Estimation of Distribution Algorithms as Logistic Regression Regularizers of Microarray Classifiers

C. Bielza[1]; V. Robles[2]; P. Larrañaga[1]
[1]Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid, Spain;
[2]Departamento de Arquitectura y Tecnología de Sistemas Informáticos, Universidad Politécnica de Madrid, Spain

Keywords
Logistic regression, regularization, estimation of distribution algorithms, DNA microarrays

Summary
Objectives: The "large *k* (genes), small *N* (samples)" phenomenon complicates the problem of microarray classification with logistic regression. The indeterminacy of the maximum likelihood solutions, multicollinearity of predictor variables and data over-fitting cause unstable parameter estimates. Moreover, computational problems arise due to the large number of predictor (genes) variables. Regularized logistic regression excels as a solution. However, the difficulties found here involve an objective function hard to be optimized from a mathematical viewpoint and a careful required tuning of the regularization parameters.
Methods: Those difficulties are tackled by introducing a new way of regularizing the logistic regression. Estimation of distribution algorithms (EDAs), a kind of evolutionary algorithms, emerge as natural regularizers. Obtaining the regularized estimates of the logistic classifier amounts to maximizing the likelihood function via our EDA, without having to be penalized. Likelihood penalties add a number of difficulties to the resulting optimization problems, which vanish in our case. Simulation of new estimates during the evolutionary process of EDAs is performed in such a way that guarantees their shrinkage while maintaining their probabilistic dependence relationships learnt. The EDA process is embedded in an adapted recursive feature elimination procedure, thereby providing the genes that are best markers for the classification.
Results: The consistency with the literature and excellent classification performance achieved with our algorithm are illustrated on four microarray data sets: *Breast*, *Colon*, *Leukemia* and *Prostate*. Details on the last two data sets are available as supplementary material.
Conclusions: We have introduced a novel EDA-based logistic regression regularizer. It implicitly shrinks the coefficients during EDA evolution process while optimizing the usual likelihood function. The approach is combined with a gene subset selection procedure and automatically tunes the required parameters. Empirical results on microarray data sets provide sparse models with confirmed genes and performing better in classification than other competing regularized methods.

Correspondence to:
Concha Bielza
Facultad de Informática
Campus de Montegancedo s/n
28660 Boadilla del Monte, Madrid
Spain
E-mail: mcbielza@fi.upm.es

Methods Inf Med 2009; 48: 236–241
doi: 10.3414/ME9223
prepublished: March 31, 2009

1. Introduction

The development of DNA microarray technology allows screening of gene expression levels from different tissue samples (e.g. cancerous and normal). The resulting gene expression data help explore gene interactions, discover gene functions and classify individual cancerous/normal samples, using different supervised learning techniques [1, 2].

Among these techniques, logistic regression [3] is widely used because it provides explicit probabilities of class membership, interpretation of the regression coefficients of predictor variables and it avoids gaussianity or correlation structure assumptions.

Microarray classification is a challenging task since these data typically involve extremely high dimensionality (thousands of genes) and small sample sizes (less than one hundred cases). This is the so-called "large *k* (variables), small *N* (samples) problem" or the "curse of dimensionality". This may cause a number of statistical problems for estimating parameters properly. First, a large number of parameters have to be estimated using a very small number of samples. Therefore, an infinite number of solutions is possible as the problem is undetermined. Second, multicollinearity largely exists. The likelihood of some gene profiles being linear combinations of other gene profiles grows as more and more variables are introduced into the model, thereby supplying no new information. Third, over-fitting may occur, i.e. the model may fit the training data well but perform badly on new samples. These problems yield unstable parameter estimates. Furthermore, there are also computational problems due to the large number of predictor variables. Traditional numerical algorithms for finding the estimates, like Newton-Raphson's method [4], require prohibitive computa-

tions to invert a huge, sometimes singular matrix, at each iteration.

To alleviate this situation within the context of *logistic regression*, many authors use techniques of dimensionality reduction and feature (or variable) selection [5]. Feature selection methods yield parsimonious models which reduce information costs, are easier to explain and understand, and increase model applicability and robustness. The goodness of a proposed gene subset may be assessed via an initial screening process where genes are selected in terms of some univariate or multivariate scoring metric (*filter* approach [6]). By contrast, *wrapper* approaches search for good gene subsets using the classifier itself as part of their function evaluation [7]. A performance estimate of the classifier trained with each subset assesses the merit of this subset.

Imposing a *penalty* on the size of logistic regression coefficients is another different solution. Finding a maximum likelihood estimate subject to spherical restrictions on the logistic regression parameters leads to *ridge* or quadratic (penalized) logistic regression [8]. Therefore, the ridge estimator is a restricted maximum likelihood estimator (MLE). Shrinking the coefficients towards zero and allowing a little bias provide more stable estimates with smaller variance.

Apart from ridge penalization, there are other penalties within the more general framework of *regularization* methods. All of them aim at balancing the fit to the data and the stability of the estimates. These methods are much more efficient computationally than wrapper methods with the similar performance. Furthermore, regularization methods are more continuous than usual discrete processes of retaining-or-discarding features thereby not suffering as much from high variability.

Here we introduce estimation of distribution algorithms (EDAs) as natural regularizers within the logistic regression context. EDAs are a recent optimization heuristic included in the class of stochastic population-based search methods [9]. EDAs work by constructing an explicit probability model from a set of selected solutions, which is then conveniently used to generate new promising solutions in the next iteration of the evolutionary process. An optimization heuristic is an appropriate tool since shaping the logistic classifier means estimating its parameters, which in turn entails solving a maximization problem. Unlike traditional numerical methods, EDAs do not require derivative information or matrix inversions. Moreover, used as fitness functions, EDAs could similarly maximize penalized likelihoods to tackle the $k>>N$ problem. This would just reveal the potential of a heuristic (EDA) against a numerical (Newton-Raphson) method. In this paper we will show that the EDA framework is so general that, under certain parameterizations, it obtains the regularized estimates in a natural way, without penalizing the original likelihood. EDAs receive the unrestricted likelihood equations as inputs and they generate the restricted MLEs as outputs.

2. Methods

2.1 Logistic Regression for Microarray Data

Assume we have a (training) data set D_N of N independent samples from microarray experiments $\mathrm{D}_N = \{(c_j, x_{j1}, ..., x_{jk}), j = 1, ..., N\}$, where $\mathbf{x}_j = (x_{j1}, ..., x_{jk})^t \in \mathrm{R}^k$ is the gene expression profile of the j-th sample, x_{ji} indicates the i-th gene expression level of the j-th sample and c_j is the known class label of the j-th sample, 0 or 1, for the different states. We assume the expression profile $\mathbf{x}$ to be preprocessed, log-transformed and standardized to zero mean and unit variance across genes.

Let $\pi_j, j = 1, ..., N$ denote $P(C = 1 \mid \mathbf{x}_j)$, i.e. the conditional probability of belonging to the class state 1 given gene expression profile $\mathbf{x}_j$. Then the logistic regression model is defined as

$$\log \frac{\pi_j}{1-\pi_j} = \beta_0 + \sum_{i=1}^{k} \beta_i x_{ji} = \eta_j \quad \Leftrightarrow \quad \pi_j = \frac{1}{1+e^{-\eta_j}} \tag{1}$$

where $\boldsymbol{\beta} = (\beta_0, \beta_1, ..., \beta_k)^t$ denotes the vector of regression coefficients including intercept β_0. From D_N, the log-likelihood function is built as

$$l(\boldsymbol{\beta}) = \sum_{j=1}^{N} (c_j \log \pi_j + (1-c_j) \log(1-\pi_j)) \tag{2}$$

where π_j is given by (1). MLEs, $\hat{\beta}_i$, are obtained by maximizing l with respect to $\boldsymbol{\beta}$. Let $\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_k)^t$ be the maximizer of l. Newton-Raphson's algorithm is traditionally used to solve the resulting *nonlinear* equations. Other methods [10] are gradient ascent, coordinate ascent, conjugate gradient ascent, fixed-Hessian Newton, quasi-Newton algorithms (DFP and BFGS), iterative scaling, Nelder-Mead and random integration.

2.2 Regularized Approaches to Logistic Regression

Ridge logistic regression seeks MLEs subject to spherical restrictions on the parameters. Therefore, the function to be maximized is the penalized log-likelihood given by

$$l^*(\boldsymbol{\beta}) = l(\boldsymbol{\beta}) - \frac{\lambda}{2} \sum_{i=1}^{k} \beta_i^2, \tag{3}$$

where $\lambda > 0$ is the penalty parameter and controls the amount of shrinkage. λ is usually chosen by cross-validation. The cross-validation deviance, error, BIC or AIC are used as the criteria to be optimized. Let $\hat{\boldsymbol{\beta}}^*$ be the maximizer of Equation 3 or ridge estimator. This estimator always exists and is unique.

In the field of microarray classification, Newton-Raphson's algorithm may be employed but it requires a matrix of dimension $k + 1$ to be inverted. Inverting huge matrices may be avoided to some extent with algorithms like the dual algorithm based on sequential minimal optimization [11] or SVD [12]. Combined with SVD, [13, 14] use a feature selection method called *recursive feature elimination* (RFE) [15] that iteratively removes genes with smaller absolute values of $\hat{\beta}_i^*$.

Within a broader context, log-likelihood can be penalized as $l(\boldsymbol{\beta}) - \frac{\lambda}{2} J(\boldsymbol{\beta})$, where the penalty function is generally $J(\boldsymbol{\beta}) = \sum_i \gamma_i \psi(\beta_i)$, $\gamma_i > 0$. The L_1 penalty $\psi(\beta_i) = |\beta_i|$ results in *lasso*, introduced by [16] in the context of logistic regression. In a Bayesian setting, the prior corresponding to this case is an independent Laplace distribution (or double exponential) for each β_i. Cawley and Talbot [17] even model the penalty parameter λ by using a Jeffreys' prior to eliminate this parameter by integrating it out analytically. Although the objective function is still concave in lasso (as in ridge regression), an added

computational problem is that this function is not differentiable. Generic methods for nondifferentiable concave problems, such as the ellipsoid method or subgradient methods, are usually very slow in practice. Faster methods have recently been investigated [18, 19]. Interest in lasso is growing because L_1 penalty encourages the estimators be either significantly large or exactly zero, which has the effect of automatically performing feature selection and hence yielding concise models.

2.3 EDAs for Regularizing Logistic Regression-based Microarray Classifiers

Among the stochastic population-based search methods, EDAs have recently emerged as a general framework that overcomes some weaknesses of other well-known methods like genetic algorithms [9]. Unlike genetic algorithms, EDAs avoid the ad hoc design of crossover and mutation operators, as well as the tuning of a large number of parameters, while they explicitly capture the relationships among the problem variables by means of a joint probability distribution (jpd). The main system underlying the EDA approach, which will be denoted *Proc-EDA*, is:

1. $D_0 \leftarrow$ Generate M points of the search space randomly
2. $h = 1$
3. **do** {
4. $D_{h-1}^{Se} \leftarrow$ Select $M' < M$ points of the search space from D_{h-1}
5. $p_h(\mathbf{z}) = p(\mathbf{z} \mid D_{h-1}^{Se}) \leftarrow$ Estimate the jpd from the selected points of the search space
6. $D_h \leftarrow$ Sample M points of the search space (the new population) from $p_h(\mathbf{z})$
7. } **until** a stopping criterion is met

M points of the search space constitute the initial population and are generated at random. All of them are evaluated by means of a fitness function (step 1). Then, $M'(M' < M)$ points are selected according to a selection method, taking the fitness function into account (step 4). Next, a multidimensional probabilistic model that reflects the interdependencies between the encoded variables in these M' selected points is induced (step 5). The estimation of this underlying jpd represents the EDA bottleneck, as different degrees of complexity in the dependencies can be considered. In the next step, M new points of the search space – the new population – are obtained by sampling from the multidimensional probabilistic model learnt in the previous step (step 6). Steps 4 to 6 are repeated until some pre-defined stopping condition is met (step 7). Likewise other numerical methods (see above) as Nelder-Mead's, EDAs work by simply evaluating the objective function at some points. However, Nelder-Mead's algorithm is deterministic and evaluates the vertices of a simplex, while EDAs are stochastic, require a population and to learn/simulate models.

If we confine ourselves to logistic regression classifiers, EDAs have been used for estimating the parameters from a multiobjective viewpoint [20]. EDAs could be successfully used to optimize *any kind* of penalized likelihood because, unlike traditional numerical methods, they do not require derivative information or matrix inversions. However, we investigate here a more interesting approach that shows that EDAs can act as an intrinsic regularizer if we choose a suitable representation. Thus, let us take $l(\boldsymbol{\beta})$ (▶see Eq. 2) as the fitness function that assesses each possible solution $\boldsymbol{\beta}$ to the (unrestricted) maximum likelihood problem. $\boldsymbol{\beta}$ is a $k+1$ dimensional *continuous* random variable. EDAs would start by randomly generating the initial population D_0 of M points of the search space $\boldsymbol{\beta}_1^{(0)},\ldots,\boldsymbol{\beta}_M^{(0)}$. After selecting M' points (e.g. the top M'), the core of the EDA paradigm is step 5 above to estimate the jpd from these selected M' points. Without losing generality, we start from a univariate marginal distribution algorithm (UMDA$_c^G$) [21] in our continuous $\boldsymbol{\beta}$-domain. UMDA$_c^G$ assumes that at each generation h all variables are independent and normally distributed, i.e.

$$p_h(\boldsymbol{\beta}) = \prod_{i=0}^{k} p_h(\beta_i) = \prod_{i=0}^{k} \frac{1}{\sigma_{ih}\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\beta_i - \mu_{ih}}{\sigma_{ih}}\right)^2} \quad (4)$$

See [22] for the UMDA$_c^G$ theoretical support. We now modify UMDA$_c^G$ to tackle the regularized logistic regression by shrinking the β_i parameters during the EDA simulation step. Specifically, we introduce a new algorithm UMDA$_c^{G*}$ that learns a UMDA$_c^G$ model given by (4) at step 5 and iteration h. This involves estimating the new μ_{ih} and σ_{ih} with the MLEs computed on the selected set D_{h-1}^{Se} of M' points of the search space from the previous generation. However, sampling at step 6 now generates points from (4) with the normal distributions $p_h(\beta_i)$ constrained to lie in an interval $[-b_h, b_h]$. This is readily achieved by generating values from a Gaussian of parameters μ_{ih} and σ_{ih} for each variable β_i and constraining its outputs, according to a standard rejection method to fall within $[-b_h, b_h]$.

The idea is that, as long as the algorithm progresses, forcing the β_i parameters to be in a bounded interval around 0 constrains and stabilizes their values, just like regularization does. At step 5, we learn, for the random variable $\boldsymbol{\beta}$, the multivariate Gaussian distribution with a diagonal covariance matrix that best fits, in terms of likelihood, the M' $\boldsymbol{\beta}$-points that are top ranked in the objective function $l(\boldsymbol{\beta})$. We then generate, at step 6, M new points from the previous distribution truncated at each coordinate at $-b_h$ (bottom) and at b_h (top). New data are ranked with respect to their $l(\boldsymbol{\beta})$ values, and the best M' are chosen and so on. In spite of optimizing function $l(\boldsymbol{\beta})$ rather than another penalized log-likelihood function like e.g. ridge regression's $l^*(\boldsymbol{\beta})$, the evolutionary process guarantees that the β_i's values belong to intervals of the desired size. Therefore, our estimates of β_i are regularized estimates. In fact, we have empirically verified that the standard errors of our estimators are smaller than those of regularized approaches like ridge logistic regression and exhibiting less outliers than lasso. Moreover, since we use the original $l(\boldsymbol{\beta})$ objective function of the logistic regression, we do not need to specify the λ parameter of other penalized approaches like (3).

Note that plenty of probability models are possible in (4), without necessarily assuming all variables to be Gaussian and independent. Different univariate, bivariate or multivariate dependencies may be designed with the benefit of having an explicit model of (possible) complex probabilistic relationships among the different parameters. Traditional numerical methods are unable to provide this kind of information.

Thus, the estimation of Gaussian network algorithm (EGNA) [21] models multivariate dependencies among β_i by learning at each generation a nonrestricted normal density that maximizes the Bayesian information

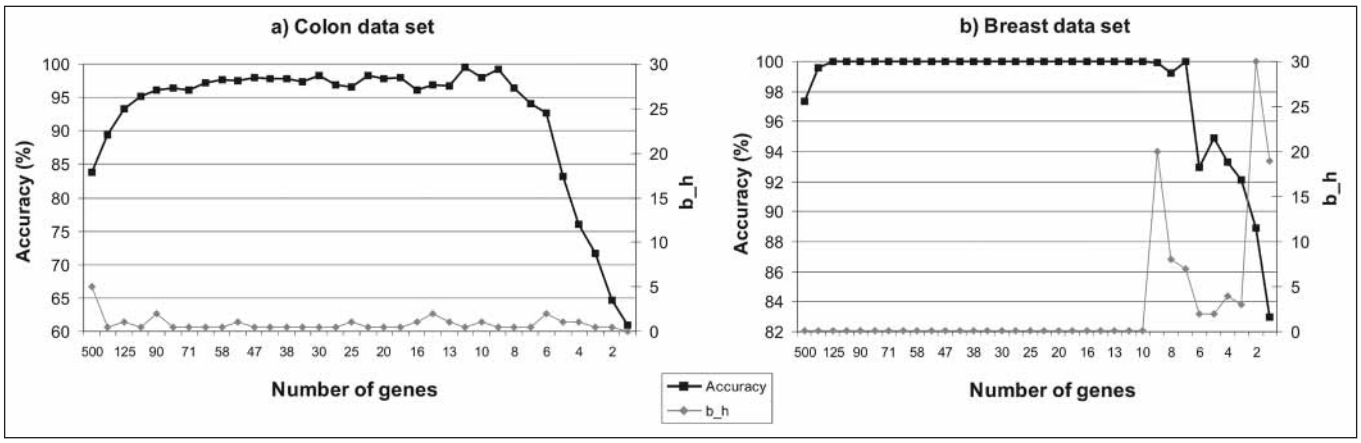

Fig. 1 Number of genes in set S vs. accuracy (%) and vs. b_h^{op} for *Breast* and *Colon* data sets

criteria (BIC) score. In EGNA, $p_h(\boldsymbol{\beta})$ factorizes as a Gaussian network [23]. The rationale for this assumption is in part justified by the fact that the MLEs asymptotically follow a multivariate normal distribution. However, in our case the number of observations N is small and, as mentioned above, we do not have MLEs either since our estimators are restricted MLEs.

Finally, the last step, say at iteration $h = T$, would contain $\boldsymbol{\beta}_1^{(T)},\ldots,\boldsymbol{\beta}_M^{(T)}$ from which $argmax_{j\in\{1,\ldots,M\}} l(\boldsymbol{\beta}_j^{(T)})$ would be chosen as the final regularized estimate of $\boldsymbol{\beta}$.

2.4 Gene Selection

Our EDA-based regularization is now embedded in a gene selection procedure. We propose it to take into account the strength of each gene i given by its regression coefficient β_i and besides to automatically search for an optimal b_h according to the classification accuracy of the associated regularized model. The general procedure, denoted *Proc-gene*, is:

1. For a subset of genes S, search for b_h of EDA approach using the classification accuracy as the criterion. Let b_h^{op} be the optimal value.
2. With b_h^{op} fixed, eliminate a percentage of the genes with the smallest β_t^2 values. Let S be the new (smaller) set of genes.
3. Repeat steps 1 and 2 until there is only one gene left. An optimal subset of genes is finally derived.

Some remarks follow. In step 1, subset S to initialize the process may be chosen in different ways. Basically, we can start with all the genes or we can use a filter approach to reduce the size of this subset. Since it is not clear which filter criterion to use and different filter criteria may lead to different conclusions, we propose here a kind of consensus among different filter criteria. Thus, for four filters f_1, f_2, f_3 and f_4, if gene i is ranked first by f_1, second by f_2, third by f_3 and fourth by f_4, then its rank aggregation would be 11. The top-ranked genes by this new agreement would be chosen. In our experiments we have used the following four filter criteria: 1) the BSS/WSS criterion (as in [24]), 2) the Pearson correlation coefficient to the class variable (as in [5, 25]), 3) a *p-metric* (as in [26]), and 4) a *t-score*.

The search for the optimal b_h for the EDA in step 1 amounts to running EDA (*Proc-EDA*) several times (for different b_h values) and measuring which of the fitted logistic regression models is the best. This is assessed by estimating the classifier's accuracy (percentage of correctly classified microarrays) as the generalization performance of the model. Braga-Neto and Dougherty [27] proved the *.632 bootstrap* estimator to be a good overall estimator in small-sample microarray classification, and it was therefore the chosen method in this paper.

In step 2 of *Proc-gene*, EDA has already provided a fitted model (with the best b_h value) and then a gene selection method inspired by RFE is carried out. As in [13, 14], we remove more than one feature at a time for computational reasons (the original RFE only removes one), based on the smallest β_t^2 values, indicators of a lower relative importance in the gene subset.

3. Results and Discussion

We illustrate how our approach really acts as a regularizer on some publicly available benchmark microarray data sets. First, the *Breast* data set [25] with 7129 genes and 49 tumor samples, 25 of them representing estrogen receptor-positive (ER+) and the other 24

Table 1 Selected top 7 genes with their β estimate for *Breast*

GenBank ID [ref.]	Description	β
X87212_at [25]	*H. sapiens* mRNA for cathepsin C	–6.988
L26336_at [32]	Heat shock 70kDa protein 2	6.980
L17131_ma1_at [16, 33]	Human high mobility group protein	–5.402
J03827_at	Y box binding protein-1 mRNA	–3.549
S62539_at [34]	Insulin receptor substrate 1	3.419
HG4716-HT5158_at [35]	Guanosine 5'-monophosphate synthase	–2.685
U30827_s_at [25, 36]	Splicing factor, arginine/serine-rich 5	2.480

Table 2 Selected top 9 genes with their β estimate for *Colon*

GenBank ID [ref.]	Description	β
T94579 [38]	Human chitotriosidase precursor mRNA, complete cds	–0.500
D26129 [40]	Ribonuclease pancreatic precursor (human)	–0.500
T40578 [39]	Caldesmon 1	–0.499
R80427 [38]	C4-dicarboxylate transport sensor protein dctb (Rhizobium leguminosarum)	–0.497
Z50753 [38]	H.sapiens mRNA for GCAP-II/uroguanylin precursor	0.496
M76378 [38]	Human cysteine-rich protein (CRP) gene, exons 5 and 6	0.494
H06061 [38]	Voltage-dependent anion-selective channel protein 1 (Homo sapiens)	0.485
H08393 [38]	Collagen alpha 2(XI) chain (Homo sapiens)	0.482
T62947 [38]	60S ribosomal protein L24 (Arabidopsis thaliana)	–0.480

being estrogen receptor-negative (ER–). Second, the *Colon* data set [28] that contains 2000 genes for 62 tissue samples: 40 cancer tissues and 22 normal tissues. Other public data sets have been studied: the *Leukemia* data set [29] and the *Prostate* cancer data set [30]. See the supplementary material on the web page[a].

We have developed our own implementation in C++ for the EDA-based regularized logistic regression (*Proc-EDA*) and in R for the gene selection method (*Proc-gene*) that calls the former. We tried two different EDA approaches: UMDA_c^G and EGNA. To run EDAs we found that an initial population of at least $M = 100$ points and of at least $M' = 50$ selected points for learning guarantee robust β estimates. The relative change in the mean fitness value between successive generations was the chosen value for assessing the convergence of the *Proc-EDA* algorithm.

As regards *Proc-gene*, we considered reasonable to initialize it with 500 genes for the size of subset S. These were selected according to the aggregation of the four filter criteria as described above. Based on our experience, a good choice in the experiments for the number of bootstrap samples used for training was 100. The percentage of genes to be removed in step 2 is fixed as 10%.

▶Figure 1a and ▶Table 1 show the experimental results on the *Breast* data set. Since perfect classification (100%) is achieved with many different gene subsets, we choose the subset with fewer genes, i.e. the 7-gene model. Note how b_h^{op} obtained at step 1 of procedure *Proc-gene* varies as long as the number of selected genes changes due to the adapted RFE. Its minimum value is 0.5. Running times on an Intel Xeon 2GHz under Linux are quite acceptable: almost 3 minutes for 500 genes, 39 s for 250, between 2.5 and 5 s for 75–125 genes, and less than 2 s for 70 genes or fewer.

The seven genes found to separate ER+ from ER– samples achieve a higher classification accuracy than other up-to-date regularized methods. Shevade and Keerthi [16] report an accuracy of 81.9% and use logistic regression with L_1 penalty solved by the Gauss-Seidel method. They propose a different gene selection procedure and retain six genes, two of them also found by us (see below). Fort and Lambert-Lacroix [31] use a combination of PLS and ridge logistic regression to achieve an about 87.5% accuracy. They perform a gene selection based on the BSS/WSS criterion choosing some fixed number of genes: 100, 500, 1000, although they do not indicate which are they. Finally, a slightly different approach followed by the original paper by West et al. [25], where a probit (binary) regression model is combined with a stochastic regularization and SVDs, yields a 89.4% accuracy using 100 genes selected according to their Pearson correlation coefficient to the class variable. When our results are compared to the most popular regularization methods, lasso and ridge logistic regressions only achieve 98.23% and 98.46% accuracies, respectively, using in both cases the same 500 selected genes provided by the aggregation of the four filter criteria. All of our seven selected genes have been linked with breast cancer proving the consistency of our results with the literature (see Table 1).

▶Figure 1b and ▶Table 2 show the results on the *Colon* data set. Classes are less well separated outputting at most a 99.65% accuracy, for the 9-gene model. Running times are longer than before: almost 10 minutes for 500 genes, 1.5 minutes for 250, between 2 and 7 s for 60–125 genes, and less than 2 s for 55 genes or fewer.

An analysis of the selected genes and the accuracy reported by other directly related methods is as follows. Shevade and Keerthi [16] achieve an accuracy of 82.3% with eight genes, three of them – Z50753, T62947 and H08393 – included in our list. Liu et al. [37] use logistic regression with L_p penalty, where $p = 0.1$ and retain 12 genes. Genes Z50753, M76378 and H08393 of their list are also in ours. They do not compute the accuracy but the AUC (0.988), which in our case for the 9-gene model is better (0.9996). Using a ridge logistic regression approach, Shen and Tan [14] keep 16 genes with a similar RFE than in our case and report a 99.3% accuracy, without any mention to the specific genes selected. When our results are compared to lasso and ridge logistic regressions, these only achieve 89.74% and 90.51% accuracies, respectively, both lower than our 99.65% accuracy. Our 9-gene list includes genes identified as relevant for colon cancer in the literature (see Table 2).

See the supplementary material for details on EGNA factorizations.

4. Conclusions

The high interest of combining a regularization with a dimension-reduction step to enhance classifier efficiency has been pointed out elsewhere [31]. Combined with a gene subset selection procedure that adapts the RFE and automatically tunes the required parameters, we have introduced a novel EDA-based logistic regression regularizer. It includes the shrinkage of the coefficients implicitly during EDA evolution process while optimizing the usual likelihood function. The

[a] http://laurel.datsi.fi.upm.es/~vrobles/eda_lr_reg

empirical results on several microarray data sets have provided models with a low number of relevant genes, most of them confirmed by the literature, and performing better in classification than other competing regularized methods.

Unlike the traditional procedures for finding maximum likelihood β_i parameters, the EDA approach is able to use any optimization objective, regardless of its complexity or the non-existence of an explicit formula for its expression. In this respect, our framework could find parameters that maximize the AUC objective (a difficult problem [41]) or it would also fit the search for parameters of any regularized logistic regression. The inclusion of interaction terms among (possibly co-regulated) genes in η_j of expression (1) would also be feasible as other future direction to explore.

Acknowledgments

The authors are grateful to the referees for their constructive comments. Work partially supported by the Spanish Ministry of Education and Science, projects TIN2007-62626, TIN2007-67148 and TIN2008-06815-C02 and Consolider Ingenio 2010-CSD2007-00018 and by the National Institutes of Health (USA), project 1 R01 LM009520-01.

References

1. Larrañaga P, Calvo B, Santana R, Bielza C, Galdiano J, Inza I, Lozano JA, Armañanzas R, Santafé G, Pérez A, Robles V. Machine learning in bioinformatics. Briefings in Bioinformatics 2006; 17 (1): 86–112.
2. Dugas M, Weninger F, Merk S, Kohlmann A, Haferlach T. A generic concept for large-scale microarray analysis dedicated to medical diagnostics. Methods Inf Med 2006; 45 (2): 146–152.
3. Hosmer DW, Lemeshow S. Applied Logistic Regression. 2nd edn. New York: J. Wiley and Sons; 2000.
4. Thisted RA. Elements of Statistical Computing. New York: Chapman and Hall; 1988.
5. Markowetz F, Spang R. Molecular diagnosis classification, model selection and performance evaluation. Methods Inf Med 2005; 44 (3): 438–443.
6. Weber G, Vinterbo S, Ohno-Machado L. Multivariate selection of genetic markers in diagnostic classification. Artif Intell Med 2004; 31: 155–167.
7. Heckerling PS, Gerber BS, Tape TG, Wigton R. Selection of predictor variables for pneumonia using neural networks and genetic algorithms. Methods Inf Med 2005; 44 (1): 89–97.
8. Lee A, Silvapulle M. Ridge estimation in logistic regression. Comm Statist Simulation Comput 1988; 17: 1231–1257.
9. Lozano JA, Larrañaga P, Inza I, Bengoetxea E (eds). Towards a New Evolutionary Computation. Advances in Estimation of Distribution Algorithms. New York: Springer; 2006.
10. Minka T. A comparison of numerical optimizers for logistic regression. Tech Rep 758, Carnegie Mellon University; 2003.
11. Keerthi SS, Duan KB, Shevade SK, Poo AN. A fast dual algorithm for kernel logistic regression. Mach Learning 2005; 61: 151–165.
12. Eilers P, Boer J, van Ommen G, van Houwelingen H. Classification of microarray data with penalized logistic regression. In: Proc of SPIE. Progress in Biomedical Optics and Images, 2001. Volume 4266 (2): 187–198.
13. Zhu J, Hastie T. Classification of gene microarrays by penalized logistic regression. Biostatistics 2004; 5: 427–443.
14. Shen L, Tan EC. Dimension reduction-based penalized logistic regression for cancer classification using microarray data. IEEE Trans Comput Biol Bioinformatics 2005; 2: 166–175.
15. Guyon I, Weston J, Barnhill S, Vapnik V. Gene selection for cancer classification using support vector machines. Mach Learning 2002; 46: 389–422.
16. Shevade SK, Keerthi SS. A simple and efficient algorithm for gene selection using sparse logistic regression. Bioinformatics 2003; 19: 2246–2253.
17. Cawley GC, Talbot N. Gene selection in cancer classification using sparse logistic regression with Bayesian regularization. Bioinformatics 2006; 22: 2348–2355.
18. Koh K, Kim SY, Boyd S. An interior-point method for large-scale L1-regularized logistic regression. J Mach Learn Res 2007; 8: 1519–1555.
19. Krishnapuram B, Carin L, Figueiredo M, Hartemink A. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. IEEE Trans Pattern Anal Mach Intell 2005; 27: 957–968.
20. Robles V, Bielza C, Larrañaga P, González S, Ohno-Machado L. Optimizing logistic regression coefficients for discrimination and calibration using estimation of distribution algorithms. TOP 2008; 16: 345–366.
21. Larrañaga P, Etxeberria R, Lozano JA, Peña JM. Optimization in continuous domains by learning and simulation of Gaussian networks. In: Workshop in Optimization by Building and Using Probabilistic Models. Genetic and Evolutionary Computation Conference, GECCO 2000. pp 201–204.
22. González C, Lozano JA, Larrañaga P. Mathematical modelling of UMDAc algorithm with tournament selection. Behaviour on linear and quadratic functions. Internat J Approx Reason 2002; 31: 313–340.
23. Shachter R, Kenley C. Gaussian influence diagrams. Manag Sci 1989; 35: 527–550.
24. Dudoit S, Fridlyand J, Speed TP. Comparison of discrimination methods for the classification of tumors using gene expression data. J Am Stat Assoc 2002; 97: 77–87.
25. West M, Blanchette C, Dressman H, Huang E, Ishida S, Spang R, Zuzan H, Olson JA, Marks JR, Nevins JR. Predicting the clinical status of human breast cancer by using gene expression profiles. Proc Natl Acad Sci USA 2001; 98 (20): 11462–11467.
26. Inza I, Larrañaga P, Blanco R, Cerrolaza A. Filter versus wrapper gene selection approaches in DNA microarray domains. Artif Intell Med 2004; 31: 91–103.
27. Braga-Neto UM, Dougherty ER. Is cross-validation valid for small-sample microarray classification? Bioinformatics 2004; 20: 374–380.
28. Alon U et al. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide microarrays. Proc Natl Acad Sci USA 1999; 96: 6745–6750.
29. Golub TR et al. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science 1996; 286: 531–537.
30. Singh D, Febbo PG, Ross K, Jackson DG, Manola J, Ladd C, Tamayo P, Renshaw AA, D'Amico AV, Richie JP, Lander ES, Loda M, Kantoff PW, Golub TR, Sellers WR. Gene expression correlates of clinical prostate cancer behavior. Cancer Cell 2002; 1: 203–209.
31. Fort G, Lambert-Lacroix S. Classification using partial least squares with penalized logistic regression. Bioinformatics 2005; 21: 1104–1111.
32. Rohde M, Daugaard M, Jensen MH, Helin K, Nylandsted J, Marja Jaattela M. Members of the heat-shock protein 70 family promote cancer cell growth by distinct mechanisms. Genes Dev 2005; 19: 570–582.
33. Chiappetta G, Botti G, Monaco M, Pasquinelli R, Pentimalli F, Di Bonito M, D'Aiuto G, Fedele M, Iuliano R, Palmieri EA, Pierantoni GM, Giancotti V, Fusco A. HMGA1 protein overexpression in human breast carcinomas: Correlation with ErbB2 expression. Clin Cancer Res 2004; 10: 7637–7644.
34. Sisci D, Morelli C, Garofalo C, Romeo F, Morabito L, Casaburi F, Middea E, Cascio S, Brunelli E, Ando S, Surmacz E. Expression of nuclear insulin receptor substrate 1 in breast cancer. J Clin Pathol 2007; 60: 633–641.
35. Turner GA, Ellis RD, Guthrie D, Latner AL, Monaghan JM, Ross WM, Skillen AW, Wilson RG. Urine cyclic nucleotide concentrations in cancer and other conditions; cyclic GMP: A potential marker for cancer treatment. J Clin Pathol 2004; 35 (8): 800–806.
36. Abba MC, Drake JA, Hawkins KA, Hu Y, Sun H, Notcovich C, Gaddis S, Sahin A, Baggerly K, Aldaz CM. Transcriptomic changes in human breast cancer progression as determined by serial analysis of gene expression. Breast Cancer Res 2004; 6: 499–513.
37. Liu Z, Jiang F, Tian G, Wang S, Sato F, Meltzer SJ, Tan M. Sparse logistic regression with Lp penalty for biomarker identification. Statistical Applications in Genetics and Molecular Biology 2007; 6: Article 6.
38. Furlanello C, Serafini M, Merler S, Jurman G. Entropy-based gene ranking without selection bias for the predictive classification of microarray data. BMC Bioinform 2003; 4: 54.
39. Gardina PJ. Alternative splicing and differential gene expression in colon cancer detected by a whole genome exon array. BMC Genomics 2006; 7: 325.
40. Lin YM, Furukawa Y, Tsunoda T, Yue CT, Yang KC, Nakamura Y. Molecular diagnosis of colorectal tumors by expression profiles of 50 genes expressed differentially in adenomas and carcinomas. Oncogene 2002; 21: 4120–4128.
41. Ma S, Huang J. Regularized ROC method for disease classification and biomarker selection with microarray data. Bioinformatics 2005; 21: 4356–4362.