Proceedings of the 28th IEEE
EMBS Annual International Conference
New York City, USA, Aug 30-Sept 3, 2006

Applications of Nonlinear System Identification in Molecular Biology

Michael J. Korenberg

Abstract—**Here we show how nonlinear system identification techniques, such as fast orthogonal search (FOS) and the orthogonal search method (OSM), can be used to analyze gene expression profiles and predict the class to which a profile belongs.**

I. Introduction

A gene expression profile $\boldsymbol{p}_j$ can be thought of as a column vector containing the expression levels $e_{i,j}$, $i = 1,...,I$ of I genes. We suppose that we have J of these profiles for training, so that j = 1,…,J. Each of the profiles $\boldsymbol{p}_j$ was created from a sample, e.g., from a tumor, belonging to some class. The samples may be taken from patients diagnosed with various classes of leukemia, e.g., acute lymphoblastic leukemia (ALL) or acute myeloid leukemia (AML), as in the paper by Golub et al. [1].

Given a training set of profiles belonging to known classes, e.g., ALL and AML, the problem is to create a predictor that will assign a new profile to its correct class. This problem was extensively considered in [1]. However, the use of FOS [2,3], OSM [2-4], or an analogous form of model building is not disclosed in that paper. Indeed, the class predictor created here through the use of OSM is different and correctly classified more profiles in an independent set, using less training data, than in [1]. The calculations below can be done more efficiently using FOS, but are described using a variant of OSM so as to be easier to follow.

First, consider distinguishing between two classes. To distinguish between ALL and AML classes, Golub et al. [1] defined "an 'ideal expression pattern' corresponding to a gene that was uniformly high in one class and uniformly low in the other". They then selected a set of "informative genes" that were "chosen based on their correlation with the class distinction", and used these genes in a voting scheme to classify a given expression profile [1]. In particular, the "set of informative genes to be used in the predictor was chosen to be the 50 genes most closely correlated with AML-ALL distinction in the known samples" [1]. A possible drawback of this approach is that some genes that might always have similar expression levels could be selected, which may unfairly bias the voting.

II. Mathematical Approach

Define the output $y(j)$ of an ideal classifier to be –1 for each profile $\boldsymbol{p}_j$ from the first class, and 1 for each profile $\boldsymbol{p}_j$ from the second class. For each of the i genes, i = 1,…,I, define

$$g_i(j) = e_{i,j}\,, \tag{1}$$

the expression level of the i-th gene in the j-th training profile, j = 1,…,J. We then wish to choose a subset $\tilde{g}_m(j),\ m = 1,...,M$, of the I candidate functions $g_i(j)$ to approximate $y(j)$ by

$$y(j) = \sum_{m=1}^{M} a_m \tilde{g}_m(j) + r(j) \tag{2}$$

where a_m is the coefficient for each term in the series, and where $r(j)$ is the model error, so as to minimize the mean-square error (MSE)

$$MSE = \frac{1}{J}\sum_{j=1}^{J}(r(j))^2 \tag{3}$$

The subset $\tilde{g}_m(j),\ m = 1,...,M$, containing the model terms can be found by using FOS or OSM to search efficiently through the larger set of candidate functions $g_i(j)$, i = 1,…,I, as follows. In succession, we try each of the I candidate functions and measure the reduction in MSE if that candidate alone were best-fit, in the mean-square sense, to $y(j)$, i.e., if M = 1 in (2). The candidate for which the MSE reduction would be greatest is chosen as the first term for the model in (2). To find the second term for the model, we set M = 2. Then each of the remaining I-1 candidates is orthogonalized relative to the chosen model term. This enables the MSE reduction to be efficiently calculated were any particular candidate added as the second term in the model. We select the candidate for which the MSE reduction would be greatest to be the second model term, and so on.

In this scheme, candidate functions are orthogonalized with respect to already-selected model terms. After the orthogonalization, a candidate whose mean-square would be

Manuscript received May 15, 2006.

M. J. Korenberg is with the Department of Electrical and Computer Engineering, Queen's University, Kingston, Ontario, K7L 3N6 Canada (phone: 613-533-2931; fax: 613-533-6615; e-mail: korenber@ post.queensu.ca).

less than some threshold value is barred from selection [2,3]. This prevents numerical errors associated with fitting orthogonalized functions having small norms. It prevents choosing near duplicate candidate functions, corresponding to genes that always have virtually identical expression levels.

In fact, to increase efficiency, the orthogonal functions need not be explicitly created. Rather, FOS [2,3] uses a Cholesky decomposition to rapidly assess the benefit of adding any candidate as a further term in the model. The method is related to, but more efficient than, a technique proposed by Desrochers [4]. The selection of model terms can be terminated once a pre-set number have been chosen. For example, since each candidate function $g_i(j)$ is defined only for J values of j, there can be at most J linearly independent candidates, so that at most J model terms can be selected. (However, there will typically be far more than J candidates that are searched.) In addition, a stopping criterion, based on a standard correlation test [3], can be employed. Alternatively, various tests such as the Information Criterion [5], or an F-test, discussed e.g. in [6], can be used to stop the process.

Once the model terms have been selected for (2), the coefficients a_m can be immediately obtained [2,3] from quantities already calculated in carrying out the FOS algorithm. Further details about OSM and FOS are contained in the cited papers. The FOS selection of model terms can also be carried out iteratively [7] for possibly increased accuracy.

Once the model of (2) has been determined, it can then function as a predictor as follows. If $\boldsymbol{p}_{J+1}$ is a novel expression profile to be classified, then let $\tilde{g}_m(J+1)$ be the expression level of the gene is this profile corresponding to the m-th model term in (2). (This gene is typically not the m-th gene in the profile, since $\tilde{g}_m(j)$, the m-th model term, is typically not $g_m(j)$, the m-th candidate function.) Then evaluate

$$z = \sum_{m=1}^{M} a_m \tilde{g}_m(J+1), \tag{4}$$

and use a test of similarity to compare z with –1 (for the first class) and 1 (for the second class). For example, if $z < 0$, the profile may be predicted to belong to the first class, and otherwise to the second class.

Alternatively, suppose that MSE_1 and MSE_2 are the MSE values for the training profiles in classes 1 and 2 respectively. For example, the calculation to obtain MSE_1 is carried out analogously to (3), but with the averaging only over profiles in class 1. The MSE_2 is calculated similarly for class 2 profiles. Then, assign the novel profile $\boldsymbol{p}_{J+1}$ to class 1 if

$$\frac{(z+1)^2}{MSE_1} < \frac{(z-1)^2}{MSE_2}, \tag{5}$$

and otherwise to class 2. In place of using a mean-square test of similarity, analogous tests using absolute values or a power higher than 2 can be employed.

Alternatively, once the model terms for (2) have been selected by FOS, the genes to which they correspond can then be used as a set of “informative genes” in a voting scheme such as described by Golub et al. [1].

Above, for simplicity, we have used the expression level of one gene to define a candidate function, as in (1). However, we can also define candidate functions in terms of powers of the gene’s expression level, or in terms of crossproducts of two or more genes’expression levels, or the candidate functions can be other functions of some of the genes’ expression levels. Also, in place of the raw expression levels, the logarithm of the expression levels can be used, after first increasing any negative raw value to some positive threshold value [1].

While FOS avoids the explicit creation of orthogonal functions, which saves computing time and memory storage, other procedures can be used instead to select the model terms and still conform to the present approach. For example, an orthogonal search method [2-4], which does explicitly create orthogonal functions can be employed, and one way of doing so is shown in the Example below. Alternatively, a process that does not involve orthogonalization can be used. For example, the set of candidate functions is first searched to select the candidate providing the best fit to $y(j)$, in a mean-square sense, absolute value of error sense, or according to some other criterion of fit. Then, for this choice of first model term, the remaining candidates are searched to find the best to have as the second model term, and so on. Once all model terms have been selected, the model can be “refined” by reselecting each model term, each time holding fixed all other model terms [7].

Alternatively, one or more profiles from each of the two classes to be distinguished can be spliced together to form a training input. The corresponding training output can be defined to be –1 over each profile from the first class, and 1 over each profile from the second class. The nonlinear system having this input and output could clearly function as a classifier, and at least be able to distinguish between the training profiles from the two classes. Then FOS can be used to build a model that will approximate the input output behavior of the nonlinear system [2,3] and thus function as a class predictor for novel profiles.

It will also be appreciated that the class distinction to be made may be based on phenotype, for example, the clinical outcome in response to treatment. In this case, the approach herein can be used to establish genotype phenotype correlations, and to predict phenotype based on genotype.

Finally, while distinctions between two classes have been considered above, predictors for more than two classes can be built analogously. For example, the output *y*(*j*) of the ideal classifier can be defined to have a different value for profiles from different classes. Alternatively, as is well known, the multi-class predictor can readily be realized by various arrangements of two-class predictors.

III. Example

The first 11 ALL profiles (#1 - #11 of Golub et al. first data set), and all 11 of the AML profiles (#28 - #38 of the same data set), formed the training data. These 22 profiles were used to build 10 concise models of the form in (2), which were then employed to classify profiles in an independent set in [1]. The first 7000 gene expression levels in each profile were divided into 10 consecutive sets of 700 values. For example, to build the first model, the expression levels of genes in positions 1 – 700 in each training profile were used to create 700 candidate functions $g_i(j)$. These candidates were defined as in (1), except that in place of each raw expression level $e_{i,j}$, its log was used:

$$g_i(j) = \log_{10} e_{i,j}\,, \tag{6}$$

$$i = 1, \dots, 700 \qquad j = 1, \dots, 22,$$

after increasing to 100 any raw expression value that was less. Similarly, genes 701 – 1400 of each training profile were used to create a second set of 700 candidate functions, for building a second model of the form in (2), and so on. The training profiles had been ordered so that the $\boldsymbol{p}_j$, for *j* = 1,…,11 corresponded to the ALL profiles, and for *j* = 12,…,22 to the AML profiles. Hence the training output was defined as

$$\begin{aligned} y(j) &= -1, \quad j = 1,\dots,11 \\ &= 1, \quad j = 12,\dots,22 \end{aligned} \tag{7}$$

The following procedure was used to find each model. First, each of the 700 candidate functions was tried as the only model term in (2) (with *M* = 1), and its coefficient chosen to minimize the *MSE* given by (3). The candidate for which the *MSE* was smallest was selected as the first model term $\tilde{g}_1(j)$. For *j* = 1,…,22, define a first orthogonal function as $\tilde{w}_1(j) = \tilde{g}_1(j)$, with its coefficient $\tilde{c}_1$. Assume that the model already has *M* terms, for *M* ≥ 1, and a (*M*+1)-th term is sought. For *j* = 1,…,22, let $\tilde{w}_m(j)$ be orthogonal functions created from the model chosen terms $\tilde{g}_m(j)$, *m* = 1,…,*M*. Let $\tilde{c}_m$ be the corresponding coefficients of the $\tilde{w}_m(j)$, where these coefficients were found to minimize the mean-square error of approximating *y*(*j*) by a linear combination of the $\tilde{w}_m(j)$, *m* = 1,…,*M*. Then, for each candidate $g_i(j)$ not already chosen for the model, the modified Gram-Schmidt procedure is used [2,4] to create a function orthogonal to the $\tilde{w}_m(j)$, *m* = 1,…,*M*. Thus, for *j* = 1,…,22, set

$$w_{M+1}^{(1)}(j) = g_i(j) - \alpha_{M+1,1}\tilde{w}_1(j)\,, \tag{8}$$

where

$$\alpha_{M+1,1} = \frac{\sum_{j=1}^{22} g_i(j)\tilde{w}_1(j)}{\sum_{j=1}^{22} (\tilde{w}_1(j))^2}\,. \tag{9}$$

And, for *r* = 2,…,*M*, and *j* = 1,…,22, set

$$w_{M+1}^{(r)}(j) = w_{M+1}^{(r-1)}(j) - \alpha_{M+1,r}\tilde{w}_r(j) \tag{10}$$

where

$$\alpha_{M+1,r} = \frac{\sum_{j=1}^{22} w_{M+1}^{(r-1)}(j)\tilde{w}_r(j)}{\sum_{j=1}^{22} \left(\tilde{w}_r(j)\right)^2} \tag{11}$$

The function $w_{M+1}^{(M)}(j)$ (which was created from the candidate $g_i(j)$) is orthogonal to the $\tilde{w}_m(j)$, *m* = 1,…,*M*. If the candidate $g_i(j)$ were added to the model as the (*M*+1)-th term, then it follows readily that the model *MSE* would equivalently be

$$e = \frac{1}{22}\sum_{j=1}^{22}\left(y(j) - \sum_{m=1}^{M} \tilde{c}_m \tilde{w}_m(j) - c_{M+1} w_{M+1}^{(M)}(j) \right)^2$$

where

$$c_{M+1} = \frac{\sum_{j=1}^{22} y(j) w_{M+1}^{(M)}(j)}{\sum_{j=1}^{22} \left(w_{M+1}^{(M)}(j)\right)^2} \tag{12}$$

Therefore, the candidate $g_i(j)$ for which *e* is smallest is taken as the (*M*+1)-th model term $\tilde{g}_{M+1}(j)$, the corresponding $w_{M+1}^{(M)}(j)$ becomes $\tilde{w}_{M+1}(j)$, and the corresponding c_{M+1} becomes $\tilde{c}_{M+1}$. Once all model terms $\tilde{g}_m(j)$ have been selected, their coefficients a_m that minimize the *MSE* can be readily calculated [2,3] using back substitution twice.

Each of the 10 models was limited to five model terms. For example, the terms for the first model corresponded to genes in positions #697, #312, #73, #238, #275 in the profiles and the model *%MSE* (expressed relative to the variance of the training output) was 6.63%. As noted, the coefficients a_m of the model terms $\tilde{g}_m(j)$ were obtained for each model so as to least-squares fit the training output $y(j)$, *j* = 1,…,22.

The 10 identified models were then used to classify profiles in an independent, second data set from [1], which contained 20 ALL and 14 AML test profiles. For each model, the value of z was calculated using (4) with $M = 5$, and with $\tilde{g}_m(J+1)$ equal to the log of the expression level of the gene in the test profile corresponding to the m-th model term. For example, for model #1, $\tilde{g}_1(J+1)$, $\tilde{g}_2(J+1)$,…, $\tilde{g}_5(J+1)$ corresponded to the log of the expression level of genes at positions #697, #312,…,#275 respectively, in the test profile. The values of z for the 10 models were summed; if the result was negative, the test profile was classified as ALL, and otherwise as AML.

By this means, all 20 of the test ALL, and all 14 of the test AML, profiles in the independent set were correctly classified. These classification results, after training with 22 profiles, compare favorably with those for the Golub et al. method. Golub et al. [1] used the entire first data set of 38 profiles to train their ALL-AML predictor, which then was able to classify correctly all of the independent set except for five where the prediction strength was too low for a decision.

In order to investigate how small an individual model could be and still allow the combination to be an effective classifier, the procedure was repeated using the above sets of 700 candidate functions, but this time to build 10 two-term models.

Again, a test profile was classified by calculating the value of z for each model using (4), this time with $M = 2$, and then adding the values of z for the 10 models; if the result was negative, the test profile was classified as ALL, and otherwise as AML. By this means, all 20 of the test ALL, and 13 of the 14 test AML, profiles in the independent set were correctly classified.

Moreover, it was found that considerably less than full training profiles sufficed to maintain this level of accuracy for the two-term models. The identical classification accuracy was obtained with only models #1 – #6 (whose construction required only the first 4200 genes of each training profile), or with additional models. The first 4200 gene expression levels in each of 22 profiles that sufficed for training here represent less than 40% of the data used by Golub et al. [1].

It should be noted that individually the models made a number of classification errors, ranging from 1 – 17 errors for the two-term and from 2 – 11 for the five-term models. This was not unexpected since each model was created after searching through a relatively small subset of 700 expression values to create the model terms. However, the combination of several models resulted in excellent classification.

IV. Discussion

The principle of this aspect of the present approach is to separate the values of the training gene expression profiles into subsets, to find a model for each subset, and then to use the models together for the final prediction, e.g. by summing the individual model outputs or by voting. Moreover, the subsets need not be created consecutively, as above. Other strategies for creating the subsets could be used, e.g. by selecting every 10th expression level for a subset.

This principle can increase classification accuracy over that from finding a single model using entire gene expression profiles. Note that here, since the output $y(j)$ was defined only for $j = 1,…,22$, at most 22 independent terms could be included in the model (which would allow no redundancy), but identifying a number of models corresponding to the different subsets allows the contributions of many more genes to be taken into account. Indeed, searching through the first 7000 expression levels to find a five-term model, using the same 22 training profiles, resulted in a *%MSE* of 1.33%, with terms corresponding to genes at positions #6200, 4363, 4599, 4554, and 3697 in the profiles. However, this model was not particularly accurate, misclassifying 4 of the 20 ALL, and 5 of the 14 AML, profiles in the independent set.

Finally, it will be appreciated that the principle of dividing the training profiles into subsets, and finding models for the subsets, then using the models in concert to make the final classification decisions, is not confined to use of FOS, OSM, or any particular model-building technique. For example, a parallel cascade model [8] can be found for each subset, and then the models can be used together to make the final predictions.

References

[1] T. R. Golub, et al., “Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring,” *Science*, vol. 286, pp. 531–537, 1999.

[2] M. J. Korenberg, “A robust orthogonal algorithm for system identification and time series analysis,” *Biol. Cybern.*, vol. 60, pp. 267–276, 1989.

[3] M. J. Korenberg, “Fast orthogonal algorithms for nonlinear system identification and time-series analysis,” in *Advanced Methods of Physiological System Modeling*, vol. 2, V. Z. Marmarelis, Ed. Los Angeles: Biomedical Simulations Resource, 1989, pp. 165–177.

[4] A. A. Desrochers, “On an improved model reduction technique for nonlinear systems,” *Automatica*, vol. 17, pp. 407–409, 1981.

[5] H. Akaike, “A new look at the statistical model identification,” *IEEE Trans. Automatic Control*, vol. 19, pp. 716–723, 1974.

[6] T. Soderstrom, “On model structure testing in system identification,” *Int. J. Control*, vol. 26, pp. 1–18, 1977.

[7] K. M. Adeney and M. J. Korenberg, “Fast orthogonal search for array processing and spectrum estimation,” *IEE Proc. Vis. Image Signal Process.*, vol. 141, pp. 13–18, 1994.

[8] M. J. Korenberg, “On predicting medulloblastoma metastasis by gene expression profiling,” *J. Proteome Research*, vol. 3, no. 1, pp. 91–96, 2004.