 47-58, 2006.

[23] Barrett, T., Troup, D.B., Wilhite, S.E., Ledoux, P., Rudnev, D., Evangelista, C., Kim, I.F., Soboleva, A., Tomashevsky, M., Edgar, R., "NCBI GEO: mining tens of millions of expression profiles-database and tools update", Database issue Nucleic Acids Res. vol. 2007.

[24] R.J. Lipshutz, S.P.A. Fodor, T.R. Gingeras, and Lockhart, D.J., "High density synthetic oligonucleotide arrays", Nature genetics, vol. 21 No. l, pp. 20-24, 1999.

[25] aidong zhang, "advanced analysis of gene expression microarray data", Published by World Scientific Publishing Co. Pte. Ltd., 2006.

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

[27] 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. 96(12), pp. 6575–6576,1999.

[28] 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.

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

[30] 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, pp.531–537, 1999.

[31] 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, pp. 503–511, 2000.

[32] 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.