An Efficient Clustering-Classification Method in an Information Gain NRGA-KNN Algorithm for Feature Selection of Micro Array Data

Akey Sungheetha[1], Dr. J.Suganthi[2]

[1] Research Scholar, Department of Information Technology, Anna University, Chennai – 600 025, Tamil Nadu, India
[2] Professor and Head, Department of Computer Science and Engineering, Hindusthan College of Engineering and Technology,
Othakalmandapam, Coimbatore – 641 032, Tamil Nadu, India
sun29it@gmail.com [1], sugi_jeyan@hotmail.com [2]

Abstract: Gene expressions by microarray data technique have been effectively utilized for classification and diagnostic of cancer nodules. Numerous data mining techniques like clustering are presently applied for identifying cancer using gene expression data. An unsupervised learning technique is a clustering technique used to find out grouping structure in a set of data. The problem of feature selection in clustering algorithm is what type of data attributes used is not known and also for data there is no class labels so there is no clear criteria to direct the search. A further issue in clustering is the identification of the number of clusters that affects the performance of feature selection. Gene expression database have a great potential as a medical diagnostic tool since they represent the state of a cell at the molecular level. Training data sets is available for the classification of cancer types generally have a fairly small sample size compared to the number of genes involved. Feature selection is considered to be a problem of optimization in machine learning, reduces the number of features, noisy and redundant data, and results in acceptable classification accuracy. Hence, selecting significant genes from the microarray data poses a dreadful challenge to researchers due to their high-dimensionality features in clustering technique and the usually small sample size. A clustering algorithm is proposed, which is a hybrid model of information gain genetic algorithm for feature selection in microarray data sets. Information Gain (IG) was used to select important feature subsets (genes) from all features in the gene expression data, and a Non-Dominated Ranked Genetic Algorithm (NRGA) was employed for actual feature selection. The K-NN method is used to evaluate the NRGA algorithm. Experimental results show that the proposed clustering based method simplifies the number of gene expression levels effectively and gives accurate feature selection while compared with other methods.
[Akey Sungheetha, Dr. J.Suganthi. **An Efficient Clustering-Classification Method in an Information Gain NRGA-KNN Algorithm for Feature Selection of Micro Array Data.** *Life Sci J* 2013; 10(7s): 691-700] (ISSN:1097-8135). http://www.lifesciencesite.com. 108

Keywords---Feature Selection, Gene Expression, Genetic Algorithm, Non-Dominated Ranked Genetic Algorithm, Information Gain, K-nearest neighbor (K-NN)

1. Introduction

The goal of clustering is to determine a natural combination in a group of patterns, points, or objects, without knowledge of any class labels. Clustering is widespread in any discipline that involves analysis of multivariate data. It is, of course, impractical to exhaustively list the numerous uses of clustering techniques. In the background of the human genome development, new technologies were emerged, it facilitate the parallel execution of experiments on a large number of genes at the same time. Hence it is called as DNA microarrays, or DNA chips, constitute a prominent example. This technology aims at the measurement of mRNA levels in particular cells or tissues for many genes at once. To this end, single strands of balancing DNA for the genes of interest which can be immobilized on spots arranged in a grid on a support which will typically be a glass slide, a quartz wafer, or a nylon membrane. Measuring the quantity of label on each spot then yields an intensity value that should be correlated to the abundance of the corresponding RNA transcript in the sample [1].

The parallelism in this kind of experiment lies in the hybridization of mRNA extracted from a single sample to many genes at once using clustering technique. The measured values are not obtained on an absolute scale. Because it depends on many factors such as the efficiencies of the various chemical reactions involved in the sample preparation, as well as on the amount of immobilized DNA available for hybridization. The class of transcripts that is probed by a spot may differ in different applications. Most commonly, each spot is meant to probe a particular gene. The representative sequence of DNA on the spot may be either a carefully selected fragment of cDNA, a more arbitrary PCR product amplified from a clone matching the gene [3]. Another level of

sophistication is reached when a spot represents, e.g., a particular transcript of a gene. In this case or for the distinction of mRNA abundances of genes from closely related gene families, careful design and selection is made of the immobilized DNA are required. Similarly, the selection of samples to study and to compare to each other using DNA microarrays requires careful planning as will become clear upon consideration of the statistical questions arising from this technology [2] [4].

Microarray data samples classification involves feature selection and classifier design. Generally, only a small number of gene expression data show a strong correlation with a certain phenotype compared to the total number of genes investigated. It means that of the thousands of genes investigated; only a small number show significant correlation with a certain phenotype. Consequently, in order to analyze gene expression profiles correctly, feature (gene) selection is crucial for the classification process. The goal of feature selection is to identify the subset of differentially expressed genes that are potentially relevant for distinguishing the sample classes. A good selection method for genes relevant for sample classification is based on the number of genes investigated is needed to increase the predictive accuracy and to avoid incomprehensibility.

Several methods have been used to perform feature selection, e.g., genetic algorithms [5], branch and bound algorithms [6] [7], sequential search algorithms [8], mutual information [9], tabu search [10], entropy-based methods, regularized least squares, random forests, instance-based methods, and least squares support vector machines. A two-stage method is used to implement feature selection. In the first stage, an information gain (IG) value was calculated each gene (feature). In the second stage, all the selected features must conform to a threshold. Consequently, feature selection was once again performed, this time capitalizing on the NRGA's unique attributes to select the features. The K-nearest neighbor method (K-NN) with leave-one-out cross-validation (LOOCV) based on Euclidean distance calculations served as an evaluator of the NRGA for more classification problems taken from the literature. This procedure improved the performance of populations by having a chromosome approximate a local optimum, reducing the number of features based on clustering method, and preventing the NRGA from getting trapped in a local optimum.

2. Related Work

Different clustering algorithms and methods have been developed to overcome the drawbacks of the previous techniques and to enhance the performance [11]. There is no absolute clustering method that can be universally used to solve all problems. So in order to select or generate a suitable clustering strategy, it is vital to investigate the features of the problem.

As Xu and Wunsch [12] revealed the step is usually combined with the selection of a corresponding proximity measure and the construction of a criterion function. Patterns are grouped according to whether they resemble each other. Once a proximity measure is selected, the construction of a clustering condition function makes the partition of clusters an optimizing problem.

K-means is a form of partition-based clustering technique mainly utilized in clustering gene expression data [13]. K-means is well known for its simplicity and speed. It performs quite well on large datasets. However, it may not provide the identical result with each run of the algorithm. It is observed that, K-means is very good at handling outliers but its performance is not satisfactory in detecting clusters of random shapes.

A Self Organizing Map (SOM) [14] is more robust than K-means for clustering noisy data. Due to the noisy data there would be some miscalculation in the accuracy. The input required is the number of clusters and the grid layout of the neuron map. Prior identification of the number of clusters is tough for the gene expression data. Furthermore, partitioning approaches are restricted to data of lower dimensionality, with intrinsic well-separated clusters of high density. Thus partitioning approaches do not perform well on high dimensional gene expression data sets with intersecting and embedded clusters. A hierarchical structure can also be built based on SOM such as Self-Organizing Tree Algorithm (SOTA) [15]. Fuzzy Adaptive Resonance Theory (Fuzzy ART) [16] is another form of SOM which measures the coherence of a neuron (e.g., vigilance criterion). The output map is accustomed by splitting the existing neurons or adding new neurons into the map, until the coherence of each neuron in the map satisfies a user specified threshold.

3. Information gain with NRGA for feature selection

3.1 Information Gain

Information gain (IG) is a feature ranking method based on decision trees that exhibits good performance [17]. Information gain used in feature selection constitutes a filter approach. The idea behind IG is to select features that reveal the most information about the classes. Ideally, such features are highly discriminative and occur in a single class [18]. Information gain is a measure based on entropy; it indicates to what extent the whole entropy is reduced if knows the value of a specific attribute.

Therefore, IG value indicates how much information this attribute contributes to the data set [17]. Each feature has its own IG value which determines whether this feature is to be selected or not. A threshold value is used for checking the features; if a feature has a greater IG value than the threshold, the feature is chosen; or else, it is not selected. Clustering is then done by learning the parameters of these models and the associated probabilities.

Let S be the set of n instances and C be the set of k classes. $Probability(Class_i, S)$ represents the fraction of the example in S that has class C_i. Then, the expected information from this class membership is given by:

$$I(S) = -\sum_{i=1}^{k} Probability(Class_i, S) \times \log(Prob(Class_i, S))$$

If a particular attribute A has v distinct values, the expected information is obtained by the decision tree in which A is the root, and the weighted sum of expected information of the subsets of A is based on the distinct values. Let S_i be the set of instances and A_i the value of attribute A:

$$I_A(S) = -\sum_{i=1}^{v} \frac{|S_i|}{|S|} \times I(S_i)$$

Then, the difference between $I(s)$ and $I_A(S)$ provides the information gained by partitioning S according to the test A

$$Gain(A) = I(S) - I_A(S)$$

A higher information gain will result in a higher likelihood of obtaining pure classes in a target class. After calculating the information gain values for all features, a threshold for the results was established. Since the results show that most IG values are zero after the computation process, not many features have an influence on the category in a data set, signifying that these features are irrelevant for classification. Threshold was 0 for most of the data sets. If the information gain value of the feature was higher than the threshold, the feature was selected; if not, the feature was not selected according to the clustering technique.

3.2 Genetic Algorithm

Genetic algorithms (GAs) are stochastic search algorithms modeled on the process of natural selection underlying biological evolution. They can be applied to many search, optimization, and machine learning problems [19,20,21]. The algorithm is proceeded in iterative manner. Each string is the encoded binary, real etc., account of a candidate result. An evaluation function acquaintances a fitness measure with every string and indicates its fitness for the problem.

GAs have been successfully applied on a variety of problems, including scheduling problems [22], machine learning problems [23,24], multiple objective problems [25,26], feature selection problems, data mining problems [27], and traveling salesman problems [28].

Recent research has identified some drawbacks in GA performance [32]. Limitations of genetic Algorithm is

- Slow convergence
- Lacks of Rank based fitness function
- Time consuming

The proposed approach uses the Non-Dominated Ranked Genetic Algorithm for the optimization purpose. The main advantages of using Non-Dominated Ranked Genetic Algorithm are that it converges very significantly than GA. Moreover, it provides rank based fitness function and it is quicker than GA.

3.3 K-Nearest Neighbor

The K-nearest neighbor (K-NN) is one of the most popular nonparametric methods [29, 30]. The advantage of the K-NN method is its simplicity and easy implementation. K-NN is not negatively affected when the training data is large, and is indifferent to noisy training data [29]. The feature subset was measured by the Leave-One-Out Cross-Validation of one nearest neighbor (1-NN).

Neighbors are calculated using their Euclidean distance. The 1-NN classifier does not require any user-specified parameters, and the classification results are implementation independent.

3.4 Proposed System - NRGA Algorithm

The two different feature selection models GA and KNN for microarray data classification were combined to select relevant genes. In the first-stage, IG, a filter method, was used to select informative genes. Initially, calculate the information gain values (IG values) for eleven gene expression data sets by Weka [31]. Information gain values were calculated for each gene in the microarray data sets by IG, and then the features were sorted in accordance with their information gain values. A feature with a higher information gain value indicates higher discrimination of this feature compared to other categories and means that the feature contains gene information useful for classification.

In the following example, gene expression data sets contain nine genes (features) which can be represented by F1, F2, F3, F4, F5, F6, F7, F8, and F9.

After the application of IG, the nine information gain scores were: F1 = 0, F2 = 0.4, F3 = 0, F4 = 0.9, F5 = 0, F6 = 1.2, F7 = 0.6, F8 = 0.5, F9 = 0. Since most of the scores were 0, so use 0 as the threshold value.

The five values that were above this threshold value (F2, F4, F6, F7, and F8) were then used to continue implementing the feature selection process in the second-stage. In the second stage the NRGA algorithm is introduced to increase the classification accuracy and searching abilities.

The i^{th} string in the population is selected with a probability proportional to $\mathbf{F}_i$. Since the population size is usually kept fixed in a simple GA, the sum of the probability of each string being selected for the mating pools must be one. Therefore, the probability for selecting the i^{th} string is

$$P_i = \frac{F_i}{\sum_{i=1}^{n} F_i}$$

Where n is the population size,

The NRGA algorithm is shown below. At first, a random parent population P is formed. The random values for $\mathbf{F}_i$ is chosen in the way that the selected random value must be within the limit specified in equation P_i.

The sorting of the population is in accordance with the non-domination. Every solution is allocated a fitness (or rank) equivalent to its non-domination level. Non-domination level of 1 represents the best level, 2 represents the next-best level, etc.

Pseudo code for NRGA Algorithm:

Initialize Population $\mathbf{F}_i$
{

Generate random populations of $\mathbf{F}_i$ – size n
Evaluate population objective values J based on

1-NN for $\mathbf{F}_i$
Assign rank (level) for random Populations of

$\mathbf{F}_i$based on pareto dominance sort
}
{
Ranked based roulette wheel selection
Recombination and mutation
}

$\mathbf{Q} \in \mathbf{F}_i$
for i=1 to g do
for each member of the combined population (PUQ) do
Assign rank (level) based on Pareto-sort
Generate sets of non-dominated fronts
Calculate the crowding distance between members of each front
end for
(elitist) Select the members of the combined population based on least dominated n solution

$t_i$make the population of the next generation. Ties are resolved by taking the less crowding distance
Create next generation
{
Ranked based Roulette wheel selection
Recombination Mutation
}
end for

The features selected during the first-stage were used for feature selection by the NRGA algorithm. The chromosome length represents the number of the features. The bit value {1} represents a selected feature, whereas the bit value {0} represents a non-selected feature. The predictive accuracy of a 1-NN determined by the LOOCV method was used to measure the fitness of an individual. For example, when a 9-dimensional data set (n = 9) is analyzed, any number of features smaller than n can be selected. When the adaptive value is calculated, these five features in each data set represent the data dimension and are evaluated by the 1-NN method. The fitness value for 1-NN evolves according to the LOOCV method for all data sets.

In the LOOCV method, a single observation from the original sample is selected as the validation data, and the remaining observations as the training data. This is repeated so that each observation in the sample is used once as the validation data. Essentially, this is the same as *K*-fold cross-validation where *K* is equal to the number of observations in the original sample.

NRGA algorithm was implemented. Initially, a Population of $\mathbf{F}_i$ is created. Random Populations of $\mathbf{F}_i$ is then generated which is of size N. Then the objective function value of J is evaluated. Rank is assigned to the Population with the best objective values based on the Pareto Dominance sort. Then the selection process is carried out based on the ranked based roulette wheel selection. Then in the reproduction phase, recombination and mutation is carried out. Reproduction phase produces new set of

population $\mathbf{Q} \in F_i$, $\mathbf{T}_{1i}$ & $\mathbf{T}_{2i}$ which are the points in the s-plane. A combined Population (RPUQ) is generated. Rank is assigned to the Population with the best objective values based on the Pareto Dominance sort. The members are selected from the combined population based on least dominated N solution (elitist).

The new population of size N is used for selection. Now, two tiers ranked based roulette wheel selection is applied, one tier to select the front and the other to select solution from the front, here the solutions belonging to the best non-dominated set have the largest probabilities to be selected. Then, in the reproduction phase, crossover and mutation are applied to create a new population RP of size N.

4. Experimental Results and Discussions

Improved Accuracy: Feature selection improves calculation efficiency and classification accuracy in classification problems with multiple features, since not all features necessarily influence classification accuracy. Selecting appropriate features attributes according to the clustering technique which improves the accuracy; on the other hand, selecting inappropriate features attributes compromises the accuracy. Hence, employing appropriate feature selection to select optimal features for a category results in higher accuracy. To compute the cluster accuracy (r), use the formula,

$$r = \frac{\sum_{i=1}^{k} a_i}{n}$$

where n denotes the number of instances in the dataset, a_i is the number of objects with class labels that dominates.

The data sets in this framework consist of three gene expression profiles. They include brain tumor, lung cancer, and prostate tumor samples. The microarray data was obtained by the oligonucleotide technique. The data format is shown in Table 1 and comprises of the data set name, the number of samples, categories, samples, genes, selected genes, and the diagnostic task. The data sets in this study consisted of many gene expression profiles, which were downloaded from http://www.gems-system.org.

Table 1: Format of Gene Expression Classification Data

Data Set Name	Samples	Categories	Genes	Genes Selected			Percentage of Gene Selected IG/KNN	Percentage of Gene Selected IG GA/KNN	Percentage of Gene Selected IG NRGA /KNN	Diagnostic Task
				IG/KNN	IG-GA/KNN	IG NRGA /KNN				
Brain Tumor	90	5	5920	1612	244	125	27.2%	4.1%	2.1%	5 human brain tumor types
Lung cancer	203	5	12600	9561	2101	1215	75.9%	16.7%	12.7%	4 lung cancer types and normal tissues
Prostate Tumour	102	2	10509	2016	3153	274	19.2%	3.3%	2.6%	Prostate tumor and normal tissues

They include tumor, brain tumor, leukemia, lung cancer, and prostate tumor samples. The microarray data was obtained by the oligonucleotide technique, except in the case of Small, round blue cell tumors (SRBCT), which was obtained by continuous image analysis.

The data samples for prostate and brain tumor are 10509 and 5920 but lesser than lung cancer which is 12600 genes. The categories for lung and brain are 5 greater than prostate which is 2. The IG/KNN method selects the gene for given dataset. The most important composition is the gene selection where lung cancer holds 9561, greater than prostate, 2016. The brain tumor is the least 1612. The selection is found to resemble more likely to given samples. The IG-GA/KNN method is incorporated that improves selection by 244 for brain, 2102 for lung and 3153 for prostate. Hence IG-GA/KNN makes good gene selection. The NRGA/KNN is the proposed method that selects 125 for brain, 1215 for lung and 274 for prostate. It has the best gene selection better than IG/KNN and IG-GA/KNN (Figure 1 to Figure 6).

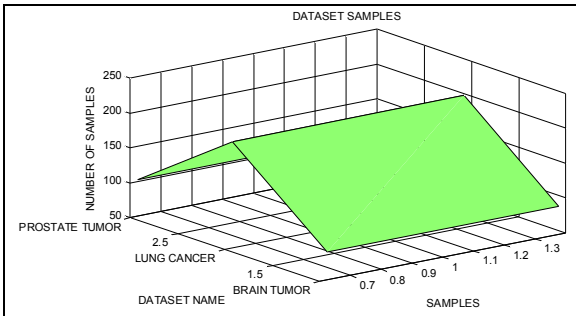

Figure 1: Dataset Samples

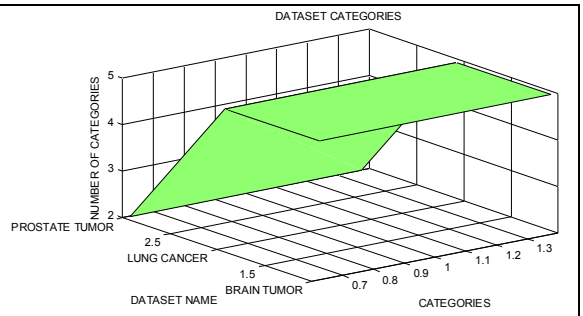

Figure 2: Dataset Categories

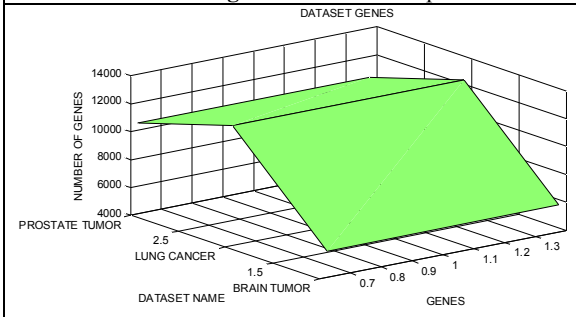

Figure 3: Dataset Genes

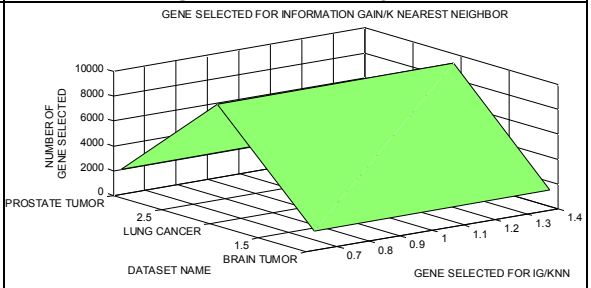

Figure 4: Gene Selected for IG/K Nearest Neighbor

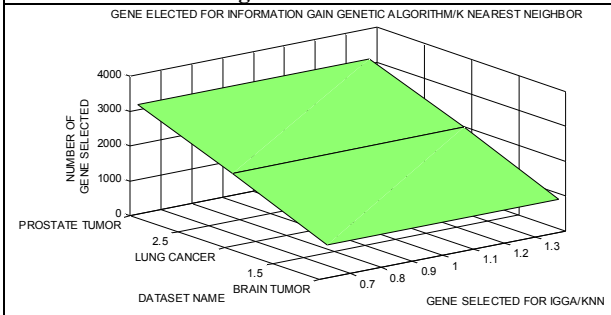

Figure 5: Gene Selected for IG-GA/K Nearest Neighbor

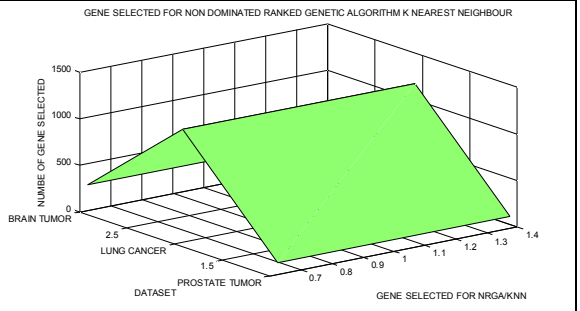

Figure 6: Gene Selected for NRGA/K Nearest Neighbor

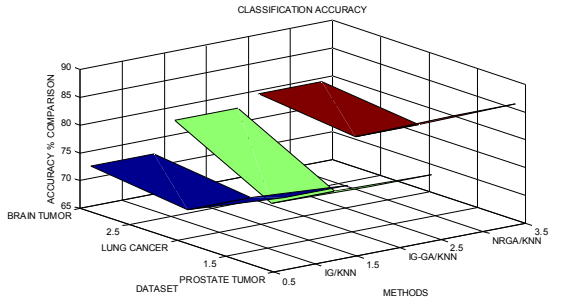

Figure 7(a): Accuracy % for 10 iterations

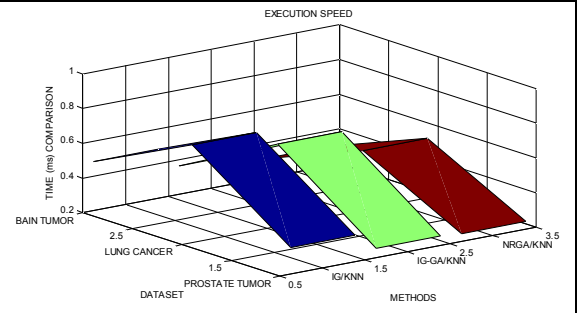

Figure 7(b): Execution Speed for 10 iterations

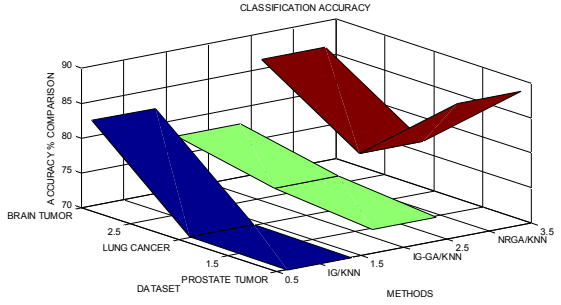

Figure 7(c): Accuracy % for 100 iterations

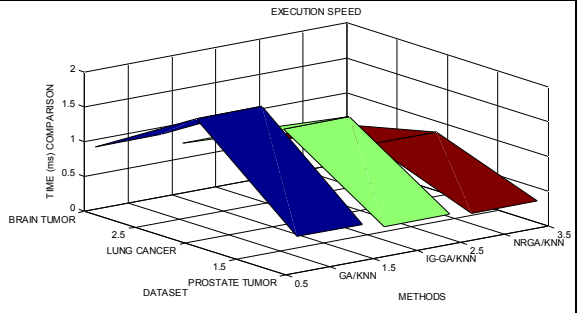

Figure 7(d): Execution Speed for 100 iterations

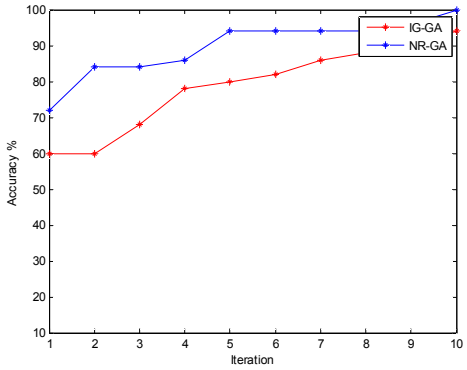

Figure 8: Feature Selection Accuracy for Brain tumor

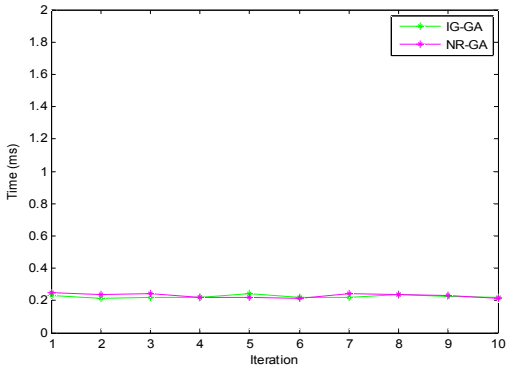

Figure 9: Execution Time for brain tumor

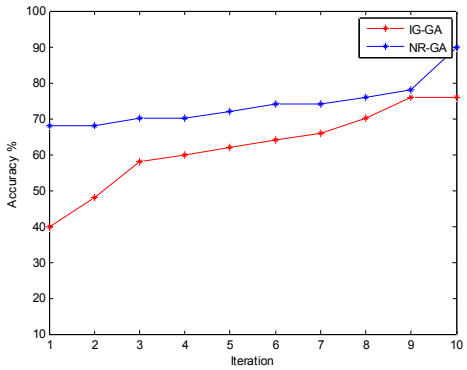

Figure 10: Feature Selection Accuracy for lung cancer

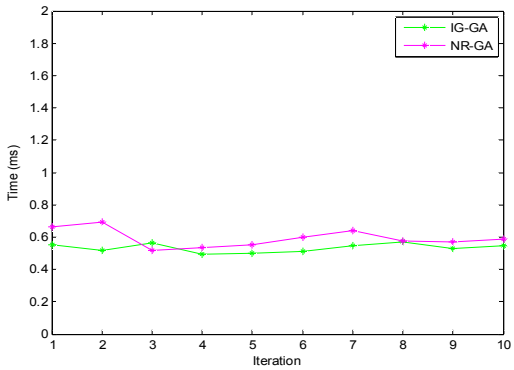

Figure 11: Execution Time for lung cancer

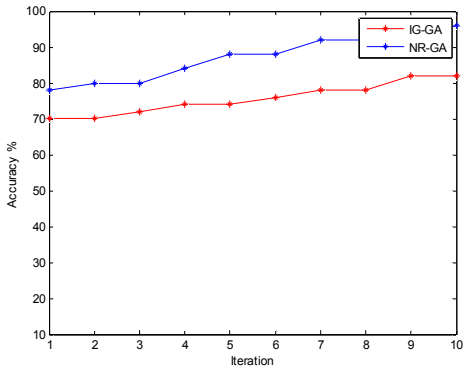

Figure 12: Feature Selection Accuracy for Prostate Tumor

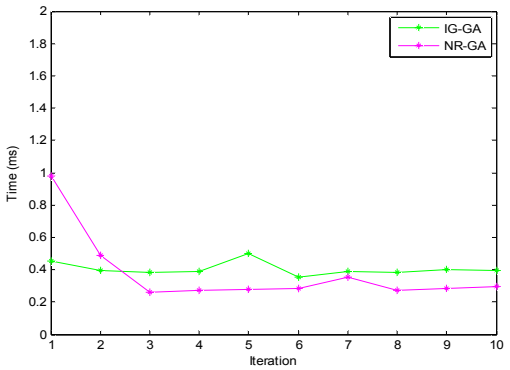

Figure 13: Execution Time for prostate tumor datasets

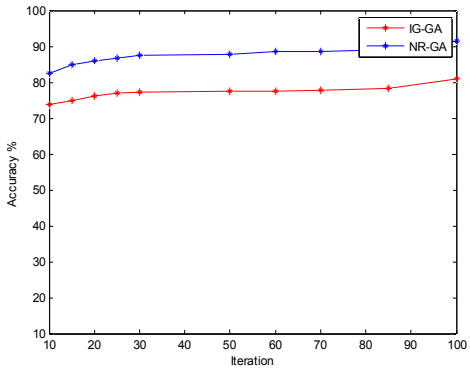

Figure 14: Feature Selection Accuracy for Brain tumor

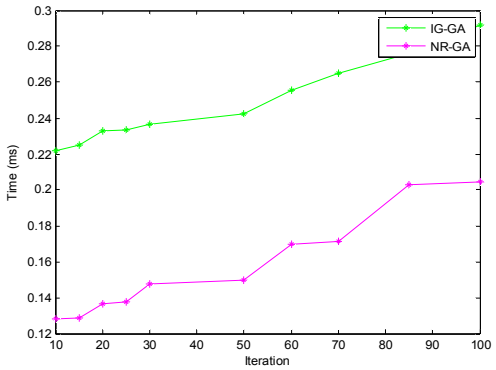

Figure 15: Execution Time for brain tumor

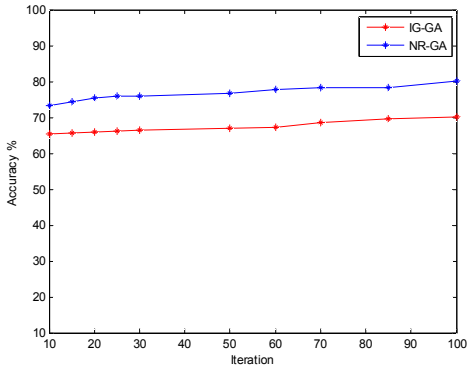

Figure 16: Feature Selection Accuracy for lung cancer

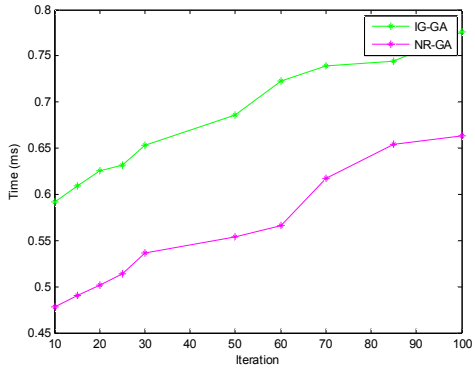

Figure 17: Execution Time for lung cancer

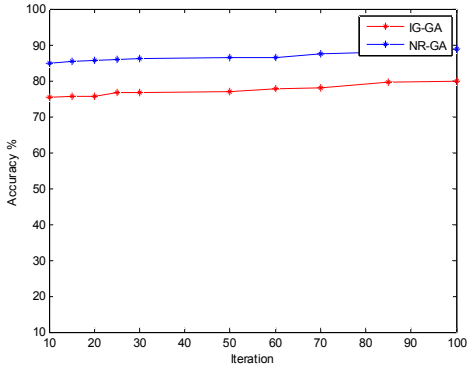

Figure 18: Feature Selection Accuracy for Prostate Tumor

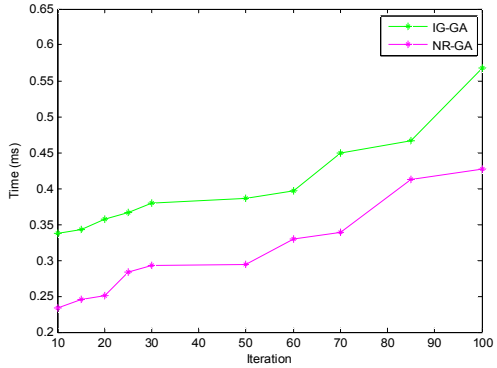

Figure 19: Execution Time for prostate tumor

Table 2: Accuracy of classification for gene expression data (Accuracy Comparison over NRGA)

Data Sets for 10 iterations	Accuracy %			Time (ms)		
	IG KNN	IG-GA KNN	NRGA KNN	IG KNN	IG-GA KNN	NRGA KNN
Brain Tumor	78	77	**86.8**	0.354	0.258	**0.248**
Lung cancer	70.15	68.4	**77.4**	0.762	0.673	**0.543**
Prostate Tumor	72.22	77.6	**79.4**	0.482	0.363	**0.313**
Data Sets for 100 iterations	Accuracy %			Time (ms)		
	IG KNN	IG-GA KNN	NRGA KNN	IG KNN	IG-GA KNN	NRGA KNN
Brain Tumor	70	73.4	**89.1**	0.524	0.434	**0.392**
Lung cancer	70.15	74.8	**77.4**	1.762	1.377	**0.925**
Prostate Tumor	82.22	77.6	**86.3**	0.896	0.715	**0.576**

Table 2 shows that the classification accuracy and NRGA execution speed for the gene expression data. Three datasets like brain tumor, lung cancer and prostrate tumor is considered here. Finally NRGA/KNN is proposed to improve the accuracy with best feature selection. From the table 2 it can be clearly seen that the proposed NRGA-KNN for 10 iterations shows 86.8% for brain, 77.4% for lung and 79.4% for prostate. To 100 iterations NRGA/KNN results are 89.1% for brain, 77.4% for lung and 86.3% for prostate. Hence the clarification accuracy shown in table 2 for IG/KNN and IG-GA/KNN are lesser when compared with NRGA/KNN.

Brain tumor, Lung cancer and prostate: It can be seen that the proposed NRGA algorithm for 10 iterations which is 2% to 9% more accurate than IG-GA/KNN in selecting the features than the existing IG-GA. For 100 iterations it is 3% to 16%.

Hence when the iteration increases accuracy also increases. It has been shown in figure 8 to figure 13 only for 10 iterations and figure 14 to figure 19 for 100 iterations. The KNN method is served as an evaluator of the NRGA algorithm. For more than 10 there exists a controversy with accuracy and time by applying NRGA without KNN. It can be seen that the proposed NRGA algorithm executes in 0.248ms for brain, 0.543ms for lung and 0.313 for prostate (10 iterations). For 100 iterations it is 0.392ms for brain, 0.925ms for lung and 0.576ms for prostate. The other two existing methods: IG/KNN and IG-GA/KNN process with more time than NRGA/KNN. NRGA algorithm sustains results with 100 iterations. The experimental results show that the proposed NRGA gives better accuracy when compared with the other existing methods like IG/KNN and IG-GA/KNN. By increasing the number of iterations from 10 to 100, the performance of the proposed NRGA algorithm is tested for good results.

4. Conclusion

The proposed work mainly concentrates on identifying the genes that provide relevant information and thus benefit the classification process. A clustering algorithm which is a hybrid model of information gain Non-Dominated Ranked genetic algorithm is presented for feature selection in microarray data sets. NRGA algorithm is used to perform feature selection based on clustering technique. The K-NN method with LOOCV served as an evaluator of the NRGA fitness functions. Experimental results showed that NRGA simplified feature selection by clustering effectively reduced the total number of features needed, and obtained a higher accuracy compared to other feature selection methods. The accuracy obtained by the proposed method had the highest accuracy, and was comparable with other techniques. IG can serve as a pre-processing tool to help optimize the feature selection process, since it either increases the accuracy, reduces the number of necessary features for classification, or both. The proposed NRGA method could conceivably be applied to problems in other areas in the future.

References

1. Wolfgang Huber, Anja von Heydebreck and Martin Vingron, “Analysis of microarray gene expression data”, Max-Planck-Institute for Molecular Genetics 14195 Berlin April 2, 2003.
2. M. Kathleen Kerr and Gary A. Churchill. Statistical design and the analysis of gene expression microarray data. *Genet. Res.*, 77:123–128, 2001.
3. G.A. Churchill. Fundamentals of experimental design for cDNA microarrays. *Nature Genetics*, 32 Suppl. 2:490–495, 2002.
4. Yee Hwa Yang and Terence P. Speed. Design issues for cDNA microarray experiments. *Nat. Rev. Gen.*, 3:579–588, 2002.
5. M. L. Raymer, W. F. Punch, E. D. Goodman, L. A. Kuhn and A. K. Jain, “Dimensionality reduction using genetic algorithms,” *IEEE Trans. Evol. Comput.*, 4: 164-171, 2000.
6. K. Crammer and Y. Singer, “On the learn ability and design of output codes for multiclass problems,” *Proc. of the Thirteenth Annual Conf. on Computational Learning Theory*, 35-46, 2000.
7. B. Yu and B. Yuan, “A more efficient branch and bound algorithm for feature selection,” *Pattern Recognit.*, 26: 883-889, 1993.
8. P. Pudil, J. Novovicova and J. Kittler, “Floating search methods in feature selection,” *Pattern Recognit. Lett.*, 15: 1119-1125, 1994.
9. R. Battiti, “Using mutual information for selecting features in supervised neural net learning,” *IEEE Trans. Neural Netw.*, 5: 537-550, 1994.
10. H. Zhang and G. Sun, “Feature selection using tabu search method,” *Pattern Recognit.*, 35: 701-711, 2002.
11. B. Everitt, “Cluster analysis” 1st ed. Heinemann, London, 1980.
12. R. Xu and D. Wunsch, “Survey of clustering Algorithms”, IEEE Trans on Neural Networks. Vol. 16, no. 3, pp.645-678, 2005.
13. J. McQueen, “Some methods for classification and analysis of multivariate observations,” in Proceedings of the Fifth Berkeley Symp. Math. Statistics and Probability, vol. 1, 1967, pp. 281–297.
14. P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E. Lander, and T. Golub, “Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation,” in Proceedings of National Academy of Sciences, vol. 96(6), USA, 1999, pp. 2907–2912.
15. J. Dopazo and J. Carazo, “Phylogenetic reconstruction using an unsupervised neural network that adopts the topology of a phylogenetic tree,” J Mol Eval, vol. 44, pp. 226–233, 1997.
16. S. Tomida, T. Hanai, H. Honda, and T. Kobayashi, “Analysis of expression profile using fuzzy adaptive resonance theory,” Bioinformatics, vol. 18(8), pp. 1073–83, 2002.
17. M. T. Martin-Valdivia, M. C. Diaz-Galiano, A. Montejo-Raez and L. A. Urena-Lopez, “Using

information gain to improve multi-modal information retrieval systems," *Inf. Process. Manage.*, 44: 1146-1158, 2008.
18. R. Mukras, N. Wiratunga and R. Lothian, S. Chakraborti and D.Harper, "Information gain feature selection for ordinal text classification using probability re-distribution," *Proc. of IJCAI Textlink Workshop*, 2007.
19. A. M. Turing, "Computing machinery and intelligence," *Mind*, 59: 433-460, 1950.
20. J. Holland, *Adaptation in Nature and Artificial Systems*, Cambridge, MA: MIT Press, 1992.
21. D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Reading, MA: Addison-Wesley Professional, 1989.
22. E. S. H. Hou, N. Ansari and H. Ren, "A genetic algorithm for multiprocessor scheduling," *IEEE Trans. Parallel Distrib. Syst.*, 5: 113-120, 1994.
23. L. Davis, "Hybrid genetic algorithms for machine learning," *Mach. Learn.*, 117: 9/1-9/3, 1990.
24. H. Vafaie and K. De Jong, "Genetic algorithms as a tool for feature selection in machine learning," *Proc. 4th Int. Conf. on Tools with Artificial Intelligence*, 200-203, 1992.
25. K. Deb, A. Pratap, S. Agrawal and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II," *IEEE Trans. Evol. Comput.*, 6: 182-197, 2002.
26. A. H. F. Dias and J. A. de Vasconcelos, "Multiobjective genetic algorithms applied to solve optimization problems," *IEEE Trans. Magn.*, 38: 1133-1136, 2002.
27. S. Kim and B. T. Zhang, "Evolutionary learning of Web-document structure for information retrieval," *Proc. of the 2001 Congress on Evolutionary Computation*, 2: 1253-1260, 2001.
28. C. F. Tsai, C. W. Tsai, C. P. Chen and F. C. Lin, "A multiple-searching approach to genetic algorithms for solving traveling salesman problem," *Proc. of Joint Conf. on Information Sciences*, 3: 362-366, 2002.
29. T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, IT-13: 21-27, 1967.
30. E. Fix and J. L. Hodges, Jr., "Discriminatory analysis – Nonparametric discrimination: consistency properties," *Project 21-49-004, Report Number 4, US Air Force School of Aviation Medicine, Randolph Field*, 261-279, 1951.
31. E. Frank, M. Hall, L. Trigg, G. Holmes and I. H. Witten, "Data mining in bioinformatics using Weka," *Bioinformatics*, 20: 2479-2481, 2004.
32. Fogel, D.: Evolutionary Computing, Toward a New Philosophy of Machine Intelligence. IEEE Press, Piscataway, NJ (1995).
33. Cheng-Huei Yang, Li-Yeh Chuang, Cheng-Hong Yang, "IG-GA: A Hybrid Filter/Wrapper Method for Feature Selection of Microarray Data," *Journal of Medical and Biological Engineering,* 30(1): 23-28, 2009

5/12/2013