A Genetic Algorithm - Support Vector Machine Approach to DNA Microarrays Supervised Learning

Nicolae Teodor MELITA[1], Stefan HOLBAN[2]
[1]"Politehnica" University of Timisoara, Faculty of Automation and Computers,
Bd. V. Parvan 2, RO-300223 Timisoara, Romania,
nt_melita@yahoo.com
[2]"Politehnica" University of Timisoara, Faculty of Automation and Computers,
Bd. V. Parvan 2, RO-300223 Timisoara, Romania,
stefan@cs.utt.ro

***Abstract*—We address the problem of collecting and analyzing vast amount of information in medicine and biology, in the light of the revolutionary technological evolution in the last decades. Currently, the methods of achieving information overcome our capacity to sort and process that information. However, we use the methods of machine learning to sort and analyze this information. In this comprehensive review we describe an experiment of analyzing DNA microarrays using Support Vector Machines. We study how the SVM performs in classifying three instances of the same dataset. We classify the brute dataset, a t-test based filtered dataset, and a dataset with features selected by a Genetic Algorithm.**

***Index Terms*—DNA Microarrays, Feature Selection, Genetic Algorithm, Supervised Learning, Support Vector Machine (SVM)**

I. INTRODUCTION

In last decades, the extraordinary evolution of technology has changed completely the way research is designed and practiced in all the fields of science. Medicine and biology obtained methods of research which provide an enormous amount of data. The complexity of the research process became so overwhelming that it is almost impossible these days to develop a breakthrough research in medicine without the collaboration of scientists from completely different fields. The final goal is a better healthcare delivery and a whole army of interdisciplinary teams work to achieve this goal.

Information technology created a revolution in medicine in all areas, from diagnosing techniques to high level surgery procedures. In the last decade we witnessed a spectacular revolution in genetics. We expect that future evolution will provide us with genetic diagnostic methods and treatments capable to heal most of the worst prognosticate diseases that concern us today.

A relatively new research technology is available for analyzing genes fixed on specially designed chips, called microarrays. There are two main types of microarrays, suitable for different tasks. First type, the Multiprobe microarrays are arrays of many probes on a single chip. They are represented by DNA microarrays and the antibody microarrays. The second type of microarrays is represented by multisample arrays, represented by protein extract arrays and tissue arrays.

The DNA microarrays (Fig. 1) are glass or plastic chips which immobilize thousands to hundred thousands samples of DNA fragments, cDNA or oligonucleotides, depending of chip construction technology.

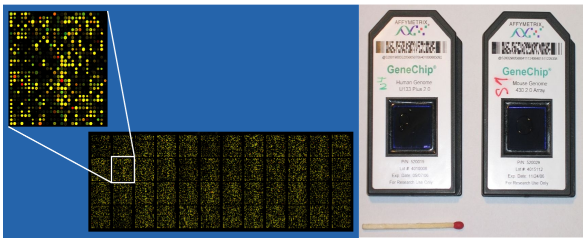

a) b)

Figure 1. a) An example of DNA microarray, Stanford technology; b) An example of Affimetrix chip (the source of the image is wikipedia.org, a public domain).

We are interested to compare groups of patients that have a specific disease with patients that do not have the disease in terms of which genes are expressed in a group compared with the genes expressed by the other group and to establish a causal relationship between a group of expressed genes and a disease. This approach is especially useful for early diagnosis, prognostic and treatment of cancer. The stage when the cancer disease is detected is, at this point, the most important predictor for the patient's evolution. Microarray technology sets the basis for very efficient methods for screening and diagnosing cancer in an early stage of development. If we will be able to know exactly which genes are involved in a specific type of cancer, we would be able to detect the exposed patients in time to provide effective treatments. The method could set the basis for developing new treatments for various types of cancers. Using better techniques to deal with cancer we would be able to win against one of the most important causes of death in our days.

II. PROBLEM STATEMENT

The problem we are addressing here is how we process this information in order to achieve knowledge. Nowadays, the methods of machine learning and statistics are the key factor of the research. As the number of the genes on an array grows every year also the complexity of the process increases. The main target of microarrays builders is to be able to put the whole human genome on a single chip. A big amount of probes will require very specialized and statistically significant processing methods.

Very important steps are made in this direction. Machine learning algorithms are implemented in very complex software packages. They include implementations of algorithms specific to both unsupervised and supervised analysis methods, as well as statistical methods to test the significance and very intuitive graphical methods to represent the outcome. Some of these projects are: Matlab, Weka, Orange and The R Project..

We will use a Support Vector Machine to address a supervised Machine Learning task and we will study how to get better results on a microarray dataset. We will use one microarray dataset processed in three different ways. First we learn the dataset with all its features, in our case genes expression levels. We will use a big part of the dataset as training set and a smaller part of it as testing set. The classes are represented equal in both the training and the testing set. In the next step, we will filter the dataset based on t-statistic and we will keep just a significant part of the features. We will learn the filtered dataset with the SVM, using the same strategy of splitting the filtered dataset in training and testing subsets. Finally, we will select a small number of features (genes) from the original dataset using a Genetic Algorithm. We will train and test a SVM on the modified dataset, using the same splitting strategy in testing and training subsets. The choose to split the dataset in training and testing sets before we ran the GA, in order to make sure that the features selection algorithm does not benefit from any information from the testing set. We will compare the results to see which method is better.

III. SURVEY OF THE LITERATURE

Because of the fact that we are dealing with a relatively new interdisciplinary field, the literature is devised between all the research fields involved. We are interested in a better understanding of our dataset, so we want to know about the methods of biotechnology for creating microarrays and providing data to be analyzed (Causton, Quackenbush & Brazma, 2003 [1]). Other approaches focus on the bioinformatics' point of view on methods of collecting and analyzing data (Dov Stekel, 2003 [2]). The books that focus on the specific machine learning methods help in developing an image of how the algorithms work, their strong and weak points (Ressom, 2007 [3]; Duda, P. E. Hart and D. G. Stork, 2001 [4]; I. Witten and E. Frank, 2005 [5]).

A very helpful set of books are focused on using the specific software tools that we need in microarray analysis with emphasis on specific features (Venables & Ripley, 2000 [6]; D. G. Stork and E. Yom-Tov, 2004 [7]). These books are designed to introduce the researchers in using these software packages fast and effective.

IV. METHOD

We selected MATLAB to facilitate in performing the experiments. We selected this specific tool for many reasons. MATLAB provides various toolboxes specialized in performing different specialized tasks of machine learning and statistical analysis. All toolboxes come with very good and integrated documentation. We verified some results using the R Project. For the unsupervised learning we used MATLAB and the software programs Cluster and TreeView written by Michael Eisen.

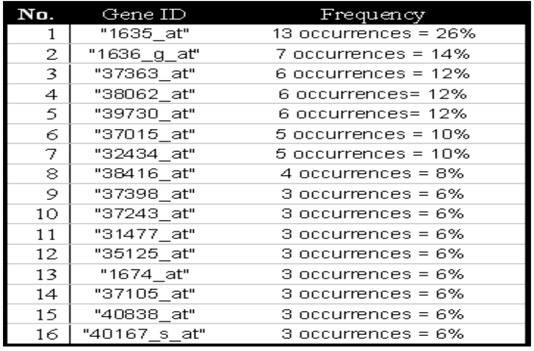
The method we will present is a supervised learning task, applied to learn information from an Affymetrix Microarray experiment. We tried to predict the molecular biology class of the patient based on the levels the expressed genes. For the filtering techniques, we adopted a t-test based method presented in the article "Bioinformatics and Computational Biology Solutions using R and Bioconductor" by Robert Gentleman, Vince Carey, Wolfgang Huber, Rafael A. Irizarry, Sandrine Dudoit [8]. For the features selection using GA we used the Genetic Algorithm and Direct Search Toolbox in MATLAB with the fitness function adjusted for searching a discrete space, proposed by Sam Roberts in "Using Genetic Algorithms to Select a Subset of Predictive Variables from a High-Dimensional Microarray Dataset" [9]. For estimating the error rate of the genetic algorithm we used 10-fold cross validation. We used the SVM implementation of MATLAB and we tested four SVM classifiers, based on different kernel functions (linear, quadratic, RBF, and MLP).

The dataset we used is called ALL (Acute Lymphocytic Leukemia). ALL is provided by the Ritz Laboratory and the current version was released in 2004. . It contains 12650 genes and 128 samples, 128 Affymetrix microarrays. The 128 samples included in the dataset represent patients with B-cell Acute Lymphocytic Leukemia, and T-cell Acute Lymphocytic Leukemia. The dataset is available for download on the Bioconductor's webpage: http://www.bioconductor.org/packages/1.8/data/experiment/bin/windows/contrib/2.3/ALL_1.2.1.zip. We transformed the dataset choosing just the cases that have B-cell Acute Lymphocytic Leukemia and the negative molecular biology cases. We started with three versions of the dataset. We used the ALL dataset with full features, normalized and keeping just the samples that presented as negative or B-cell Acute Lymphocytic Leukemia, a dataset with 79 samples and 12650 features. In another version of the dataset we filtered the dataset using t-statistic considering just the most 1000 differentially expressed genes, and again, keeping just the negative or B-cell Acute Lymphocytic Leukemia cases, a dataset with 79 samples and 1000 features. The most 1000 differentially used genes were processed with the limma package; an open-source package specialized in analyzing microarrays based on empirical Baesyan methods. Finally, we used the genetic algorithm to select the best 10 features, so we had a dataset with 79 samples and 10 features.

We used the first 59 samples as training set, and the other 20 as test set. The negative cases and the B-cell Acute Lymphocytic Leukemia cases were equally represented in both the training set and the testing set. The three datasets were learned using four SVMs, based of four different kernel functions. We evaluated the performance of the algorithms in terms of accuracy, sensitivity and specificity.

After we ran the Genetic Algorithm with 50 repetitions, over the training set with 59 samples, 16 genes (Table 1) appeared with a frequency more than 6% in the selected features subsets.

TABLE 1 – THE MOST FREQUENT GENES IN THE GA OUTPUT



No.	Gene ID	Frequency
1	"1635_at"	13 occurrences = 26%
2	"1636_g_at"	7 occurrences = 14%
3	"37363_at"	6 occurrences = 12%
4	"38062_at"	6 occurrences= 12%
5	"39730_at"	6 occurrences= 12%
6	"37015_at"	5 occurrences = 10%
7	"32434_at"	5 occurrences = 10%
8	"38416_at"	4 occurrences = 8%
9	"37398_at"	3 occurrences = 6%
10	"37243_at"	3 occurrences = 6%
11	"31477_at"	3 occurrences = 6%
12	"35125_at"	3 occurrences = 6%
13	"1674_at"	3 occurrences = 6%
14	"37105_at"	3 occurrences = 6%
15	"40838_at"	3 occurrences = 6%
16	"40167_s_at"	3 occurrences = 6%

We created a third dataset consisting of the 16 selected features and 79 samples.

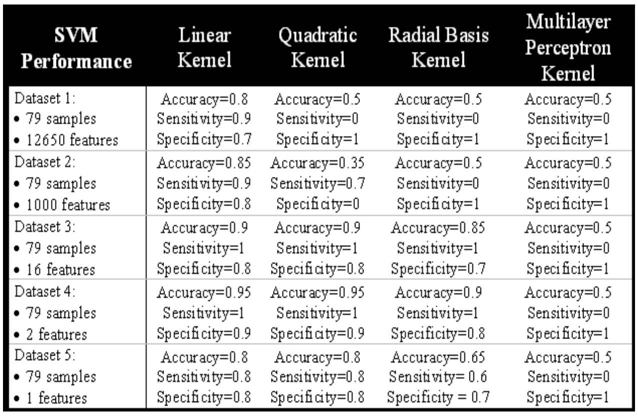
First, we analyzed the five datasets with unsupervised learning methods. We analyzed each dataset with unsupervised methods (Hierarchical Clustering based on Correlation for arrays and/or genes and K-means with 2 centroids for grouping arrays). We selected just two centers for K-means because we clustered based on samples and we wanted to study how similar the samples are. Some results of the unsupervised learning methods are presented in the Appendix A (Fig. 2-7). After we analyzed the unsupervised learning techniques results we decided to create another two datasets, one with the most frequent two features and the second, with just the most frequent feature.

In a second step, we trained 20 SVMs as it follows. We trained with each of the five datasets four SVMs, with different kernel functions. Then we tested the performance of each SVM over the corresponding training set.

V. RESULTS

We presented the five testing sets consisting in 20 samples each, to their corresponding SVMs previously trained with the 59 samples matching training sets. We used five different training sets, each with a different number of features to train the SVMs. We trained four SVMs, with each training set, each SVM with different Kernel Function (linear, quadratic, Gaussian radial basis, and MLP). The results we got are presented in the (Table 2). We notice that the supervised learning results were in agreement with the beliefs we had after analyzing the unsupervised learning results, and the creation of the fourth dataset was indeed very useful.

TABLE 2 – THE SVM PERFORMANCE ON VARIOUS DATASETS



SVM Performance	Linear Kernel	Quadratic Kernel	Radial Basis Kernel	Multilayer Perceptron Kernel

Dataset 1:
• 79 samples
• 12650 features
	Accuracy=0.8
	Sensitivity=0.9
	Specificity=0.7
		Accuracy=0.5
		Sensitivity=0
		Specificity=1
			Accuracy=0.5
			Sensitivity=0
			Specificity=1
				Accuracy=0.5
				Sensitivity=0
				Specificity=1

Dataset 2:
• 79 samples
• 1000 features
	Accuracy=0.85
	Sensitivity=0.9
	Specificity=0.8
		Accuracy=0.35
		Sensitivity=0.7
		Specificity=0
			Accuracy=0.5
			Sensitivity=0
			Specificity=1
				Accuracy=0.5
				Sensitivity=0
				Specificity=1

Dataset 3:
• 79 samples
• 16 features
	Accuracy=0.9
	Sensitivity=1
	Specificity=0.8
		Accuracy=0.9
		Sensitivity=1
		Specificity=0.8
			Accuracy=0.85
			Sensitivity=1
			Specificity=0.7
				Accuracy=0.5
				Sensitivity=0
				Specificity=1

Dataset 4:
• 79 samples
• 2 features
	Accuracy=0.95
	Sensitivity=1
	Specificity=0.9
		Accuracy=0.95
		Sensitivity=1
		Specificity=0.9
			Accuracy=0.9
			Sensitivity=1
			Specificity=0.8
				Accuracy=0.5
				Sensitivity=0
				Specificity=1

Dataset 5:
• 79 samples
• 1 features
	Accuracy=0.8
	Sensitivity=0.8
	Specificity=0.8
		Accuracy=0.8
		Sensitivity=0.8
		Specificity=0.8
			Accuracy=0.65
			Sensitivity= 0.6
			Specificity = 0.7
				Accuracy=0.5
				Sensitivity=0
				Specificity=1

VI. CONCLUSION

1. The SVMs were able to predict the correct molecular biology of the cancer better on the dataset with just two features (Accuracy=0.95, Sensitivity=1, Specificity=0.9). We got the same results from the linear kernel based and quadratic kernel based SVMs over the fourth dataset. These results encouraged us to believe that these features are really significant in predicting the type of the cancer.
2. After a research on internet we found these two features' significance. The 1635_at represents the Proto-oncogene tyrosine-protein kinase ABL1. The 1636_g_at also represents the Proto-oncogene tyrosine-protein kinase ABL1. The ABL1 is v-abl Abelson murine leukemia viral oncogene homolog 1, which is a known oncogene, strong related with B-cell Acute Lymphocytic Leukemia. http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene&cmd=etrieve&dopt=raphics&list_uids=25
3. The fact that our results are biologically proven makes us believe that the feature selection method with a GA is effective. We can believe that our fourth dataset, with just two selected features is the best for predicting the type of cancer. More general, we see a case where features selection provides a better classification performance than the original dataset.

APPENDIX A

Unsupervised Learning Results (N=negative, P=positive case):

Dataset 1 (79 samples, 12650 features)

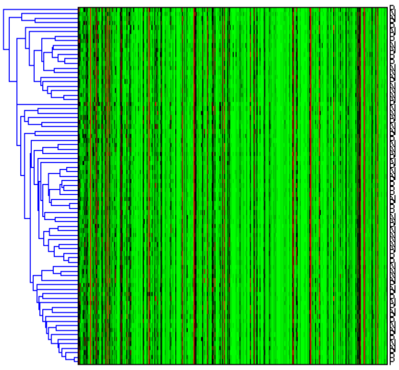


Figure 2. Hierarchical Clustering for the Arrays - Dendrogram and Heatmap.

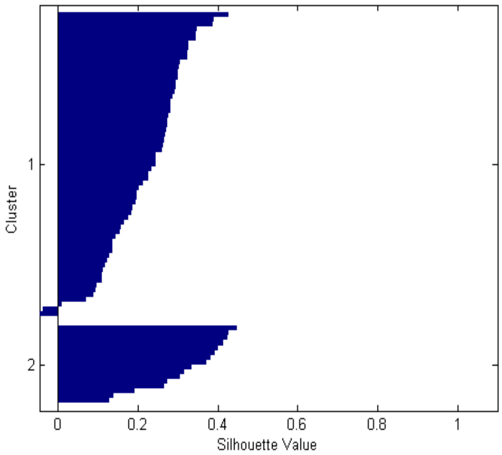


Figure 3. K-Means based on Correlation with 2 centroids for the Arrays.

Dataset 2 (79 samples, 1000 features)

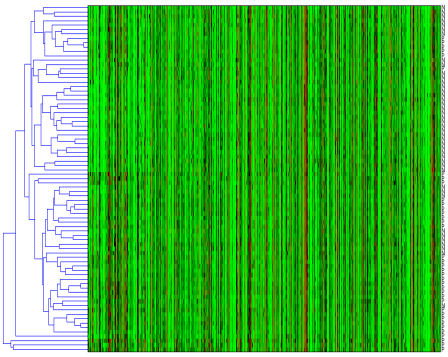


Figure 4. Hierarchical Clustering for the Arrays - Dendrogram and Heatmap.

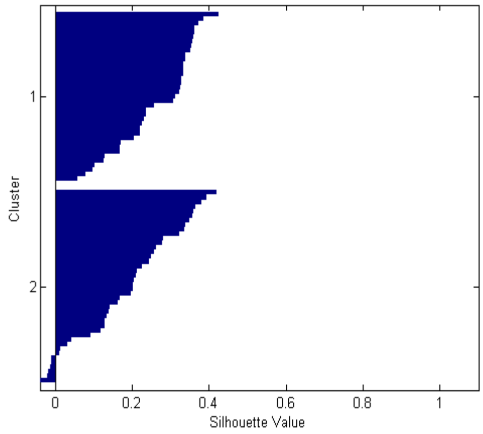


Figure 5. K-Means based on Correlation with 2 centroids for the Arrays.

Dataset 3 (79 samples, 16 features)

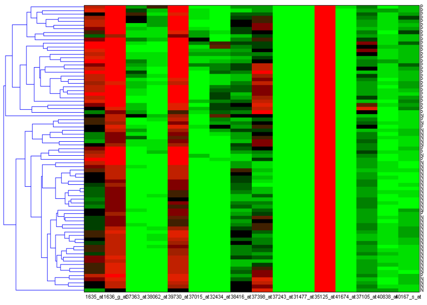


Figure 6. Hierarchical Clustering for the Arrays - Dendrogram and Heatmap.

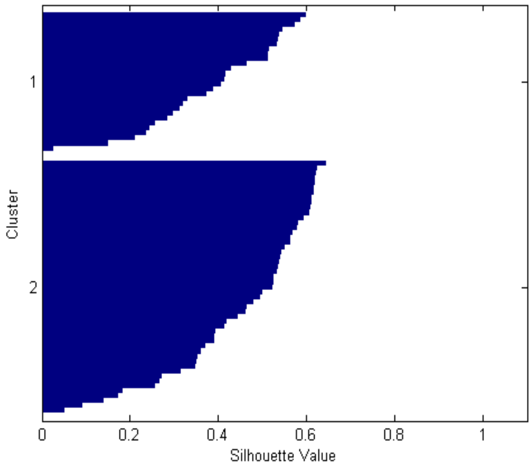


Figure 7. K-Means based on Correlation with 2 centroids for the Arrays.

REFERENCES

[1] Helen Causton, John Quackenbush, Alvis Brazma. Microarray Gene Expression Data Analysis: A Beginner's Guide, Blackwell Publishing Professional, 2003.

[2] Dov Stekel, Microarray Bioinformatics Cambridge University Press, 2003.

[3] H. Ressom, Lecture Notes, Georgetown University, 2007.

[4] R. O. Duda, P. E. Hart and D. G. Stork, Pattern Classification, Second Edition, Wiley, 2001.

[5] I. Witten and E. Frank, Data Mining (2nd Ed.), Morgan Kaufmann, 2005.

[6] W. N. Venables, D. M. Smith & the R Development Core Team, An Introduction to R, 2006.

[7] D. G. Stork and E. Yom-Tov, Computer Manual in MATLAB to Accompany Pattern Classification, Second Edition, Wiley, 2004.

[8] Robert Gentleman, Vince Carey, Wolfgang Huber, Rafael A. Irizarry, Sandrine Dudoit, Bioinformatics and Computational Biology Solutions using R and Bioconductor, 2005.

[9] Sam Roberts, Using Genetic Algorithms to Select a Subset of Predictive Variables from a High-Dimensional Microarray Dataset, 2005.

[10] William N. Venables and Brian D. Ripley, Modern Applied Statistics with S. Fourth Edition. Springer, New York, 2002.

[11] William N. Venables and Brian D. Ripley, S Programming. Springer, New York, 2000.

[12] MATLAB Digest.