Neural Process Lett (2011) 34:117–131
DOI 10.1007/s11063-011-9187-8

Neuro-logistic Models Based on Evolutionary Generalized Radial Basis Function for the Microarray Gene Expression Classification Problem

A. Castaño · F. Fernández-Navarro · C. Hervás-Martínez · P. A. Gutierrez

Published online: 3 June 2011

Abstract Gene expression detection is a key bioinformatic problem which has been tackled as a classification problem of microarray gene expression, obtained by the light reflection analysis of genomic material. A typical microarray dataset may contain thousands of genes but only a small number of patterns (often less than two hundred). When the dataset presents these kinds of characteristics, state-of-the-art classification models show a high lack of performance. A two-stage algorithm has been proposed to successfully address the problem of microarray classification. In the first stage, two filter algorithms identify salient expression genes from thousands of genes. In the second stage, the proposed methodology is performed using selected gene subsets as new input variables. The methodology proposed is composed of a combination of Logistic Regression (LR) and Evolutionary Generalized Radial Basis Function (EGRBF) neural networks which have shown to be highly accurate in previous research in the modeling of high-dimensional patterns. Finally, the results obtained are contrasted with nonparametric statistical tests and confirm good synergy between EGRBF and LR models.

Keywords Evolutionary algorithm · Generalized radial basis function · Logistic regression · Neural networks · Classification · Microarray gene expression

A. Castaño
Department of Informatics, University of Pinar del Río, Pinar del Río, Cuba
e-mail: adiel@info.upr.edu.cu

F. Fernández-Navarro (✉) · C. Hervás-Martínez · P. A. Gutierrez
Department of Computer Science and Numerical Analysis, University of Córdoba, Córdoba, Spain
e-mail: i22fenaf@uco.es

C. Hervás-Martínez
e-mail: chervas@uco.es

P. A. Gutierrez
e-mail: pagutierrez@uco.es

1 Introduction

The analysis of Deoxyribo Nucleic Acid (DNA) allows researchers to explore the seeds of life. It has also shown how the composition of their sequences produces changes in the phenotype of living organisms. An important technique for analyzing the expression of genetic material is microarray dataset analysis.

A microarray dataset contains the data of intense light reflected on genetic material. The microarray production process is composed of different steps, roughly described as: target preparation, hybridization, slide scanning, data analysis and expression profile clustering. This process converts DNA sequences into light intensity value records, labeled in certain classes. During the microarray creation process, the genetic material of dissimilar laboratory tissue patterns undergoes the above processes. In the final stage, the process yields a sheet of pre-processed genetic material which is excited by laser. Then the image is gridded with a template and the intensities of the features (several pixels make up a feature) are quantified. Later the raw data is normalized and the dataset is elaborated. The resultant dataset has a great number of features whose classification cannot be managed by state-of-the-art algorithms due to the great dimension of the problem. This paper evaluates an alternative for handling microarray datasets, combining feature selection with a novel neurologistic method to improve the classification performance of these genomic datasets.

Our proposal is a combination of the Evolutionary Generalized Radial Basis Function (EGRBF) and Logistic Regression(LR) methods. The LR methods, apply a logit function to the linear combination of input variables. The coefficient values of each input variable are estimated by means of the Iterative Reweighted Least Square (IRLS) algorithm. Roughly, the methodology is divided into three steps. Firstly, an Evolutionary Algorithm (EA) is applied to estimate the parameters of the GRBF. Secondly, the input space is increased by adding the nonlinear transformation of the input variables given by the GRBFs of the best individual in the last generation of the EA. Finally, the LR algorithm is applied in this new covariate space. The first two steps are devoted to exploring the model parameter space, and to obtaining a promising initial solution. After that, the IRLS algorithm (exploiter algorithm) is performed to estimate the final parameters of the model, in a way similar to how hybrid algorithms perform [2]. It is important to note that a first proposal of a combination of neural networks and LR is given in two recent studies [19,20]. The model is based on the hybridization of a linear multilogistic regression model and a nonlinear Product Unit Neural Network (PUNN) model for binary and multiclass classification problems. This paper considers GRBFs (which are local approximators) for the nonlinear part of the model, while the previously proposed method is based on Product Units (PUs) (which are global approximators).

RBFs are widely used in real life problems involving face detection [21], channel equalization [24], or predictive microbiology [9]. Due to the known ability of RBF to fit variable interactions, some theoretical advances have been proposed to train models based on RBFs [15,23,32,33]. However, the Standard Gaussian RBF (SRBF) has some drawbacks. For example, it decreases its performance when dimensionality grows. In high dimensional space, the patterns are concentrated far from the cluster center [13], making SRBF assign similar belonging values to these patterns. However, the novel RBF analyzed (GRBF) also adds a τ shape parameter to SRBF. This parameter allows the curvature to adapt, making the GRBF assign significantly different activation values to the patterns located near the boundaries of the cluster.

The original formulation of the GRBF [13] presents two problems: the first one, inherited from SRBF, is that radii variation produces changes in the curvature of the basis functions. The second one, concerns the fact that the same modification in the exponent leads to

different curvatures values, depending on the original value of the exponent. To tackle these two problems, a reformulation of the GRBF is proposed in sections of this paper.

To avoid dimensionality problems, two feature selectors are applied in the preprocessing stage: the Fast Correlation-Based Filter [31] (FCBF) and Best Agglomerative Ranked Subset [27] (BARS). The motivation for applying feature selection techniques has shifted from being optional to becoming a real prerequisite for model building. Theoretically, having more genes should give us more discriminating power. However, this can cause several problems: increased computational complexity and cost; too many redundant or irrelevant genes; and estimation degradation in the classification error. The resulting datasets have fewer features (between 150 and 250 features) but can still be seen as datasets of high dimensionality. Therefore, these datasets have been considered adequate to assess the effectiveness of our proposal.

This paper is organized as follows: Sect. 2 formally presents the GRBF model considered in this work. Section 3 describes a reformulation of the original GRBF. Section 4 introduces the neurologistic model used in this paper. Afterwards, the neurologistic model-fitting process is described in Sect. 5. Section 6 describes the experiments carried out and discusses the results obtained. Finally, Sect. 7 completes the paper with the main conclusions and future directions suggested by this study.

2 Generalized Radial Basis Function

The original Generalized Radial Basis Function (GRBF) was proposed by Francois [13]. Recently different approaches have been introduced to estimate the parameters of this novel model [5,12]. The GRBF is defined by replacing the square power in the exponent of the SRBF by the τ parameter:

$$B_j(\mathbf{x}, \mathbf{w}_j) = \exp\left(-\left(\frac{\|\mathbf{x} - \mathbf{c}_j\|}{r_j}\right)^{\tau_j}\right), \tag{1}$$

where $\mathbf{w}_j = (\mathbf{c}_j, r_j, \tau_j)$, $\mathbf{c}_j = (c_{j1}, c_{j2}, \ldots, c_{jk})$ is the center of the cluster represented by j-th GRBF transformation, r_j is the corresponding radii or standard deviation, τ_j is the exponent of the basis function, and $c_{ji}, r_j, \tau_j \in \mathbb{R}$. Figure 1 presents the activation for the GRBF with different values of τ. The incorporation of the τ parameter causes the contraction-relaxation of the GRBF curvature. Thanks to the additional τ parameter, the GRBF can define more accurately the membership of the patterns that are located near the decision boundary between clusters. In particular GRBFs have been analyzed in high-dimensional problems. Then we will justify why these basis functions are especially accurate in this kind of problems.

One problem in high dimensional spaces is that of distances *concentrate*: the range of possible distances is not fully spanned anymore, and most of the patterns are very far from one another [3]. Figure 2 illustrates this situation for a set of 200 patterns drawn from normal distribution, in a two-dimensional space on the left and in a 100-dimensional space on the right. For the two-dimensional case, distances exist in the whole possible [0, 4] range, while in dimension 100, only a very small part of the histogram is filled, mostly with great distances (in the range [8, 12]). Therefore, the probability of finding patterns near the center, when the dimension is high, is almost zero.

Additionally, the problem of the concentration of distances is justified by taking into account that the probability density function of finding a point at distance q from the center of the distribution is given by [13]:

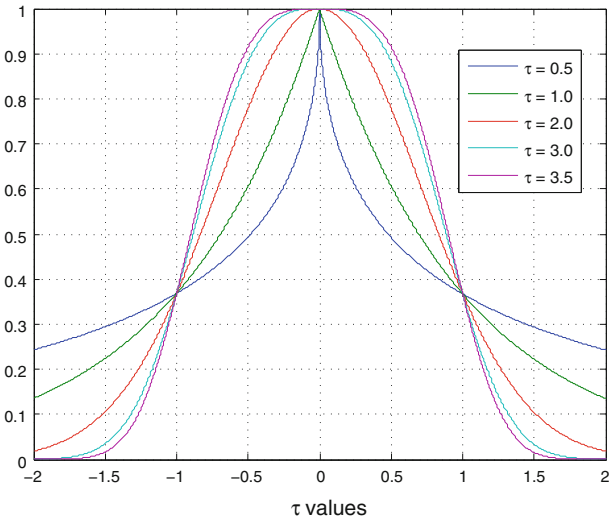

Fig. 1 GRBF activation in one-dimensional space with $c = 0$ and $r = 1$ for different τ values

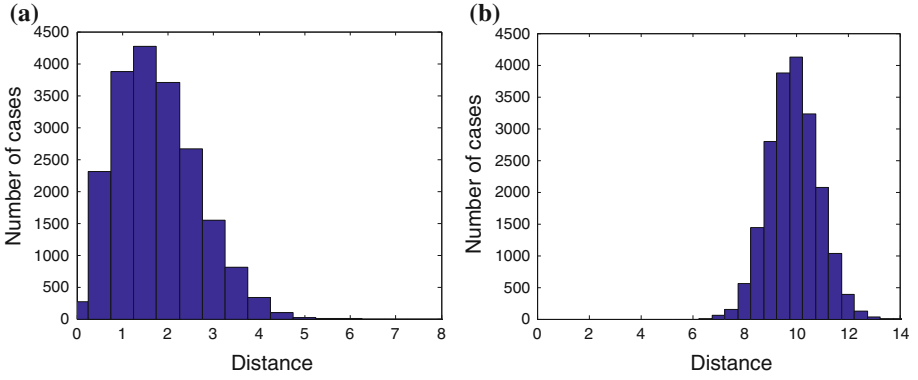

Fig. 2 Illustration of the concentration of distances effect for a set of normally distributed patterns: histogram of the pairwise distances between normally drawn patterns, in dimension 2 (*left*) and 100 (*right*)

$$f(q,d) = \frac{q^{d-1}}{2^{(d/2)-1}} \cdot \frac{e^{-q^2/2}}{\Gamma(d/2)} (\text{for } \sigma = 1, q > 0 \text{ and } d > 0), \tag{2}$$

where d is the data dimension. Figure 3a represents the probability density function of finding a point from a normal distribution at a distance q for $d = 2$ and 100, respectively. Additionally, Fig. 3b shows a graphic illustration of why GRBF quantifies similarities better than the SRBF in high dimensional spaces: red and cyan lines represent SRBFs centered at the origin with $r = 1.6$ and 10.6 respectively and the magenta line shows a GRBF centered at the origin with $\tau = 13$ and $r = 10.6$. As can be observed in Fig. 3b when $d = 2$, the SRBF shows its ability to fit distance distribution, assigning membership values in the interval $[0, 1]$ (see red line); however when $d = 100$, the SRBF assigns membership values in the interval $[0.27, 0.57]$ (see cyan line) because for this dimension the SRBF needs a high value of r to include these patterns. The problem is that the SRBF has a slightly pronounced curvature

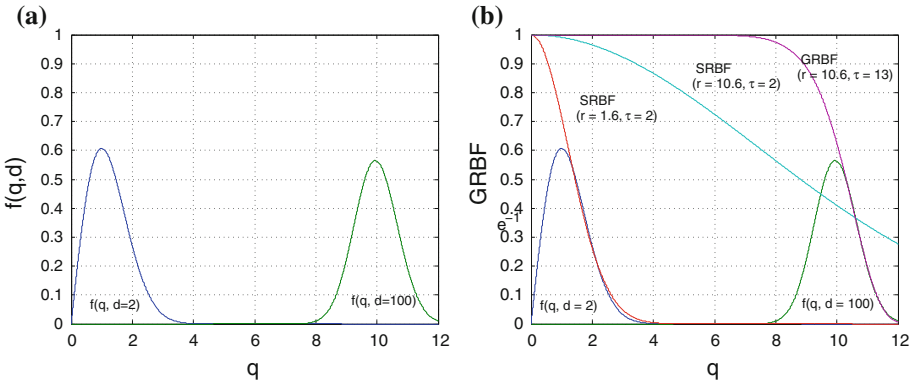

Fig. 3 RBF values as a function of the distance to their centers for two space dimensions ($d = 2$ and 100), along with the distribution of distances for normally distributed data ($f(q, d)$)

when it has a high value of r. On the other hand, the GRBF assigns membership values in the interval [0, 1] even in high dimensional spaces (see magenta line). In our opinion, this justifies considering GRBF as a suitable kernel for quantifying similarity in high dimensional spaces.

However, the original formulation of the GRBF based on the τ parameter has some drawbacks in that the same variation in the τ value produces different effects on the GRBF curvature. Figure 1 shows that an increase of 0.5 in the τ value ($\Delta\tau = +0.5$) when $\tau = 1$ produces a significant variation in the GRBF curvature, although when $\tau = 3$, an increase of 0.5 barely modifies GRBF curvature.

Furthermore, the τ parameter causes different curvatures for different GRBF radii values. This instability on the contraction-relaxation of the GRBF for different r values makes the convergence very difficult when using stochastic or gradient-based algorithms to directly optimize the τ value.

3 Our Proposal: Reformulation of the Generalized Radial Basis Function

Because of the drawbacks of the original formulation of the GRBF, we propose the reformulation of the GRBF τ parameter as a function compounded by the radii and the α angle formed by the x-axis with the tangent to the GRBF at the point where $x = r$ (Fig. 4).

The reformulation of the GRBF, from the α angle (Fig. 4), is obtained as follows: firstly, we derive the GRBF with respect to the input variable ($c = 0$):

$$\frac{\partial B_j(x, \mathbf{w}_j)}{\partial x} = -e^{-(\frac{x}{r})^\tau} \cdot \tau \cdot x^{\tau-1} \cdot r^{-\tau}. \tag{3}$$

Secondly, the derivative at the point $x = r$ is calculated

$$\tan(\beta) = \frac{\partial B_j(x, \mathbf{w}_j)}{\partial x}(x = r) = -\frac{\tau}{e \cdot r}. \tag{4}$$

Finally, taking into account that $\tan(\alpha) = -\tan(\beta)$ (Fig. 4) and that the derivative of the GRBF with respect to the input variable in the point $x = r$ is equal to $\tan(\beta)$, the α angle is determined as:

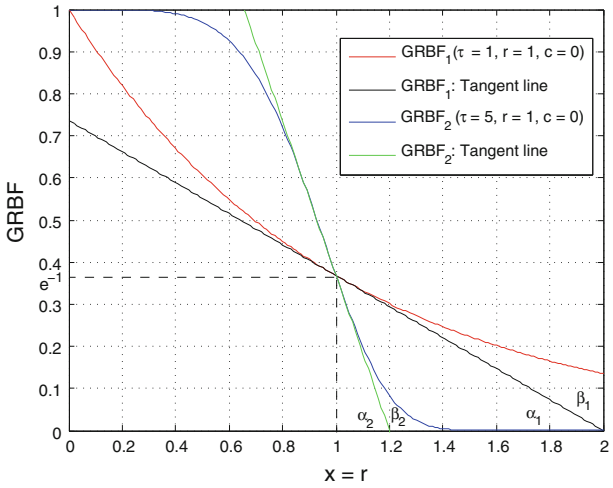

Fig. 4 GRBF according to the new α parameter for a one dimensional input space

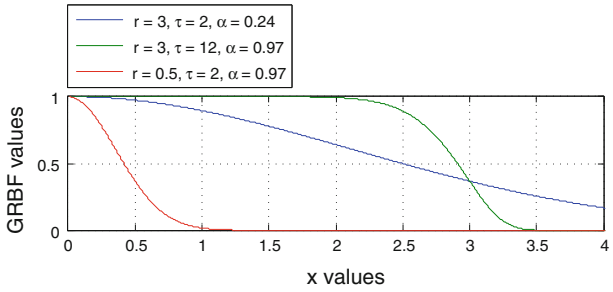

Fig. 5 GRBF representations in terms of α and their equivalent τ value

$$\alpha = \arctan\left(\frac{\tau}{e \cdot r}\right). \tag{5}$$

Therefore, the reformulated GRBF is expressed as:

$$B_j(\mathbf{x}, \mathbf{w}_j) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{c}_j\|}{r_j}\right)^{e \cdot r_j \cdot \tan \alpha_j}. \tag{6}$$

Thus, interesting behavior achieved with this reformulated model is the ability to vary the magnitude of the radii keeping the basis function curvature. This effect can be observed in Fig. 5 where two GRBF (red and green lines) are represented in terms of the α parameter with the same value of $\alpha = 0.97$ and different radii values. However, using the original GRBF formulation, a modification in the r value maintaining the τ value constant causes variations in the curvature (red and blue lines, from $\alpha = 0.97$ to 0.24).

4 Neuro-Logistic Models

In the classification problem, some measurements $x_i, i = 1, 2, \ldots, k$ are taken on a single pattern, and the patterns are classified into one of J populations. The measurements x_i are

random observations from these J classes. Let $D = \{(\mathbf{x}_n, \mathbf{y}_n); n = 1, 2, \ldots, N\}$ be a training dataset, where $\mathbf{x}_n = (x_{1n}, \ldots, x_{kn})$ is the vector of measurements taking values in $\Omega \subset \mathbb{R}^k$, and $\mathbf{y}_n$ is the class level of the n-th individual. The common technique of representing the class levels using a "1-of-J" encoding vector is adopted, $\mathbf{y} = \left(y^{(1)}, y^{(2)}, \ldots, y^{(J)}\right)$, such as $y^{(l)} = 1$ if $\mathbf{x}$ corresponds to an example belonging to class l and $y^{(l)} = 0$ otherwise.

Logistic model supposes that the conditional probability that $\mathbf{x}$ belongs to class l verifies: $p\left(y^{(l)} = 1 \middle| \mathbf{x}, \boldsymbol{\theta}_l\right) > 0, l = 1, 2, \ldots, J, \mathbf{x} \in \Omega$, and sets the function:

$$f_l(\mathbf{x}, \boldsymbol{\theta}_l) = \log \frac{p\left(y^{(l)} = 1 \middle| \mathbf{x}, \boldsymbol{\theta}_l\right)}{p\left(y^{(J)} = 1 \middle| \mathbf{x}, \boldsymbol{\theta}_l\right)}, \tag{7}$$

where $\boldsymbol{\theta}_l$ is the weight vector corresponding to class l, and $f_J(\mathbf{x}, \boldsymbol{\theta}_J) = 0$. Under a multinomial logistic regression, the probability that $\mathbf{x}$ belongs to class l is then given by:

$$p\left(y^{(l)} = 1 \middle| \mathbf{x}, \boldsymbol{\theta}\right) = \frac{\exp f_l\left(\mathbf{x}, \boldsymbol{\theta}_l\right)}{1 + \sum_{j=1}^{J-1} \exp f_j\left(\mathbf{x}, \boldsymbol{\theta}_j\right)}, \; l = 1, 2, \ldots, J, \tag{8}$$

where $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_{J-1})$. The hybrid Neuro-Logistic models are based on the combination of the standard linear model and nonlinear terms constructed with GRBFs, which captures possible locations in the input space. The general expression of the model is given by:

$$f_l(\mathbf{x}, \boldsymbol{\theta}_l) = \alpha_0^l + \sum_{i=1}^{k} \alpha_i^l x_i + \sum_{j=1}^{m} \beta_j^l B_j(\mathbf{x}, \mathbf{w}_j) \tag{9}$$

where $l = 1, 2, \ldots, J-1$, $\boldsymbol{\theta}_l = (\boldsymbol{\alpha}^l, \boldsymbol{\beta}^l, \mathbf{W})$ is the vector of parameters for each discriminant function, $\boldsymbol{\alpha}^l = (\alpha_0^l, \alpha_1^l, \ldots, \alpha_k^l)$ and $\boldsymbol{\beta}^l = \left(\beta_1^l, \ldots, \beta_m^l\right)$ are the coefficients of the multilogistic regression model, $\mathbf{W} = (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_m)$ are the parameters of the nonlinear transformations and $B_j(\mathbf{x}, \mathbf{w}_j$ is the GRBF (described in Sect. 2)).

5 Estimation of Neuro-Logistic Parameters

In the supervised learning context, the components of the weight vectors $\boldsymbol{\theta}=(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_{J-1})$ are estimated from the training dataset D. To perform the maximum likelihood estimation of $\boldsymbol{\theta}$, one can minimize the negative log-likelihood function:

$$\begin{aligned} L(\boldsymbol{\theta}) &= -\frac{1}{N} \sum_{n=1}^{N} \sum_{l=1}^{J} \left(y_i^l \log p\left(y_n^{(l)} = 1 \middle| \mathbf{x}_n, \boldsymbol{\theta}\right)\right) \\ &= \frac{1}{N} \sum_{n=1}^{N} \left[-\sum_{l=1}^{J} y_n^{(l)} f_l(\mathbf{x}_n, \boldsymbol{\theta}_l) + \log \sum_{l=1}^{J} \exp f_l(\mathbf{x}_n, \boldsymbol{\theta}_l) \right], \end{aligned} \tag{10}$$

where $f_l(\mathbf{x}, \boldsymbol{\theta}_l)$ corresponds to the hybrid model defined in (9) and $p\left(y^{(l)} = 1 \middle| \mathbf{x}_n, \boldsymbol{\theta}\right)$ corresponds to the conditional probability defined in (8).

The methodology proposed tries to maximize the log-likelihood function where classical gradient methods are not recommended due to the convolved nature of the error function. It is based on the combination of an Evolutionary Programming algorithm (EP) (global explorer) and a local optimization procedure (local exploiter) carried out by the standard maximum likelihood optimization method.

In this paper, the MultiLogistic algorithm has been considered for obtaining the maximum likelihood solution for the neuro-logistic models, available in the WEKA[1] workbench [30]. The MultiLogistic algorithm builds a multinomial logistic regression with a ridge estimator to prevent overfitting by penalizing large coefficients. This model is trained with a Quasi-Newtonian Method [16].

The estimation of the model coefficients is divided into three steps.

Step 1 We apply an Evolutionary Programming (EP) algorithm to find the basis functions:

$$\mathbf{B}(\mathbf{x}, \mathbf{W}) = \{B_1(\mathbf{x}, \mathbf{w}_1), B_2(\mathbf{x}, \mathbf{w}_2), \ldots, B_m(\mathbf{x}, \mathbf{w}_m)\}, \tag{11}$$

corresponding to the nonlinear part of $f(\mathbf{x}, \boldsymbol{\theta}_l)$. We have to determine the number of basis functions m and the weight matrix $\mathbf{W} = (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_m)$.

The weight matrix $\mathbf{W}$, the parameters of the output layer ($\boldsymbol{\beta}$ vector) and the structure of the GRBF are estimated by means of an evolutionary neural network algorithm that optimizes the error function given by the negative log-likelihood for N observations associated with the neural network model (see Eq. 10). The specific details of this EP algorithm can be found in some previous works [10,11].

As discussed previously, the model introduces a new parameter, the α angle, which has to be estimated during the evolutionary process. In the initialization step of the EP, the α angle of each basis function is initialized as $\alpha = \arctan\left(\frac{2}{e \cdot r}\right)$ (Eq. 5). This expression allows the whole basis function to be initialized with $\tau = 2$ since the GRBF for $\tau = 2$ reproduces the SRBF.

On the other hand, the parametric mutator modifies the α parameter of each basis function by adding a random uniform ζ value in the interval $[-1, 1]$ radians. Finally, when the structural mutator adds a new GRBF hidden node, it is included in the model with a $\tau = 2$. Therefore, the α angle is set to $\alpha = \arctan\left(\frac{2}{e \cdot r}\right)$.

We only consider the estimated weight matrix $\hat{\mathbf{W}} = (\hat{\mathbf{w}}_1, \hat{\mathbf{w}}_2, \ldots, \hat{\mathbf{w}}_m)$, which builds the basis functions. The values for the $\boldsymbol{\beta}$ vector are updated during the EA but the final configuration will be determined in step 3 together with those of the $\boldsymbol{\alpha}$ coefficient vector.

Finally, it is important to bear in mind that the *"perfect multicollinearity"* problem occurs when one input variable takes a constant value in all observations. When this problem appears, multinomial logistic regression performance suffers drastically. As discussed above, this approach adds the GRBFs of the best neural network model in the EP algorithm as new covariates. Due to the problem of perfect multicollinearity in multinomial logistic regression methods, the validity of each GRBF is checked four times in the EP. When the difference between the maximum and minimum activation values of the GRBF hidden node is lower than 0.05, this basis function is removed. To compensate the removal of the GRBF, the bias values of each output neuron that this basis function is connected to are incremented using the following expression:

$$\alpha_0^{j'} = \alpha_0^j + \frac{\max\left(B_h(\mathbf{x}, \mathbf{w}_h)\right) + \min\left(B_h(\mathbf{x}, \mathbf{w}_h)\right)}{2} \cdot \beta_h^j \tag{12}$$

where j is the index of the output neuron, h is the index of the GRBF hidden node to be deleted, α_0^j and $\alpha_0^{j'}$ are the bias of j-th output neuron before and after the removal of the GRBF, $\max\left(B_h(\mathbf{x}, \mathbf{w_h})\right)$ and $\min\left(B_h(\mathbf{x}, \mathbf{w_h})\right)$ are the maximum and minimum activation values of the h-th GRBF using the training set and β_h^j is the link weight from the selected

[1] http://www.cs.waikato.ac.nz/ml/weka/.

node to be deleted, to the j-th output function. The algorithm's computational cost is reduced because the elimination of hidden nodes implies that less operations needs to be computed during the EP.

Step 2 We consider the following transformation of the input space by including the non-linear basis functions obtained by the EP algorithm in step 1:

$$H : \mathbb{R}^k \rightarrow \mathbb{R}^{k+m}, (x_1, x_2, \ldots, x_k) \rightarrow (x_1, x_2, \ldots, x_k, z_1, \ldots, z_m), \tag{13}$$

where $z_1 = B_1(\mathbf{x}, \hat{\mathbf{w}}_1), \ldots, z_m = B_m(\mathbf{x}, \hat{\mathbf{w}}_m)$.

Step 3 In the third step, we minimize the negative log-likelihood function for N observations:

$$L(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{N} \sum_{n=1}^{N} \left[- \sum_{l=1}^{J} y_n^{(l)} (\boldsymbol{\alpha}^l \mathbf{x}_n + \boldsymbol{\beta}^l \mathbf{z}_n) + \log \sum_{l=1}^{J} \exp(\boldsymbol{\alpha}^l \mathbf{x}_n + \boldsymbol{\beta}^l \mathbf{z}_n) \right], \tag{14}$$

where $\mathbf{x}_n = (1, x_{1n}, \ldots, x_{kn})$ and $\mathbf{z}_n = (z_{1n}, \ldots, z_{mn})$. Now, the Hessian matrix of the negative log-likelihood in the new variables $x_1, x_2, \ldots, x_k, z_1, \ldots, z_m$ is semi-definite positive. The estimated coefficient vector $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}, \hat{\mathbf{W}})$ determines the model of (9) with $B_j(\mathbf{x}, \mathbf{w}_j)$ defined as (1).

In this final step, the logistic regression algorithm has been used for obtaining the parameter matrix $\boldsymbol{\theta}$. Moreover, two different versions of the hybrid neuro-logistic models have been considered: LR models with only the non-linear part, i.e. the model does not include the initial covariates of the problem ($\boldsymbol{\beta}^l \mathbf{z}_n$), and LR models with both the linear and the non-linear part ($\boldsymbol{\alpha}^l \mathbf{x}_n + \boldsymbol{\beta}^l \mathbf{z}_n$). The combined application of the logistic regression algorithm with the evolutionary algorithms with and without out initial covariates results into three different methods: Evolutionary Generalized Radial Basis Function (EGRBF), MultiLogistic regression with EGRBFs (MLEGRBF), and MultiLogistic Including covariates and EGRBFs (MLIEGRBF).

6 Experiments

6.1 Datasets Description

To validate the effectiveness of our method, a series of experiments were performed on six publicly available gene microarray datasets. They are often used to validate the performance of the classifier and gene selector. Due to high dimensionality and small sample size, gene selection is an essential prerequisite for further data analysis. There are brief descriptions given below.

Breast consists of 97 patterns collected from breast cancer patients. 46 of them are from patients labeled as *relapse*, the rest of the 51 patterns are from patients who remain healthy from the disease and are regarded as *non-relapse*. Each sample is described by 24,481 genes.

CNS (central nervous system) is derived from patient patterns in embrional tumors of the central nervous system. The total number of genes to be tested is 7,129 and the number of patterns is 60. There are two types of patterns in the dataset, where 21 are survivors (those who survive the treatment) and 39 are failures (who succumbed to the disease).

Colon uses Affymetrix oligonucleotide arrays to monitor expression levels of over 6,500 human genes from 40 tumor and 22 normal colon tissue patterns. The 2,000 genes with the highest minimal intensity across the 62 tissues were used in this analysis.

Table 1 Characteristics of the six datasets used for the experiments

Dataset	Source	Genes	FS	Size	R	B	N	#In	#Out	NPC	$[M_{\min}, M_{\max}]$	*Gen*
Breast	[28]	24481	BARS	97	183	–	–	183	2	(46,51)	[1, 3]	100
			FCBF		493	–	–	493				
CNS	[25]	7129	BARS	60	187	–	–	187	2	(21,39)	[1, 3]	10
			FCBF		170	–	–	170				
Colon	[1]	2000	BARS	62	58	–	–	58	2	(40,22)	[1, 3]	10
			FCBF		59	–	–	59				
Leukemia	[17]	7129	BARS	72	225	–	–	225	2	(42,25)	[1, 3]	50
			FCBF		203	–	–	203				
Lung	[4]	12600	BARS	203	237	–	–	237	5	(139,17,6,21,20)	[5, 8]	100
			FCBF		250	–	–	250				
Gcm	[26]	16063	BARS	253	311	–	–	311	14	(11,10,11,11,22,	[25, 28]	400
			FCBF		264	–	–	264		11,11,11,20)		

Leukemia refers to the primary disorders of bone marrow. This dataset contains 72 patterns with malignant neoplasms of hematopoietic stem cells, of which 47 are *acute lymphobastic leukemia* (ALL) and 25 *acute myeloid leukemia* (AML). The total number of genes to be tested is 7,129.

Lung has 12,600 genes in 203 patterns. The 203 patterns consist of 139 lung adenocarcinomas (AD), 21 squamous (SQ) cell carcinoma cases, 20 pulmonary carcinoid (COID) tumors and 6 small cell lung cancer cases (SCLC), as well as 17 normal lung (NL) patterns.

GCM contains 190 patterns. These patterns are divided into 14 varieties of tumor. The expression levels of 16,063 genes are reported.

As we discussed above, the motivation for applying feature selection (FS) techniques has shifted from being optional to becoming a real prerequisite for model building. In this work, to avoid dimensionality problems, two feature selectors are applied: Fast Correlation-Based Filter [31] (FCBF) and Best Agglomerative Ranked Subset [27] (BARS), to obtain relevant features and to remove redundancy. These features are considered input variables in the neuro-logistic models proposed in this paper.

Table 1 shows the characteristics of the six datasets used for the experiments: the original number of genes (Genes), feature selection type (FS), number of instances (Size), number of Real (R), Binary (B) and Nominal (N) input variables, total number of inputs (#In.), number of classes (#Out.), number of patterns per-class (NPC), minimum and maximum number of hidden nodes used for each dataset ($[M_{\min}, M_{\max}]$) and number of generations (#Gen.).

In these six microarray datasets, all gene expression values are numeric. For convenience sake, we did a simple linear rescaling of the input variables over the interval $[-2, 2]$, with X_i^* being the transformed variables, after feature selection. In this work, the GRBFs combine the input variables via the Euclidean distance function. The contribution of an input variable will depend heavily on its variability with respect to other input variables. If one input has a range of 0–1, while another input has a range of 0–1,000, then the contribution of the first input variable to the distance will be swamped by the second input variable. So it is essential to rescale all the input variables at the same interval in order for each variable to be equally important. Finally, the rescaling interval was $[-2, 2]$ in order to consider symmetric probability distributions with zero mean.

6.2 Experimental Framework

Different state-of-the-art statistical and artificial intelligence algorithms have been considered for the purpose of comparison. Specifically, the results of the following algorithms have been compared to the methodologies presented in this paper:

1. Baseline classifiers non-related with the proposal:
   (a) The C4.5 classification tree inducer [22].
   (b) The AdaBoost.M1 algorithm [22], using C4.5 as the base learner and the maximum number of iterations set to 100 iterations (Ada(C4.5)).
   (c) A Multilayer Perceptron (MLP) [22] with sigmoid units as hidden nodes, obtained by means of the backpropagation algorithm.
2. Baseline classifiers related with the proposal:
   (a) A Gaussian Radial Basis Function Network (RBFN) [30], deriving the centres and widths of hidden nodes using k-means, and combining the outputs obtained from the hidden layer using logistic regression.
   (b) The MultiLogistic (MLogistic) algorithm. It is a method for building a multinomial logistic regression model with a ridge estimator to guard against overfitting by penalizing large coefficients [6].
   (c) The SimpleLogistic (SLogistic) algorithm. It is based on applying LogitBoost algorithm with simple regression functions and determining the optimum number of iterations by a five fold cross-validation [22].
   (d) The Logistic Model Tree (LMT) [22] classifier.
   (e) The Support Vector Machine (SVM) classifier [29] with RBF kernels.

   These algorithms have been selected because they are closely related to our proposal. The four first methods are based on logistic regression approaches and the last one is related to our methodologies from a structural point of view. Many of these approaches have also been tested before in the classification problem on microarray gene expression. The detailed description and some previous results of these methods can be found in [18,22,30].

All the parameters used in the EA except the maximum and minimum number of RBFs in the hidden layer ($[M_{\min}, M_{\max}]$) and the number of generations ($\#Gen$) have the same values in all problems analyzed below (Table 1). The connections between hidden and output layer are initialized in the $[-5, 5]$ interval (i.e. $[-I, I] = [-5, 5]$). The size of the population is $N = 500$. For the structural mutation, the number of nodes that can be added or removed is within the $[1, 2]$ interval, and the number of connections to add or delete in the hidden and the output layer during structural mutations is within the $[1, 7]$ interval.

For the selection of the SVM hyperparameters (regularization parameter, C, and width of the Gaussian functions, γ), a grid search algorithm has been applied with a ten-fold cross-validation, using the following ranges: $C \in \{2^{-5}, 2^{-3}, \ldots, 2^{15}\}$ and $\gamma \in \{2^{-15}, 2^{-13}, \ldots, 2^{3}\}$.

The experimental design was conducted using a holdout cross validation procedure with $3n/4$ instances for the training dataset and $n/4$ instances for the generalization dataset. The evaluation of the different models has been performed using the Correctly Classified Rate (CCR) or accuracy measure. In order to evaluate the stability of the methods, the evolutionary algorithm was run 30 times. Finally, the EA and the neuro-logistic model proposed were implemented in JAVA. We also used "libsvm" [7] to obtain the results of the SVM method, and WEKA to obtain the results of the remaining methods.

6.3 Analysis of Results

This section analyzes the results obtained. Specifically, we check the performance (mean accuracy value from the 30 executions of each dataset) of the neuro-logistic approaches and the single accuracy of the other five related methodologies.

In Table 2, the mean and the standard deviation of the correct classification rate in the generalization set (C_G) is shown for each dataset and a total of 30 executions. From the analysis of the results, it can be concluded, from a purely descriptive point of view, that the MLIEGRBF methodology yields the best mean ($\overline{C}_{\mathrm{G}} = 91.08\%$) and ranking ($\overline{R}_{C_{\mathrm{G}}} = 2.08$) in C_{G}.

To determine the statistical significance of the rank differences observed for each method in the different datasets, we have carried out a non-parametric Friedman test [14] with the ranking of C_{G} of the best models as the test variable (since a previous evaluation of the C_{G} values results in rejecting the normality and the equality of variances' hypothesis). The test shows that the effect of the method used for classification is statistically significant at a significance level of 10%, as the confidence interval is $C_0 = (0, F_{0.10} = 1.65)$ and the F-distribution statistical values is $F^* = 4.25 \notin C_0$ for C_{G}. Consequently, we reject the null-hypothesis stating that all algorithms perform equally in mean ranking.

Based on this rejection, the Holm post-hoc test is used to compare all classifiers to each other. Holm test is a multiple comparison procedure that considers a control algorithm and compares it with the remaining methods [8]. The test statistics for comparing the i-th and j-th method using this procedure is:

$$z = \frac{R_i - R_j}{\sqrt{\frac{k(k+1)}{6N}}}, \tag{15}$$

where k is the number of algorithms and N the number of dataset. The z value is used to find the corresponding probability from the table of normal distribution, which is then compared with an appropriate level of confidence α. Holm's test adjusts the value for α in order to compensate for multiple comparison. Holm's test is a step-up procedure that sequentially tests the hypotheses ordered by their significance. We will denote the ordered p-values by $p_1, p_2, \ldots, p_k$ so that $p_1 \leq p_2 \leq \cdots \leq p_{k-1}$. Holm's test compares each p_i with $\alpha/(k-i)$, starting from the most significant p value. If p_1 is below $\alpha/(k-1)$, the corresponding hypothesis is rejected and we allow to compare p_2 with $\alpha/(k-2)$. If the second hypothesis is rejected, the test proceeds with the third, and so on. As soon as a certain null hypothesis cannot be rejected, all the remain hypotheses are retained as well.

The results of the Holm test for $\alpha = 0.10$ can be seen in Table 2, using the corresponding p and α'_{Holm} values. From the results of this test, it can be concluded that the MLIEGRBF methodology obtains a significantly higher ranking of C_{G} when compared to the remaining methods, which justifies the proposal.

7 Conclusions

In this paper, the combination of Evolutionary Generalized Radial Basis Function (EGRBF) and Logistic Regression (LR) methods is analyzed to tackle classification problems. The proposed methodologies are compared to five state-of-the-art algorithms using six microarray gene expression datasets. The nonparametric Holm test indicated that our approach outperformed other methods using accuracy as the test variable. From these results, we can conclude that the combined effect of LR and EGRBF is greater than the average of their

Table 2 Comparison of the proposed method to the remaining methods: Mean and Standard Deviation (SD) of the accuracy results ($C_G(\%)$) from 30 executions, mean accuracy ($\overline{C}_G(\%)$), mean ranking ($\overline{R}$) , p-Value and α' for the Holm post-hoc non-parametric tests in C_G with $\alpha = 0.05$ (MLIEGRBF is the control method)

Dataset	FS	Method($C_G(\%)$)										
		C4.5	Ada100(C4.5)	MLP	RBFN	MLogistic	SLogistic	LMT	SVM	EGRBF	MLEGRBF	MLIEGRBF
		Result	Result	Result	Result	Result	Result	Result	Result	Mean_{SD}	Mean_{SD}	Mean_{SD}
Breast	(1)	72.00	76.00	*80.00*	*80.00*	*80.00*	72.00	72.00	*80.00*	$73.60_{4.88}$	$77.87_{5.53}$	$\mathbf{80.93_{3.59}}$
	(2)	64.00	**88.00**	**88.00**	80.00	*84.00*	*84.00*	*84.00*	76.00	$74.80_{7.50}$	$74.93_{7.71}$	$\mathit{87.73_{1.01}}$
CNS	(1)	66.66	73.33	73.33	**80.00**	73.33	66.66	66.66	66.67	$70.22_{9.22}$	$70.00_{10.47}$	$\mathit{77.56_{3.27}}$
	(2)	60.00	86.66	80.00	86.66	**100.00**	80.00	80.00	66.67	$84.89_{8.01}$	$84.00_{7.95}$	$\mathit{99.78_{1.22}}$
Colon	(1)	100.00	81.25	81.25	93.75	93.75	**100.00**	**100.00**	62.50	$\mathit{99.17_{2.16}}$	$\mathit{99.17_{2.16}}$	$\mathbf{100.00_{0.00}}$
	(2)	**87.50**	75.00	75.00	**87.50**	75.00	81.25	75.00	62.50	$77.92_{8.16}$	$78.33_{7.83}$	$\mathit{87.08_{1.59}}$
Leukemia	(1)	83.33	83.33	*94.44*	*94.44*	**100.00**	88.89	88.89	66.67	$97.22_{3.79}$	$97.59_{3.77}$	$\mathbf{100.00_{0.00}}$
	(2)	83.33	*94.44*	*94.44*	*94.44*	*94.44*	83.33	83.33	66.67	$95.93_{4.36}$	$98.15_{3.04}$	$\mathbf{100.00_{0.00}}$
Lung	(1)	94.11	**98.03**	**98.03**	96.07	96.07	**98.03**	**98.03**	**98.03**	$93.53_{1.94}$	$97.59_{3.77}$	$\mathit{97.91_{0.50}}$
	(2)	90.19	92.15	96.07	94.11	94.11	*98.03*	*98.03*	94.11	$87.58_{5.00}$	$98.14_{3.04}$	$\mathbf{100.00_{0.00}}$
Gcm	(1)	57.69	73.07	67.30	75.00	73.07	63.49	71.15	75.00	$24.04_{6.50}$	$54.81_{4.84}$	$\mathbf{77.56_{0.93}}$
	(2)	67.30	80.76	82.69	82.00	80.76	71.15	67.30	80.76	$26.24_{6.68}$	$54.09_{7.39}$	$\mathbf{84.35_{2.71}}$
$\overline{C}_G(\%)$		77.17	83.50	84.21	86.99	*87.04*	82.23	82.03	74.63	75.42	82.05	**91.08**
$\overline{R}_{C_G}$		8.50	5.87	5.25	*4.66*	5.12	6.45	6.79	7.58	7.62	6.04	**2.08**
p-Value		0.000	0.005	0.019	0.056	0.024	0.001	5.1E-4	5.0E-5	4.0E-5	0.003	–
α'_{Holm}		0.010	0.025	0.033	0.100	0.050	0.016	0.014	0.0125	0.011	0.020	–

In columns 3 to 10 best result is in bold face and the second best result in italics. Mean_{SD} values are given in the last three columns
(1): BARS; (2): FCBF

individual effects, demonstrating the good synergy between these two techniques. Finally, this approach is presented by the scientific biology community as a competitive alternative to classify provided datasets.

Acknowledgements This work has been partially subsidized by the TIN 2008-06681-C06-03 project granted by the Spanish Inter-Ministerial Commission of Science and Technology (MICYT), FEDER funds and the P08-TIC-3745 project of the "Junta de Andalucía" (Spain). The research of Francisco Fernández-Navarro has been funded by the "Junta de Andalucía" Predoctoral Program, grant reference 390015-P08-TIC-3745. This work has been partially subsidized with the "Doctoral Training on Softcomputing" project subsidized by the Junta de Andalucía, the Ibero-American University Postgraduate Association (AUIP) and the Ministry of Higher Education of the Republic of Cuba.

References

1. Alon U, Barka N, Notterman DA, Gish K, Ybarra S, Mack D, Levine AJ (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(12):6745–6750
2. Bandurski K, Kwedlo W (2010) A lamarckian hybrid of differential evolution and conjugate gradients for neural network training. Neural Process Lett 32(1):31–44
3. Beyer K, Goldstein J, Ramakrishnan R, Shaft U (1999) When is Nearest neighbor meaningful? In: International conference on database theory, pp 217–235
4. Bhattacharjee A, Richards W, Staunton J, Li C, Monti S, Vasa P, Ladd C, Beheshti J, Bueno R, Gillette M, Loda M, Weber G, Mark E, Lander E, Wong W, Johnson B, Golub T, Sugarbaker D, Meyerson M (2001) Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses. Proc Natl Acad Sci USA 98(24):13,790–13,795
5. Castaño A, Fernández-Navarro F, Hervás-Martínez C, Gutierrez PA, García MM (2010) Classification by evolutionary generalized radial basis functions. Int J Hybrid Intell Syst 7(1):1–10
6. Cessie Sle , Houwelingen Jvan (1992) Ridge estimators in logistic regression. Appl Stat 41(1):191–201
7. Chang C, Lin C (2011) Libsvm: a library for support vector machines
8. Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30
9. Fernández-Navarro F, Hervás-Martínez C, Cruz M, Gutierrez PA, Valero A (2011) Evolutionary q-gaussian radial basis function neural network to determine the microbial growth/no growth interface of *Staphylococcus aureus*. Appl Soft Comput 11(3):3012–3020
10. Fernández-Navarro F, Hervás-Martínez C, Gutíerrez PA (2011) A dynamic over-sampling procedure based on sensitivity for multi-class problems. Pattern Recognition. http://dx.doi.org/10.1016/j.patcog.2011.02.019
11. Fernández-Navarro F, Hervás-Martínez C, Gutierrez PA, Carboreno M (2011) Evolutionary q-gaussian radial basis functions neural networks for multi-classification. Neural Networks In Press. http://dx.doi.org/10.1016/j.neunet.2011.03.014
12. Fernández-Navarro F, Hervás-Martínez C, Sánchez-Monedero J, Gutierrez PA (2011) MELM-GRBF: a modified version of the extreme learning machine for generalized radial basis function neural networks. Neurocomputing (in press)
13. Francois D (2008) High dimentional data analisis, from optimal metric to feature selection. In: Seeking on right metric. VDM Verlag, Saarbrucken, pp 54–55
14. Friedman M (1940) A comparison of alternative tests of significance for the problem of m rankings. Ann Math Stat 11(1):86–92
15. Fu L, Zhang M, Li H (2010) Sparse rbf networks with multi-kernels. Neural Process Lett 32(3):235–247
16. Gill PE, Murray W, Wright MH (1982) Practical optimization. Academic Press, New York
17. Golub TR, Slonim DK, Tamayo P, Huard C, Gaasenbeek M, Mesirov JP, Coller H, Loh ML, Downing JR, Caligiuri MA, Bloomfield CD, Lander ES (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286(5439):531–537
18. Hastie T, Tibshirani R, Friedman JH (2001) The elements of statistical learning. Springer, New York
19. Hervás-Martínez C, Martínez-Estudillo F (2007) Logistic regression using covariates obtained by product-unit neural network models. Pattern Recognit 40(1):52–64
20. Hervás-Martínez C, Martínez-Estudillo FJ, Carbonero-Ruz M (2008) Multilogistic regression by means of evolutionary product-unit neural networks. Neural Netw 21(7):951–961

21. Howell AJ, Buxton H (2002) RBF network methods for face detection and attentional frames. Neural Process Lett 15(3):197–211
22. Landwehr N, Hall M, Frank E (2005) Logistic model trees. Mach Learn 59(1–2):161–205
23. Li J, Liu X (2011) Melt index prediction by RBF neural network optimized with an MPSO-SA hybrid algorithm. Neurocomputing 74(5):735–740
24. Li M, Huang G, Saratchandran P, Sundararajan N (2005) Performance evaluation of gap-rbf network in channel equalization. Neural Process Lett 22(2):223–233
25. Pomeroy SL, Tamayo P, Gaasenbeek M, Sturla LM, Angelo M, McLaughlin ME, Kim JYH, Goumnerova LC, Black PM, Lau C, Allen JC, Zagzag D, Olson JM, Curran T, Wetmore C, Biegel JA, Poggio T, Mukherjee S, Rifkin R, Califano A, Stolovitzky G, Louis DN, Mesirov JP, Lander ES, Golub TR (2002) Prediction of central nervous system embryonal tumour outcome based on gene expression. Nature 415(6870):436–442
26. Ramaswamy S, Tamayo P, Rifkin R, Mukherjee S, Yeang C, Angelo M, Ladd C, Reich M, Latulippe E, Mesirov JP, Poggio T, Gerald W, Loda M, Lander ES, Golub TR (2001) Multiclass cancer diagnosis using tumor gene expression signatures. Proc Natl Acad Sci USA 98(26):15,149–15,154
27. Ruiz R, Aguilar-Ruiz J, Riquelme J (2008) Best agglomerative ranked subset for feature selection. JMLR Workshop Conf Proc 4:146–160
28. Van't Veer LJ, Dai H, Vande Vijver MJ, He YD, Hart AAM, Mao M, Peterse HL, VanDer Kooy K, Marton MJ, Witteveen AT, Schreiber GJ, Kerkhoven RM, Roberts C, Linsley PS, Bernards R, Friend SH (2002) Gene expression profiling predicts clinical outcome of breast cancer. Nature 415(6871): 530–536
29. Vapnik VN (1999) The nature of statistical learning theory. Springer, New York
30. Witten IH, Frank E (2005) Data mining: practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann series in data management systems. Elsevier, Amsterdam
31. Yu L, Liu H (2003) Feature selection for high-dimensional data: a fast correlation-based filter solution. In: Fawcett T, Mishra NICML. AAAI Press, San Francisco, pp 856–863
32. Zhang M (2009) Ml-rbf: Rbf neural networks for multi-label learning. Neural Process Lett 29(2):61–74
33. Zhang ML, Zhou ZH (2006) Adapting RBF neural networks to multi-instance learning. Neural Process Lett 23(1):1–26