Online journal ISSN - 1676-5680
GMR Genetics and Molecular Research
EVOLUTION AND TECHNOLOGY

Gene selection based on multi-class support vector machines and genetic algorithms

Bruno Feres de Souza and André Ponce de Leon F. de Carvalho

Instituto de Ciências Matemáticas e de Computação,
Universidade de São Paulo, Av. Trabalhador São-Carlense, 400,
13560-970 São Carlos, SP, Brasil
Corresponding author: B.F. de Souza
E-mail: bferes@icmc.usp.br

Genet. Mol. Res. 4 (3): 599-607 (2005)
Received May 20, 2005
Accepted July 8, 2005
Published September 30, 2005

ABSTRACT. Microarrays are a new technology that allows biologists to better understand the interactions between diverse pathologic state at the gene level. However, the amount of data generated by these tools becomes problematic, even though data are supposed to be automatically analyzed (e.g., for diagnostic purposes). The issue becomes more complex when the expression data involve multiple states. We present a novel approach to the gene selection problem in multi-class gene expression-based cancer classification, which combines support vector machines and genetic algorithms. This new method is able to select small subsets and still improve the classification accuracy.

Key words: Multi-class SVM, Gene expression, Feature selection, Genetic algorithms, Microarrays

INTRODUCTION

Microarrays use hybridization-based methodology, which allows monitoring of the expression levels of thousands of genes simultaneously (Schena and Knudsen, 2004). This enables the measurement of the levels of mRNA molecules inside a cell and, consequently, the proteins being produced. Therefore, the role of the genes of a cell at a given moment can be better understood by assessing their expression levels. In this context, the comparison between gene expression patterns through the measurement of the levels of mRNA in healthy versus unhealthy cells can supply important information about pathological states, as well as information that can lead to earlier diagnosis and more efficient treatment.

In order to make this feasible, it is desirable to deal with the genes involved in the development of the particular health condition. However, in most cases, which genes are involved in the process is still an open issue. Thus, monitoring of the largest possible number of genes is necessary. In this sense, microarrays are valuable medical tools.

Amongst the most trivial applications of microarrays, the classification of tissue samples is an essential step for the assessment of severe diseases. This classification can be carried out by machine learning algorithms, such as support vector machines (SVMs) (Cristianini and Shawe-Taylor, 2000). SVMs have exhibited good performance in a wide range of applications, especially in bioinformatics. Their principal advantage is their good generalization capability. However, when dealing with gene expression data, there is usually a disproportionate ratio between the high number of features (genes) and the low number of analyzed samples (tissues), which can make the analysis of the results difficult and damage classification.

The dimensionality problem is expected to become more serious when dealing with multi-class domains, since these are intrinsically more difficult than the binary ones, because the classification algorithm has to learn to construct a greater number of separation boundaries or relations (Rifkin et al., 2003). In a comparative study on feature selection and multi-class classification problems, Li et al. (2004) suggested that working with small subsets of genes could ameliorate the difficulties of multi-class problems.

We present a novel way to perform gene selection using SVMs and genetic algorithms (GA) (Goldberg, 1989) for multi-class problems. It is called the multi GA-SVM method. Fröhlich et al. (2003) have already used the GA-SVM combination for this purpose. However, it was employed as just a single example of their general framework. We propose specific solutions, such as optimization of the SVM hyperparameters *C* independently for all classifiers, normalization of the samples to unit length, allowing the use of a correlation-based similarity measure, together with the SVM and the use of the generalized approximate cross-validation (Wahba, 1999) as a generalization estimator for SVM.

MATERIAL AND METHODS

Support vector machines

Binary support vector machines

Let $S = \{(x_1,y_1),\ldots, (x_n,y_n)\}$, where $x\in X\subset R^m$ and $y_i\in \{0,1,\ldots,c\}$, be a training dataset with *n* samples and *N* classes. Each x_i is an *m*-dimensional input vector, and each y_i corresponds to the class associated to x_i. In the microarray domain, x_i is the *ith* tissue sample, represented by a set of *m* genes, and y_i can be different types of cancer, for example.

The task of a classification algorithm is to learn a mapping of $x_i \rightarrow y_i$ using data from *S*. SVMs (Cristianini and Shawe-Taylor, 2000) handle this by constructing a hyperplane $<\mathrm{w} \cdot \mathrm{x}> + b = 0$, where $\mathrm{w} \in R^m$ represents the normal vector associated with the hyperplane and *b* is the bias, that maximally separates positive and negative training samples. The margin corresponds to the distance from the separating hyperplane to the closest samples of each class. It is obviously inversely proportional to ||w||. Thus, to have the maximal margin hyperplane one needs to minimize the Euclidean norm of vector w.

Formally, this goal can be translated into the following quadratic programming problem:

$$\begin{aligned} &\text{min: } \|\mathrm{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\ &\text{s.t.: } y_i\left(<\mathrm{w}\cdot\mathrm{x_i}> + b\right) \geq 1 - \xi_i,\ i = 1, ..., n \\ &\quad \xi_i > 0,\ i = 1, ..., n \end{aligned} \qquad \text{(Equation 1)}$$

where *C* is a term that controls the compromise between training error and classifier complexity and ξ_i are slack variables that permit to deal with non-linearly separable data.

Considering its dual formulation can more easily solve the problem presented in Equation 1:

$$\begin{aligned} &\text{max: } Q(\alpha) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j \mathrm{y}_i\mathrm{y}_j <\mathrm{x_i}\cdot\mathrm{x_j}> \\ &\text{s.t.: } 0 \leq \alpha_i \leq C, i = 1, ..., n \\ &\quad \sum_{i=1}^{n}\alpha_i\mathrm{y}_i = 0 \end{aligned} \qquad \text{(Equation 2)}$$

where α_i are named Lagrange Multipliers.

With the optimal values b^* and w^* (indirectly) obtained by solving Equation 2, the discriminant function can be expressed as:

$$f(\mathrm{x}) = \left(\sum_{i=1}^{n_{sv}} y_\mathrm{i}\alpha_\mathrm{i}^* <\mathrm{x}\cdot\mathrm{x_i}> + b^*\right) \qquad \text{(Equation 3)}$$

where n_{sv} is the number of samples associated with non-zero α_i in Equation 3, the so-called support vectors.

Due to the nature of the data (a few samples with very high dimensionality), the authors considered only linear discriminant functions, such as that expressed by Equation 3. However, other decision surfaces could be used in SVMs thanks to the concept of a kernel (Cristianini and Shawe-Taylor, 2000), which plays the role of dot product in a higher-dimensional space. As dot products, kernels can also be considered a type of similarity measure between the samples.

Recent studies (Alon et al., 1999) have shown that correlation-like similarity measures are better suited for gene expression data, as they focus on the relative shape of the sample profile and not on its magnitude. One such measure is the cosine of the angle between two tissue samples. Incorporating that domain knowledge into the SVM is straightforward. Assuming that the samples x and z are normalized to Euclidean length 1 (this normalization is used in the Results and Discussion section), the linear kernel used here is:

$$k\ (\mathrm{x,z}) = <\mathrm{x} \cdot \mathrm{z}> = \cos\ (\mathrm{x,z}) \qquad \text{(Equation 4)}$$

Generalization error of support vector machines

Several research projects have been conducted to estimate the generalization error of SVMs. The most common studies involve the radius/margin ratio (Vapnik, 1995), the concept of span of support vectors (Vapnik and Chapelle, 2000), the VC dimension (Vapnik, 1995), the $\xi\alpha$-estimator (Joachims, 2000) and the generalized approximate cross-validation (GACV) (Wahba, 1999), besides the ordinary k-fold cross-validation.

Based on a recent comparison between these measures in the problem of choosing optimal SVM hyperparameters (Duan et al., 2002), we chose to use the GACV, because of its smoother variation with respect to different subsets of genes when compared with the $\xi\alpha$-estimator and the k-fold cross-validation (the other measures performed poorly when employed with the type of SVM used here).

The GACV is a computable proxy for the generalized Kullback-Liebler distance for SVMs (Wahba, 1999), which is considered to be an upper bound on misclassification rate. The GACV for a given set of tunable SVM parameters (in this study, it is a subset of genes) λ is defined as:

$$GACV(\lambda) = \frac{1}{n}\left[\sum_{i=1}^{n}\xi_i + 2\sum_{y_i f_{\lambda i} < -1}\alpha_i \mathrm{K}_{ii} + \sum_{y_i f_{\lambda i} \in [-1,1]}\alpha_i \mathrm{K}_{ii}\right] \qquad \text{(Equation 5)}$$

where n is the number of training samples, ξ_{i} are the slack variables, y_i are the class labels, $f_{\lambda i} = f(\mathrm{x}_i)$ and $K_{ii} = k(x_i, x_i)$.

In order to understand this estimate, it is important to examine each of the three terms of Equation 5. The first one penalizes training errors, assuring that at least the training set could be learnt. The second one penalizes severe errors, i.e., samples for which the classifier is pretty sure that its prediction is correct ($|f_{\lambda i}|>1$), while it is indeed not. This type of error is the most critical, so it gets a factor 2 penalization. The third one penalizes weak predictions, i.e, those that can be as right as wrong, but with little confidence.

Multi-class support vector machines

SVMs are methods originally designed to work only with binary problems. A standard way to solve multi-class problems is to consider them as a collection of binary subproblems, and then combine their solutions. In this context, two approaches are most commonly employed: the one-versus-all (OVA) and the one-versus-one (OVO) (Rifkin et al., 2003).

The OVA method constructs *N* (the number of classes) SVMs. The *ith* SVM is trained with all the samples in the *ith* class that have a positive class label and with all the other samples with negative class labels. The final output is the class that corresponds to the SVM with the highest output (Equation 3).

The OVO method constructs *N(N-1)/2* SVMs, taking into consideration all binary combinations of classes. Then, test samples are used on all the SVMs, and their outputs are combined (usually using a MaxWins scheme or the DDAG algorithm (Platt et al., 1999)).

The choice of the OVA or the OVO approach is domain specific. In the case of gene expression data, for which the number of training samples is usually very reduced, the OVA approach seems to work better, as it uses all the available data to construct the *N* SVMs. In an empirical study of multi-class methods and algorithms applied to microarray data, Rifkin et al. (2003) showed that the OVA approach, in combination with SVM, gave the most accurate method by a significant margin. Consequently, we employed the OVA approach.

In order to measure the generalization performance of the multi-class classifier, the GACV estimator of the individual SVMs was calculated and summed. The error of the multi-class classifier with respect to the subset of genes λ is given by:

$$Error_{multi}(\lambda) = \frac{1}{N}\sum_{i=1}^{N} GACV_i(\lambda) \qquad \text{(Equation 6)}$$

where *N* is the number of classes and $GACV_i$ (λ) is the measure related to the *ith* SVM.

Genetic algorithms for gene selection

GAs are search and optimization techniques inspired by the process of evolution of biological organisms (Goldberg, 1989). A simple GA uses a population of individuals to solve a given problem. Each individual, or chromosome, of the population corresponds to an encoded possible solution for the problem. A reproduction-based mechanism is applied to the population, generating a new population. The population usually evolves through several generations, until a suitable solution is reached. A GA starts generating an initial population formed by a random group of individuals, which can be seen as first guesses to solve the problem. The initial population is evaluated and a score (named fitness) is given for each chromosome, reflecting the quality of the solution associated with it.

The main features of the GA were:

Representation

Chromosomes are encoded by a binary representation. They consist of two parts. The first one encodes the set of genes. It has length *m*, where *m* is the total number of genes in the domain. A 1 or 0 at the *ith* position of the chromosome indicates the presence or absence of the *ith* gene in the sample. The second part encodes the values of the SVM hyperparameters *C*. The *C* for each SVM in the domain is codified by two bits. So, four different values can be expressed for each SVM. These are 0.1, 1, 10, and 100. We tried other values, but they did not give better performance. In fact, they unnecessarily enlarged the search space for selection parameters.

Initialization

Only a few positions of the first part of the chromosomes were initialized. We believe that only a few genes are relevant for tissue classification. The second part was uniformly initialized.

Mutation

Mutation is the genetic operator responsible for maintaining diversity in the population. Mutation operates by randomly "flipping" bits of the chromosome, based on some probability. A usual mutation probability is *1/p*, where *p* is the length of each of the two parts of the chromosomes.

Crossover

This genetic operator is used to guide the evolutionary process through potentially better solutions. This is performed by interchanging genetic material of chromosomes in order to create individuals that can benefit from their parents' fitness. We used a uniform crossover with an empirically defined probability rate.

Replacement

Replacement schemes determine how a new population is generated. We used the concept of overlapping populations, where parents and offspring are merged, and the best individuals from this union will form the next population.

Selection

This is the process of choosing parents for reproduction. Usually, it emphasizes the best solutions in the population, but since the replacement scheme employed here already offers enough evolutionary pressure, a random selection approach was chosen.

Random immigrant

This is a method that helps to keep diversity in the population, minimizing the risk of premature convergence (Congdon, 1995). It works by replacing the individuals whose fitness is under the mean by recently initialized individuals. Random immigrant is invoked when the best individual does not change for a certain number of generations (here named re-start frequency).

Fitness

A good function for gene selection should focus on subsets of genes that minimize an estimate of generalization error of the classifier, while punishing subsets with many genes. The fitness function associated with a gene subset λ is expressed by:

$$fitness(\lambda) = error_{multi}(\lambda) + dim(\lambda)/max\_dim \qquad \text{(Equation 7)}$$

where $error_{multi}(\lambda)$ is defined in Equation 6, $dim(\lambda)$ is the dimensionality of the subset λ and *max_dim* is the original dimension of the full set of genes.

Although this function satisfies both requirements for a fitness function in the gene selection problem, it was defined in an *ad hoc* manner. Thus, readers are encouraged to try out their own fitness functions in their domains.

RESULTS AND DISCUSSION

The implementation of the multi GA-SVM method was based on the LIBSVM package (available at http://www.csie.ntu.edu.tw/~cjlin/libsvm/) and on the GALIB library (available at http://lancet.mit.edu/ga/). The principal aims of the experiments were two-fold: first, we compared the performance of the classifiers trained with the set of genes selected by the proposed method and trained with the whole set of genes. Second, we compared the results obtained with the multi GA-SVM method with other results found in the literature, regarding both accuracy rate and dimensionality reduction.

The parameters used by the GA were all empirically defined: population: 100, generations: 1000, crossover prob.: 0.8, and re-start frequency: 20. Since GAs are stochastic methods, there are no guarantees that the same subset of genes will be encountered in different runs. Actually, this is very unlikely. Thus, mean (μ) and standard deviation (σ) were used in 10 runs of the multi GA-SVM method.

Small round blue cell tumor dataset

The small round blue cell tumors of childhood (SRBCT) dataset was first analyzed by Khan et al. (2001). It consists of the expression of 6,567 genes measured for 88 samples. The technology employed was cDNA microarrays. After an initial filtering, 2,308 genes were included. The tumors were classified as Burkitt lymphoma (BL), Ewing sarcoma (EWS), neuroblastoma (NB), or rhabdomyosarcoma (RMS). The dataset was split into a training set of 63 samples (23 EWS, 8 BL, 12 NB, and 20 RMS) and a testing set of 25 samples (6 EWS, 3 BL, 6 NB, 5 RMS, and 5 non-SRBCT).

With 2,308 genes, the SVM-OVA approach made five errors, yielding an accuracy of 80%. The optimal values of *C* for the SVMs trained for each class were determined by the minimizers of the GACV estimator over the four values of *C*. The values found were: 0.1, 0.1, 0.1, and 0.1.

Li et al. (2004) studied the interaction between various feature selection techniques and classification methods (including SVM) for gene expression data. With 150 genes, they achieved an accuracy greater than 93%. Using a complex neural network approach, Khan et al. (2001) achieved a test error of 0 and identified 96 genes for the classification. With their nearest shrunken centroid method, Tibshirani et al. (2002) reduced the required number to make a perfect classification to 43 genes. The best performance found in the literature was with Lee and Lee's (2003) method, which achieved an accuracy of 100% using only 20 genes.

Using 28.6 genes, on average, the multi GA-SVM method made perfect scores in 6 of 10 runs, yielding an accuracy of 98.00%/2.58% (μ/σ). In four runs, two samples were misclassified: sample TEST-2 (one time) and sample TEST-20 (three times). In Khan et al.'s (2001) work, the latter was correctly classified but, due to its very low confidence level, it was not diagnosed. The optimal values of *C* varied over the GA runs, influenced by (and influencing) the selected subset of genes.

CONCLUSIONS

We have presented a novel gene selection method for multi-class problems based on GAs and SVM. Besides selecting an optimal subset of genes, it also jointly optimizes the SVM hyperparameter *C* for each SVM in the multi-class framework, which seems ideal, since the subset of genes influences the parameters *C*, and vice versa.

Due to space constraints, the selected genes are not shown here. They can be downloaded from http://www.icmc.usp.br/~bferes/wob2004/. Most of the top selected genes are very likely to be biologically relevant for the investigated phenomena. By comparing these genes with those selected by other gene selection methods, one can see that there is a strong overlap between the sets.

In the first experimental tests using the SRBCT dataset, small subsets of genes were selected and a significant improvement was reported (applying a two-tailed *t*-test) when compared with the accuracies using the subsets of genes selected by the multi GA-SVM method and the whole set of genes. A similar performance was achieved with another gene expression dataset (MLL); the results of the analysis of this dataset can be retrieved from the above-mentioned website. Comparing the results with the best report found in the literature (see the Results and Discussion section) for the SRBCT dataset, no significant difference was found, although the Lee and Lee (2003) method selected a slightly smaller subset of genes than the average selected by the multi GA-SVM method.

We plan to test this new method with other gene expression datasets and to deeply investigate the biological interpretation of the selected genes. A systematic comparison with other feature selection techniques is also in progress.

ACKNOWLEDGMENTS

The authors thank CAPES for research support.

REFERENCES

Alon, U., Barkai, N., Notterman, D.A., Gish, K., Ybarra, S., Mack, D. and **Levine, A.J.** (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natl. Acad. Sci. USA 96*: 6745-6750.

Congdon, C.B. (1995). A comparison of genetic algorithm and other machine learning systems on a complex classification task from common disease research. PhD thesis, Computer Science and Engineering Division, Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI, USA.

Cristianini, N. and **Shawe-Taylor, J.** (2000). *An Introduction to Support Vector Machines* (*And Other Kernel-Based Learning Methods*). Cambridge University Press, Cambridge, UK.

Duan, K., Keerthi, S.S. and **Poo, A.** (2002). Evaluation of simple performance measures for tuning SVM hyperparameters. *Neurocomputing 51*: 41-59.

Fröhlich, H., Chapelle, O. and **Schölkopf, B.** (2003). Feature Selection for Support Vector Machines by Means of Genetic Algorithms. In: ICTAI'03: *Proceedings of the 15th IEEE International Conference on Tools with Artificial Inteligence*, Washington, DC, USA. IEEE Computer Society, p. 142.

Goldberg, D.E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing, Boston, MA, USA.

Joachims, T. (2000). Estimating the Generalization Performance of a SVM Efficiently. In: *Proceedings of ICML-00*. Morgan Kaufmann Publishers, San Francisco, CA, USA, pp. 431-438.

Khan, J., Wei, J.S., Ringner, M., Saal, L.H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C.R., Peterson, C. and **Meltzer, P.S.** (2001). Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nat. Med.* 7: 673-679.

Lee, Y. and **Lee, C.** (2003). Classification of multiple cancer types by multicategory support vector machines using gene expression data. *Bioinformatics 19*: 1132-1139.

Li, T., Zhang, C. and **Ogihara, M.** (2004). A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression. *Bioinformatics 20*: 2429-2437.

Platt, J., Cristianini, N. and **Shawe-Taylor, J.** (1999). Large margin DAGs for multi-class classification. *Adv. Neural Inf. Process. Syst. 12*: 547-553.

Rifkin, R., Mukherjee, S., Tamayo, P., Ramaswamy, S., Yeang, C.H., Angelo, M., Reich, M., Poggio, T., Lander, E.S., Golub, T.R. and **Mesirov, J.P.** (2003). An analytical method for multi-class molecular cancer classification. *SIAM Reviews 45*: 706-723.

Schena, M. and **Knudsen, S.** (2004). *Guide to Analysis of DNA Microarray Data*. Wiley Publishers, Canada.

Tibshirani, R., Hastie, T., Narasimhan, B. and **Chu, G.** (2002). Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proc. Natl. Acad. Sci. USA 99*: 6567-6572.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, USA.

Vapnik, V. and **Chapelle, O.** (2000). Bounds on error expectation for support vector machines. *Neural Comput. 12*: 2013-2036.

Wahba, G. (1999). Support vector machine, reproducing kernel Hilbert spaces and the randomized GACV. In: *Advances in Kernel Methods - Support Vector Learning* (Scholkopf, B., Burges, C. and Smola, A., eds.). MIT Press, Cambrigde, MA, USA, pp. 69-88.