Talanta 83 (2011) 1014–1022

Contents lists available at ScienceDirect

Talanta

journal homepage: www.elsevier.com/locate/talanta

Prediction of capillary gas chromatographic retention times of fatty acid methyl esters in human blood using MLR, PLS and back-propagation artificial neural networks

Vinod Kumar Gupta[a,b,*], Hadi Khani[c], Behzad Ahmadi-Roudi[d], Shima Mirakhorli[c], Ehsan Fereyduni[c], Shilpi Agarwal[e]

[a] Department of Chemistry, Indian Institute of Technology Roorkee, Roorkee, UA 247667, India
[b] Department of Chemistry, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia
[c] Faculty of Chemistry, Tarbiat Moallem University, Tehran, Iran
[d] Chemistry Department, Faculty of Science, Vali-e-Asr University, Rafsanjan, Iran
[e] School of Studies in Chemistry, Jiwaji University, Gwalior, MP, India

ARTICLE INFO

Article history:
Received 16 September 2010
Received in revised form 2 November 2010
Accepted 4 November 2010
Available online 11 November 2010

Keywords:
Quantitative structure–retention relationship (QSRR)
Fatty acid methyl esters
Gas chromatography
Artificial neural network (ANN)

ABSTRACT

Quantitative structure–retention relationship (QSRR) models correlating the retention times of fatty acid methyl esters in high resolution capillary gas chromatography and their structures were developed based on non-linear and linear modeling methods. Genetic algorithm (GA) was used for the selection of the variables that resulted in the best-fitted models. Gravitational index (G2), number of *cis* double bond (N*c*DB) and number of *trans* double bond (N*t*DB) were selected among a large number of descriptors. The selected descriptors were considered as inputs for artificial neural networks (ANNs) with three different weights update functions including Levenberg–Marquardt backpropagation network (LM-ANN), BFGS (Broyden, Fletcher, Goldfarb, and Shanno) quasi-Newton backpropagation (BFG-ANN) and conjugate gradient backpropagation with Polak–Ribiére updates (CGP-ANN). Computational result indicates that the LM-ANN method has better predictive power than the other methods. The model was also tested successfully for external validation criteria. Standard error for the training set using LM-ANN was SE = 0.932 with correlation coefficient $R = 0.996$. For the prediction and validation sets, standard error was SE = 0.645 and SE = 0.445 and correlation coefficient was $R = 0.999$ and $R = 0.999$, respectively. The accuracy of 3–2–1 LM-ANN model was illustrated using leave multiple out-cross validations (LMO-CVs) and Y-randomization.

1. Introduction

During the past two decades, determination of fatty-acid content in bio-fluids, such as blood and plasma, has emerged into an intense focus of research in several areas, including, e.g. environmental chemistry, food science, and medicine [1,2].

The determination of different classes of fatty acids in human blood is very important because it could caused the certain human cancers, including carcinoma of the breast [3], prostate [4], colon [5], ovary [6] and endometrium [7] at the high levels of fatty acid.

However, due to the diversity of fatty acids in terms of their chain length, branching, degree of unsaturation, geometry and position of the double bonds, as well as the presence of other substituents and annular structures their analysis is still a challenge today [8].

Capillary gas chromatography (CGC) is the traditionally used technique for the analysis of fatty acids, which are commonly separated as their methyl ester derivatives (fatty acid methyl esters, FAMEs) [9]. However, it is hard work to determine experimentally the retention times for all possible FAME compounds, due to not only the extremely large number of isomers, but also the lack of synthesized FAME standards. Moreover, mass spectra of fatty acid (methyl ester) isomers are very similar and often show a very strong fragmentation, resulting in nonspecific spectra, even for fatty acids with different carbon chain lengths [1]. Therefore, a method for accurately predicting retention times/indices would be of help for the identification of individual fatty acid methyl esters.

The methodology of relating chemical structure with chromatographic retention parameters is known as quantitative structure–retention relationship (QSRR) which has two main goals including the prediction of retention coefficients and the explanation of the chromatographic mechanisms [10].

* Corresponding author at: Department of Chemistry, Indian Institute of Technology Roorkee, Roorkee, UA 247667, India. Tel.: +91 1332285801; fax: +91 1332273560.
E-mail addresses: vinodfcy@iitr.ernet.in, vinodfcy@gmail.com (V.K. Gupta).

0039-9140/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.talanta.2010.11.017

The advantage of this approach over the other methods lies in the fact that the descriptors used to build models are mostly obtained from the structures of the analytes, and it only depends on few experimental properties [11].

One of the most important problems is how to represent molecular structure for QSRR. Generally, the descriptors encoding the molecular structure are classified as physicochemical, quantum-chemical, topological, geometrical, constitutional, etc. descriptors. The second crucial factor is to select the most informative descriptors from among a large number of correlated descriptors [12].

Various modeling techniques have been widely used in QSRR, such as multiple linear regression (MLR) [13], partial least square (PLS) [13], artificial neural network (ANN) [14,15] and support vector machine (SVM) [16,17].

MLR yields models that are simpler and easier to interpret than PLS, because these methods perform regression on latent variables that do not have physical meaning. Due to the co linearity problem in MLR analysis, one may remove the collinear descriptors before MLR model development. MLR equations can describe the structure activity relationships well but some information will be discarded in MLR analysis. On the other hand, factor analysis-based methods such as PLS regression can handle the collinear descriptors and therefore better predictive models will be obtained by PLS method [18].

ANNs have grown in popularity due to their ease of use and success in solving problems where complex nonlinear relationship exist and often produce superior QSRR models compared to models derived by the more traditional approach MLR and PLS [18–20].

There are a lot of optimization methods that can be used, of which genetic algorithm (GA) is one of the best. The genetic algorithm which is well known as the most interesting and more widely used variable selection method [21–23] is employed in this research.

Farkas et al. developed linear methods to build models for description and prediction of Kováts retention indices for a wide variety of fatty acid methyl esters replacing the measured properties by easily calculated two dimensional descriptors [24].

In this study, we have considered the relationship between the structure of fatty acid methyl ester derivatives and their retention times using different linear and non-linear chemometrics methods. In due course, the genetic algorithm (GA) was used for the selection of the variables that resulted in the best-fitted models. Artificial neural networks with three different weight update functions including Levenberg–Marquardt backpropagation network (LM-ANN), BFGS quasi-Newton backpropagation (BFG-ANN) and conjugate gradient backpropagation with Polak–Ribiére updates (CGP-ANN), have been used as non-linear methods for modeling and predicting of the retention times of FAMEs in human blood. In addition, genetic algorithm-multiple linear regression (GA-MLR) and partial least square (PLS), as linear methods; have also been used to treat the same data set. Obtained results indicate that Gravitational index (G2) and number of *cis* and *trans* double bonds have important role in capillary gas chromatographic retention time of studied compounds.

2. Theory

2.1. Artificial neural network

Artificial neural networks are parallel computational devices consisting of groups of highly interconnected processing elements called neurons. Neural networks are characterized by topology, computational characteristics of their elements, and training rules. Traditional neural networks have neurons arranged in a series of layers. The first layer is termed the input layer, and each of its neurons receives information from the exterior, corresponding to one of the independent variables used as inputs. The last layer is the output layer, and its neurons handle the output from the network. The layers of neurons between the input and output layers are called hidden layers. Each layer may make its independent computations and may pass the results yet to another layer.

Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons, and the same happens with artificial neural networks. The strength of the synapse from neuron i to neuron j is determined by means of a weight, w_{ij}. In addition, each neuron j from the hidden layer, and eventually the output neuron, are associated with a real value bj, named the neuron's bias and with a non-linear function, named the transfer or activation function.

The backpropagation network receives a set of inputs, which is multiplied by each node and then a non-linear transfer function is applied [14,25]. The goal of training the network is to change the weights between the layers in a direction to minimize the output errors. The changes in the values of the weights can be obtained using Eq. (1):

$$w_{ij,n} = F_n + \alpha w_{ij,n-1}; \quad (1)$$

where $w_{ij,n}$ is the change in the weight factor for each network node, α is the momentum factor, and F is a weight update function, which indicates how weights are changed during the learning process. There is no single best weight update function, which can be applied for all non-linear optimizations. One needs to choose a weight update function based on the characteristics of the problem and the data set of interest. Various types of algorithms have been found to be effective for most practical purposes. However, in this work we have used three different weights update functions of Levenberg–Marquardt (LM) algorithm, BFGS quasi-Newton backpropagation (BFG) and conjugate gradient backpropagation with Polak–Ribiére updates (CGP) which are discussed below.

2.1.1. Backpropagation neural networks

Backpropagation neural networks (BNNs) have non-linear differentiable transfer functions and multilayer feed-forward neural networks trained by backpropagation of errors (traditionally) using two algorithms including gradient descent or gradient descents with momentum [26]. These two backpropagation training algorithms are often too slow for practical problems. This section discusses several high-performance algorithms that can converge from ten to one hundred times faster than these two algorithms.

2.1.2. BFGS quasi-Newton backpropagation (BFG)

Trainbfg is a network training function that updates weight and bias values according to the BFGS quasi-Newton method. Trainbfg can train any network as long as its weight, net input, and transfer functions have derivative functions. Backpropagation is used to calculate derivatives of performance perf with respect to the weight and bias variables X. Each variable is adjusted according to Eq. (2):

$$X = X + a^* dX; \quad (2)$$

where dX is the search direction. The parameter a is selected to minimize the performance along the search direction. The line search function searchFcn is used to locate the minimum point. The first search direction is the negative of the gradient of performance. In succeeding iterations the search direction is computed according to Eq. (3):

$$dX = \frac{-H}{gX}; \quad (3)$$

where gX is the gradient and H is a approximate Hessian matrix [27].

2.1.3. Conjugate gradient backpropagation with Polak–Ribiére updates (CGP)

Traincgp is a network training function that updates weight and bias values according to conjugate gradient backpropagation with Polak–Ribiére updates. Traincgp can train any network as long as its weight, net input, and transfer functions have derivative functions. Backpropagation is used to calculate derivatives of performance perf with respect to the weight and bias variables *X*. Each variable is adjusted according to Eq. (2):

In succeeding iterations the search direction is computed from the new gradient and the previous search direction according to the following formula: Eq. (4):

$$dX = -gX + dX\_old^{*}Z; \tag{4}$$

where *gX* is the gradient. The parameter *Z* can be computed in several different ways. For the Polak–Ribiére variation of conjugate gradient, it is computed as Eq. (5):

$$Z = \frac{(gX - gX\_\text{old})^{*}gX}{\text{norm\_sqr}}; \tag{5}$$

where norm_sqr is the norm square of the previous gradient, and *gX*_old is the gradient on the previous iteration [27].

2.1.4. Levenberg–Marquardt (trainLM)

This algorithm appears to be the fastest method for training moderate-sized feedforward neural networks (up to several hundred weights). The Levenberg–Marquardt algorithm was designed to approach second-order training speed without having to compute the Hessian matrix. When the performance function has the form of a sum of squares (as is typical in training feedforward networks), then the Hessian matrix can be approximated as Eq. (6):

$$H = J^{T}J \tag{6}$$

In addition, the gradient can be computed as Eq. (7):

$$g = J^{T}e \tag{7}$$

where *J* is the Jacobian matrix that contains first derivatives of the network errors with respect to the weights and biases, and *e* is a vector of network errors. The Jacobian matrix can be computed through a standard backpropagation technique that is much less complex than computing the Hessian matrix.

The Levenberg–Marquardt algorithm uses this approximation to the Hessian matrix in the following Newton-like update: Eq. (8):

$$X_{k+1} = X_k - [J^{T}J + \mu I]^{-1}J^{T}e \tag{8}$$

when the scalar μ is zero, this is just Newton's method, using the approximate Hessian matrix. When μ is large, this becomes gradient descent with a small step size. Newton's method is faster and more accurate near an error minimum, so the aim is to shift toward Newton's method as quickly as possible. Thus, μ is decreased after each successful step (reduction in performance function) and is increased only when a tentative step would increase the performance function. In this way, the performance function is always reduced at the each iteration of the algorithm [18,27].

2.2. Genetic algorithm (GA)

Genetic algorithm is a (global) minimum search algorithm to solve optimization problems defined by a fitness criteria applying evolution hypothesis of Darwin and different genetic functions, i.e. crossover and mutation. GA has three basic operations (selection, cross-over and mutation) and takes the intermediate position between the stepwise and random approaches [28]. The result of GA is a whole population of solutions (variable combinations) and researchers have the opportunity to choose one for validation and development in future experiments.

The purpose of a variable selection is to select the variables significantly contributing to prediction and discard the other variables by a fitness function. In GA for variable selection, an individual (or chromosome), i.e. solution, represents a set of variables, there are the following basic steps in algorithms: (1) a chromosome is represented by a binary bit string and an initial population of chromosomes is created in a random way; (2) a value for the fitness function of each chromosome is evaluated; (3) according to the values of the fitness function, the chromosomes of the next generation are reproduced by selection, crossover and mutation operations [29].

3. Experimental

3.1. Data sets

Retention times of studied fatty acid methyl ester were taken from Ref. [1]. The retention times and names of these compounds are taken in Tables 1 and 2. The detailed descriptions of abbreviations using the shorthand annotation for all fatty acid methyl esters were given in Ref. [1]. Prior to the calculation of the molecular descriptors, the 3D structures of the studied compounds were optimized using semi-empirical quantum-chemical methods of AM1 implemented in HyperChem computer program [30].

3.2. Molecular descriptors

The main step in every QSRR study is choosing and calculating the structural descriptors as numerical encoded parameters representing the chemical structures. In the present work the molecular descriptors were generated using Dragon, version web 3.0 [31] and HYPERCHEM softwares; Moreover due to existence of *cis* and *trans* double bond in some studied fatty acid methyl ester, we use of number of *cis* double bound (N*c*DB) and number of *trans* double bound (N*t*DB) as two descriptors. Descriptors with constant or almost constant values for all molecules were eliminated. In addition, pairs of variables with a correlation coefficient greater than 0.90 were classified as intercorrelated and only one of them were considered in developing the models. A total of 172 descriptors were considered for further investigations after discarding the descriptors with constant and intercorrelated ones.

3.3. Variable selection

A major step in constructing the QSRR models is finding one or more molecular descriptors that represent variation in the structural property of the molecules numerically. The genetic algorithm (GA) was used for the selection of the variables that resulted in the best-fitted models. It has already been shown that genetic algorithms (GA) can be successfully used as a feature selection technique [21–23].

In this paper, size of the population is 50, the probability of initial variable selection is 3: V (V is the number of independent variables), the probability of crossover is 0.6, the probability of mutation is 0.01 and the number of evolution generations is 1000. For each set of data, 3000 runs were performed. Appropriate models with low standard errors and high correlation coefficients were obtained. Consequently, among different models, the best model was chosen, whose specifications are presented in Table 3. It is obvious that as the number of descriptors increase the R^2 will increase. Fig. 1a shows the effect of increasing the number of descriptors on R_p^2 values. It can be seen from this figure that increasing the number of parameters only up to three has a large influence on improving correlation. Therefore, we have chosen three descriptors as optimum number of parameters. The descriptors appearing in this model are G2, N*c*DB and N*t*DB, whose definitions are given in Table 3. As it can

Table 1
The observed and calculated retention time (Rt) of fatty acid methyl esters – training set for the ANNs, PLS and MLR models.

No.	Name[a]	Rt exp	Rt ANN-LM	Rt ANN-BFG	Rt ANN-CGP	Rt MLR	Rt PLS
1	C8:0	12.57	13.213	13.667	13.344	4.293	12.734
2	C15:0 anteiso	19.25	19.487	19.232	19.128	20.832	18.304
3	C15:0	19.68	19.522	19.265	19.166	20.886	18.937
4	C16:0 iso	20.44	21.170	20.825	20.921	23.204	21.085
5	C17:0	23.22	23.151	22.778	22.951	25.626	23.561
6	C18:0 iso	24.26	25.304	24.996	25.067	27.944	26.043
7	C18:0	25.51	25.357	25.052	25.118	27.998	26.302
8	C9:0	13.41	13.683	14.079	13.612	6.661	13.064
9	C19:0	28.07	27.829	27.718	27.509	30.37	28.857
10	C21:0	34.26	33.611	34.209	33.415	35.111	34.739
11	C22:0	37.75	37.009	37.906	37.148	37.483	37.942
12	C23:0 iso	39.46	40.757	41.602	41.235	39.801	40.925
13	C23:0	41.48	40.842	41.681	41.325	39.851	40.98
14	C24:0	45.49	45.126	45.288	45.578	42.223	44.331
15	C25:0	49.50	49.454	48.436	49.280	44.595	47.515
16	C26:0	53.67	52.733	50.889	51.894	46.963	51.285
17	C14:1 n5c	19.39	18.812	18.907	20.389	20.525	20.170
18	C15:1 n5c	20.61	20.421	20.399	22.055	22.848	18.865
19	C16:1 n7c	22.57	22.388	22.263	23.856	25.266	25.250
20	C11:0	15.13	14.965	15.192	14.548	11.401	13.291
21	C18:1 n10t	26.47	26.814	27.751	26.784	28.316	27.270
22	C18:1 n9t	26.47	26.829	27.776	26.815	28.337	27.250
23	C18:1 n8t	26.54	26.814	27.751	26.784	28.316	27.270
24	C18:1 n7t	26.72	26.814	27.751	26.784	28.316	26.936
25	C18:1 n6t	26.84	26.814	27.751	26.784	28.316	26.685
26	C18:1 n7c	27.20	27.186	27.004	27.727	30.006	29.752
27	C18:1 n6c	27.36	27.186	27.004	27.727	30.006	27.164
28	C18:1 n5c	27.56	27.186	27.004	27.727	30.006	29.212
29	C12:0	16.07	15.820	15.937	15.307	13.773	14.273
30	C18:1 n3c	27.78	27.214	27.033	27.750	30.031	31.117
31	C20:1 n9c	32.80	33.305	33.352	32.904	34.775	35.406
32	C24:1 n9c	47.90	48.532	47.696	48.297	44.231	47.094
33	C18:2 n6tt	28.76	28.810	30.899	30.994	28.663	28.326
34	C13:0	17.11	16.849	16.842	16.318	16.145	15.423
35	C18:2 n6ct	28.96	32.016	29.984	30.587	30.307	27.457
36	C18:2 n6cc	29.47	29.056	29.154	29.792	31.993	28.401
37	C20:2 n6c	35.79	35.960	36.035	35.400	36.771	35.116
38	γ-C18:3 n6c	31.19	30.999	31.541	31.682	33.997	30.847
39	C20:3 n6c	38.05	38.569	38.854	38.031	38.787	36.754
40	C20:4 n6c	39.79	41.101	41.735	40.889	40.832	39.069
41	C22:4 n6c	48.20	48.679	48.442	48.718	45.547	45.999
42	α-C18:3 n3c	32.39	31.029	31.570	31.704	34.017	32.859
43	C20:3 n3c	36.65	38.569	38.854	38.031	38.787	39.218
44	C22:6 n3c	54.30	51.660	51.800	52.282	49.534	54.571
45	c9,t11-CLA	33.01	32.062	30.034	30.643	30.345	34.574
46	t10,c12-CLA	33.24	32.072	30.045	30.655	30.353	32.645
47	t10,t12-CLA	34.53	32.097	30.073	30.687	30.374	33.561
48	C14:0	18.31	18.088	17.950	17.619	18.538	17.034
49	C15:0 iso	18.92	19.487	19.232	19.128	20.832	18.638

[a] The detailed description of abbreviations for all these fatty acid methyl esters is given in Ref. [1].

Table 2
The observed and calculated retention time (Rt) of fatty acid methyl esters – test and validation sets for the ANNs, PLS and MLR models (for ANNs and MLR: Test set no.: 1–9 and validation set no.: 10–15).

No.	Name[a]	Rt exp	Rt ANN-LM	Rt ANN-BFG	Rt ANN-CGP	Rt MLR	Rt PLS
1	C17:0 anteiso	22.65	22.573	22.489	22.660	25.576	22.94
2	C20:0	30.96	30.961	30.976	30.974	32.738	31.927
3	C10:0	14.28	14.305	14.328	14.28	9.0328	12.829
4	C18:1 n9*c*	26.96	27.514	27.369	27.301	30.006	30.412
5	C18:1 n4*c*	27.64	27.138	27.343	27.282	29.985	28.698
6	C18:2 n6*tc*	29.10	29.100	31.143	31.147	30.374	29.461
7	C22:5 n6*c*	49.80	50.718	51.124	51.133	47.526	49.598
8	C22:5 n3*c*	52.56	51.315	51.157	51.183	47.592	50.589
9	*c*11, *t*13-CLA	33.17	33.170	31.125	31.12	30.361	32.763
10	C16:0	21.38	20.712	20.895	20.948	23.258	21.274
11	C17:0 iso	22.21	22.519	22.541	22.564	25.601	22.983
12	C18:1 n11*t*	26.47	26.470	26.471	26.480	28.328	25.783
13	C22:1 n9*c*	39.98	39.976	39.984	40.122	39.516	40.822
14	C14:0 iso	17.66	18.172	17.906	17.858	18.464	16.147
15	C20:5 n3*c*	43.73	43.728	43.73	43.66	42.852	44.22

[a] The detailed description of abbreviations for all these fatty acid methyl esters is given in Ref. [1].

Table 3
Selected descriptors of genetic algorithm model.

Descriptor	Notation	Regression coefficient	Standard error	Mean effect
Number of *cis* double bond (constitutional descriptor)	NcDB	1.925	±0.347	0.031
Number of *trans* double bond (constitutional descriptor)	NtDB	0.260	±0.964	0.001
Gravitational index G2 (bond-restricted) (geometrical descriptor)	G2	4.140	±0.213	0.938
Constant		−22.721	±2.585	

$R^2_{training}$ = 0.918, $SE_{training}$ = 3.002, R^2_{test} = 0.940, SE_{test} = 2.889.

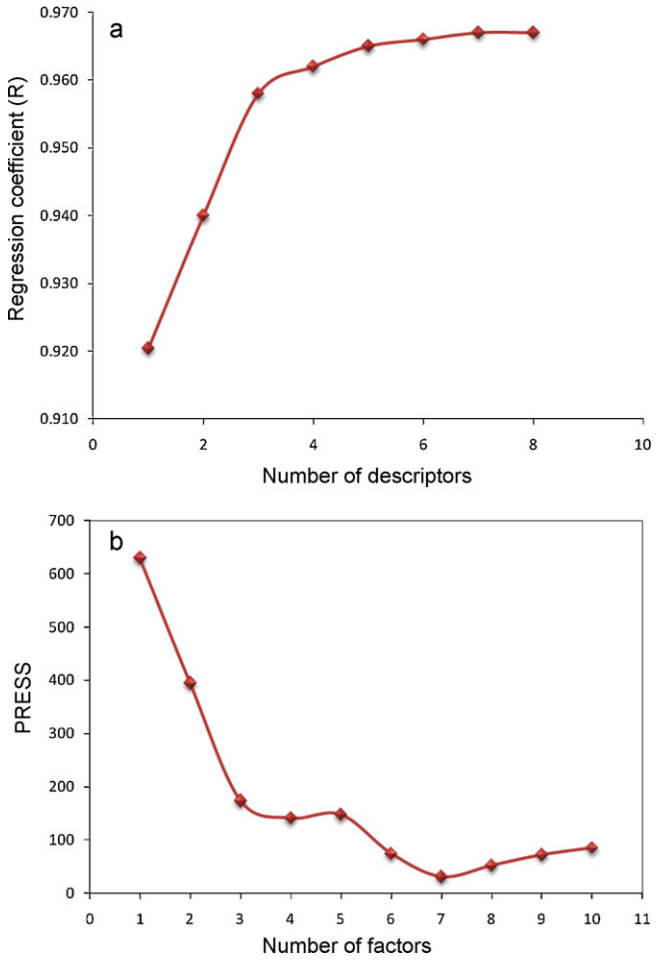

Fig. 1. (a) Influence of number of descriptors on R^2 for the GA model, (b) PRESS versus the number of factors for the PLS model.

be seen from the correlation matrix (Table 4) there is no significant correlation between the selected descriptors.

3.4. PLS

PLS is a linear modeling technique where information in the descriptor matrix *X* is projected onto a small number of underlying ("latent") variables called PLS components, referred to as latent variables. The matrix *Y* is simultaneously used in estimating the "latent" variables in *X* that will be most relevant for predicting the *Y* variables.

Table 4
Correlation matrix for the three selected descriptors (degree of freedom: 46).

	NcDB	NtDB	G2
NcDB	1		
NtDB	−0.186	1	
G2	0.267	−0.001	1

At the present work, the modeling by PLS method was performed using MINITAB 15. For regression analysis, data set was separated into two groups: training and prediction sets (Tables 1 and 2). The number of significant factors for the PLS algorithm was determined using the cross-validation method. With cross-validation, ten samples was kept out (leave ten out) of the calibration and used for prediction. The predicted values of left-out samples were then compared to the observed values using prediction error sum of squares (PRESSs). The PRESS obtained in the cross-validation was calculated each time that a new principal component (PC) was added to the model. The optimum number of PLS factors is the one that minimizes PRESS [32]. PRESS is defined as Eq. (9):

$$\text{PRESS} = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 \quad (9)$$

where $\hat{y}_i$ the estimated value of the *i*th is objects and y_i is the corresponding reference value of this object. Fig. 1b shows the plot of PRESS versus the number of factors for the PLS model. The best PLS model contained seven factors.

3.5. ANNs

The networks were generated using the three descriptors appearing in the GA models as their inputs and Rts as their output. For ANN generation, data set was separated into three groups: training, test and validation set (Tables 1 and 2). All molecules were randomly placed in these sets. The training set, consisted of 49 molecules, was used for the model generation. However, the test set, consisted of 9 molecules, was used to take care of the overtraining. The prediction set, consisted of 6 molecules, was used to evaluate the generated model.

A three-layer network with a sigmoid transfer function was designed for each ANN. Before training the networks, the input and output values were normalized between zero and one. The ANNs program was coded in MATLAB 7.1 for windows [27]. The network was then trained using the training set by the backpropagation strategy for optimization of the weights and bias values. The proper number of nodes in the hidden layer was determined by training the network with different number of nodes in the hidden layer.

The root-mean-square error (RMSE) value measures how good the outputs are in comparison with the target values. It should be noted that for evaluating the overfitting, the training of the network for the prediction of Rt must stop when the RMSE of the test set begins to increase while RMSE of calibration set continues to decrease. Therefore, training of the network was stopped when overtraining began. All of the above-mentioned steps were carried out using BFGS quasi-Newton backpropagation (BFG), conjugate gradient backpropagation with Polak–Ribiére updates (CGP) and Levenberg–Marquardt (trainlm) weight update functions.

3.6. Testing for chance correlations

Part of validating the models is to check for the possibility of chance correlations. This can be done by performing the entire sequence of computations over but with the dependent variables

scrambled. This scrambling destroys any relationship between the descriptors and the dependent variable. No model that exceeds chance performance should be found. The results obtained are compared to the results achieved with the actual computations to demonstrate that the actual results were achieved by finding relationships rather than by finding chance correlations [18].

3.7. Cross-validation technique

The consistency and reliability of a method can be explored using the cross-validation technique [33]. Two different strategies of leave-one-out (LOO) and leave-multiple-out (LMO) can be carried out in this method. In LOO strategy, by deleting each time one object from training set, a number of models will be produced. Obviously, the number of models produced by the LOO procedure is equal to the number of available examples n ($n = 64$). Prediction error sum of squares (PRESSs) is a standard index to measure the precision of a modeling method based on the cross-validation technique. Based on the PRESS and SSY (sum of squares of deviations of the experimental values from their mean) statistics, the Q^2_{LOO} can be easily calculated by Eq. (10):

$$Q^2_{LOO} = \frac{\text{PRESS}}{\text{SSY}} = 1 - \frac{\sum_{i=1}^{n}(y_{exp} - y_{pred})^2}{\sum_{i=1}^{n}(y_{exp} - \bar{y})^2} \quad (10)$$

In the case of LMO, M represents a group of randomly selected data points, which would leave out at the beginning and would be predicted by the model, which was developed using the remaining data points. So, M molecules are considered as prediction set. The R^2_{LMO} can be calculated by Eq. (11):

$$R^2_{LMO} = 1 - \frac{\sum_{i=1}^{test}(y_{exp} - y_{pred})^2}{\sum_{i=1}^{test}(y_{exp} - \bar{y}_{train})^2} \quad (11)$$

The higher the Q^2_{LOO} or R^2_{LMO} the higher the predictive power of the model [33].

4. Results and discussion

The main aim of the present work was developing a QSRR model to prediction of the retention times of fatty acid methyl esters in human blood. Chromatographic retention is based on interactions between the solute and the stationary phase, and the aim of the present work is to find which of the available topological, geometrical, constitutional, and physical descriptors that we computed are related to the retention of the FAMEs present in human blood. Therefore, the development of a robust and interpretable QSRR model, which is able to accurately predict the Rt, is necessary.

Genetic algorithm (GA) was used for the selection of the variables that resulted in the best-fitted models. Gravitational index (G2), number of *cis* double bond (N*c*DB) and number of *trans* double bond (N*t*DB) were selected among a large number of descriptors.

As first step, for the selection of the most important descriptors genetic algorithm was used which finally selected three descriptors whose specifications are given in Table 3. Due to existence of *cis* and *trans* isomers in studied fatty acid methyl ester we expected that the N*c*DB, N*t*DB and 3-dimensional descriptors play an important role for the prediction of retention times in the modeling. The number of double bonds can make a distinction between compounds with the same chain length. Furthermore, the methoxycarbonyl fragment is constant in each solute so we cannot expect the leading role of polarity and polarity related parameters. It can be seen from Table 3, three descriptors, including G2 (3D-descriptor), N*c*DB and N*t*DB (0D-descriptors), were chosen among 172 parameters. These descriptors can be classified as geometrical (G2) and constitutional (N*c*DB and N*t*DB) descriptors.

Gravitational index (G2) (bond-restricted) is a geometrical descriptor that reflecting the mass distribution in a molecule and defined as Eq. (12):

$$G_2 = \sum_{a=1}^{A}\left(\frac{m_i \cdot m_j}{r_{ij}^2}\right)_a \quad (12)$$

where m_i and m_j are the atomic masses of the considered atoms; r_{ij} the corresponding interatomic distances; and A the number of all pairs of bonded atoms of the molecule. This index is related to the bulk cohesiveness of the molecules, accounting, simultaneously, for both atomic masses (volumes) and their distribution within the molecular space. This index can be extended to any other atomic property different from atomic mass, such as atomic polarizability, atomic, van der Waals volumeetc. [34]. This descriptor has a positive coefficient in the linear model; therefore it indicates that the molecules with larger number of bonded atoms and lower interatomic distances are expected to bind more tightly to CGC column.

The N*c*DB and N*t*DB are other descriptors that selected with GA method. The positive coefficient of these descriptors, especially N*c*DB, in the model implies that existence of *cis* and *trans* double bonds in the structure of fatty acid methyl esters can lead to a longer Rt value in the column. In the other hand increasing the number of *cis* and *trans* double bonds in the studied compounds caused the longer Rt value.

The second step was developing of MLR model to assess the linear relationship between these descriptors and retention times. A value of 0.940 for R^2_p of this model reveals that it is able to account 94.0% of the variances of the Rt.

We have used the proposed linear model to interpret the mechanism of the retention time of fatty acid methyl esters. This means we should investigate the variables that are the most important predictors among the three descriptors appearing in the MLR model. In the case of the MLR, the mean effect of each descriptor can be considered as a measure of its role in predicting the retention time of FAMEs. Mean effect is defined as Eq. (13):

$$\text{MF}_i = \frac{\beta_j \sum_{i=1}^{i-n} d_{ij}}{\sum_{j}^{m} \beta \sum_{i=1}^{i-n} d_{ij}} \quad (13)$$

where MF_i represents the mean effect for the considered descriptor j, β_j is the coefficient of the descriptor j, d_{ij} stands for the value of the target descriptors for each molecule and, eventually, m is the descriptor number in the model. The MF value indicates the relative importance of a descriptor, compared with the other descriptors in the model. Its sign exhibits the variation direction in the values of the activities as a result of the increase (or reduction) of these descriptor values.

The mean effects for each variable in the MLR model are shown in Table 3. As can be seen from this table, G2 and N*c*DB are more important parameters than N*s*DB affecting the retention time of the molecules. Fig. 2(a) and (b) indicates the changes of these descriptors against the Rt values of the molecules. As can be seen from these figures, Rt values increases with increasing of N*c*DB in the structure of FAMEs with the same chain length, and also Rt values increases with increasing the G2 values.

For the sake of comparison, a PLS analysis was also performed using all 172 variables. However, seven latent variables were selected using PLS. Standard error for the training and prediction set using PLS was SE = 1.412 and SE = 1.369 and correlation coefficient was $R = 0.991$ and $R = 0.993$, respectively. The R^2 value of 0.987 for the prediction set reveals that this model is able to account 98.7% of the variances of the Rt. As can be seen from Table 5 the statistical parameters of PLS model are superior to MLR model. This is due to the collinearity problem in MLR analysis, while PLS regression can handle the collinear descriptors and therefore better predictive

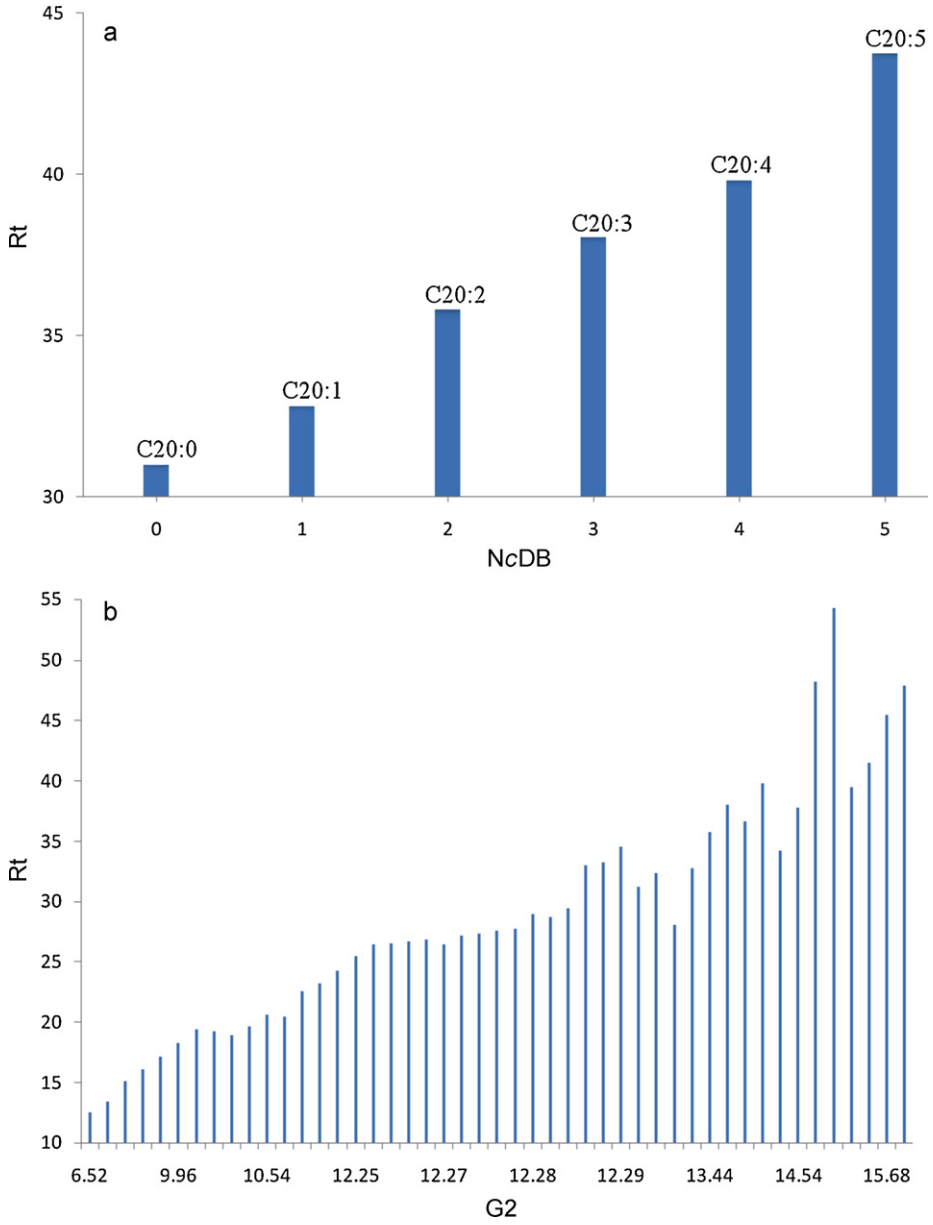

Fig. 2. (a) and (b) Plot of Rt values against the descriptor values: (a) NcDB, (b) G2.

models will be obtained [18]. The MLR and PLS calculated values of retention times for training and prediction sets are shown in Tables 1 and 2.

The next step was developing of the feed forward backpropagation artificial neural network using the descriptors appearing in the GA model as its inputs. It is a common practice to optimize the parameters of number of nodes in the hidden layer, learning rate and momentum in developing a reliable network. The procedure for optimizing these parameters are given elsewhere [35–50]. However, as it can be seen from Eq. (1), there is a term called weight update function, which indicates the way that weights are changed during the learning process. This paper focuses on investigating the role of weight update function. The statistical parameters for three different LM, BFG and CGP algorithms are given in Table 5; also the statistical parameters MLR and PLS models is shown in this table to compare the performance of the models. It can be seen from this table that the statistical parameters of LM-ANN model are superior to that of other models. Inspection of this table reveals the importance of the role of algorithms by which the weight update functions are considered. Therefore, a backpropagation network with a 3–2–1 architecture was developed using Levenberg–Marquardt algorithm (LM-ANN) to predict the retention times of FAMEs in human blood.

Table 5
The statistical parameters for ANN, MLR, and PLS models.[a]

Model	Training		Test		Validation	
	R^2	SE	R^2	SE	R^2	SE
LM-ANN	0.992	0.932	0.998	0.645	0.999	0.445
CGP-ANN	0.988	1.130	0.990	1.326	0.999	0.304
BFG-ANN	0.984	1.347	0.990	1.329	0.999	0.317
MLR	0.918	3.002	0.924	3.611	0.989	1.289
PLS	0.982	1.412	0.987	1.369	–	–

[a] Number factors of the PLS model are 6 and number of descriptors for MLR and ANN models are 3.

Table 6
Cross-validation results for LM-ANN and MLR models.

Model	R^2_{L10O}	R^2_{L12O}	R^2_{L15O}	Q^2_{LOO}
LM-ANN	0.989	0.994	0.997	0.981
MLR	0.812	0.768	0.732	0.765

It can be seen from Table 5 that the R^2 and SE values have improved from 0.918 and 3.002 for the MLR model to 0.992 and 0.932 for the LM-ANN model for the training set, respectively. It means that this non-linear model is able to account 99.2% of the variances of the capillary gas chromatographic retention times of FAMEs in human blood. The calculated values of Rt for training, test and validation sets using three different ANNs are shown in Tables 1 and 2.

In order to ensure the reliability of the proposed model we have also used leave multiple out-cross validation (LMO-CV). Based on this technique, a number of modified data sets were created by deleting in each step a small group of objects (here 10, 12 and 15 objects) and then the model was evaluated by measuring its accuracy in predicting the responses of the deleted group (the ones that have not been utilized in the development of the model). The results of L10O, L12O and L15O for MLR and LM-ANN methods are reported in Table 6. The consistency in the statistic of R^2 for different data sets of L10O, L12O and L15O reveals the stability and robustness of LM-ANN is superior models compared to the MLR model.

To further check the robustness of the LM-ANN model, the Y-randomization test was performed in this contribution. The dependent variable vector (Rt) was randomly shuffled and a new QSRR model was developed using the original independent variable matrix. The new QSRR model is expected to have low R^2 and high SE values. Several random shuffles of the *y* vector were performed and the results are shown in Table 7. The R^2 and SE values indicate that the good results for the LM-ANN model are not due to a chance correlation or structural dependency of the training set. The observed and LM-ANN predicted values of the Rt for all of the FAMEs studied in this work are shown in Tables 1 and 2. Fig. 3 demonstrates the plot of the LM-ANN predicted versus the experimental values of the Rt for the data set. A correlation coefficient of this plot indicates the reliability of the model. The residuals of the LM-ANN calculated values of Rt are plotted against the experimental values in Fig. 4. The propagation of the residuals on both sides of zero line indicates that no systematic error exists in the development of the LM-ANN model.

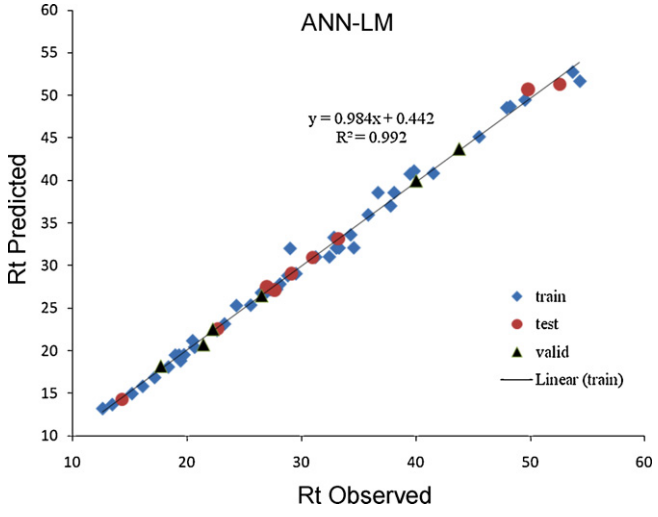

Fig. 3. Plot of experimental Rt values of fatty acid methyl esters against the calculated values for LM-ANN.

Table 7
Regression coefficient (R^2) and SE values for Y-randomization tests for LM-ANN and MLR models.

Iteration	LM-ANN		MLR	
	R^2	SE	R^2	SE
1	0.039	10.615	0.011	11.005
2	0.027	10.607	0.019	10.886
3	0.036	11.006	0.070	11.048
4	0.170	9.350	0.051	10.214
5	0.043	10.716	0.021	11.074
6	0.030	10.569	0.005	10.940
7	0.021	10.852	0.022	11.087
8	0.018	9.998	0.045	10.080
9	0.013	10.837	0.015	11.065
10	0.106	10.432	0.087	10.774

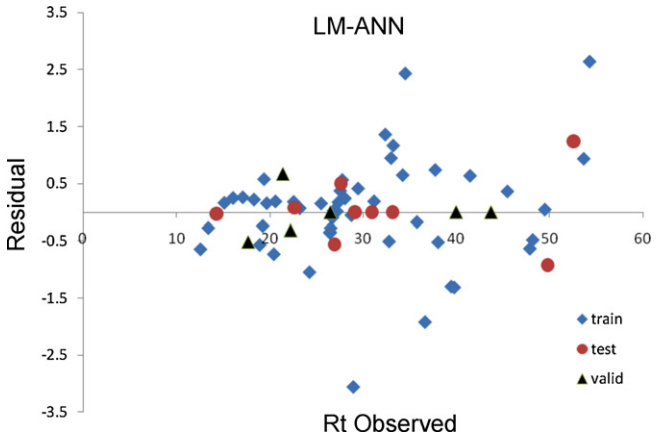

Fig. 4. Plot of residuals versus experimental Rt values for LM-ANN.

5. Conclusion

Fatty acid methyl ester derivatives including some pairs of isomers with similar structures but different retention behaviors were included to build QSRR models. Comparison of the values of statistical parameters obtained using models of ANNs (with three different weight update functions including LM-ANN, BFG-ANN and CGP-ANN), PLS and MLR for predicting of capillary gas chromatographic retention time of FAMEs shows superiority of the Levenberg–Marquardt back propagation network (LM-ANN) over those of non-linear and especially linear models. Obtained models indicated that geometrical (G2) and constitutional (N*c*DB and N*t*DB) descriptors, selected by using genetic algorithm method, have important role in retention times of studied compounds. By focusing on the role of the weight update function, we realized that the algorithms of weight update functions are important in the performance of the network. The main conclusion of this study is that Levenberg–Marquardt backpropagation artificial neural network with a 3–2–1 architecture is a reliable tool for the prediction of capillary gas chromatographic retention time of fatty acid methyl ester derivatives.

References

[1] B. Bicalho, F. David, K. Rumplel, E. Kindt, P. Sandra, J. Chromatogr. A 1211 (2008) 120.
[2] L. Hodson, C. Murray Skeaff, B.A. Fielding, Prog. Lipid Res. 47 (2008) 348.
[3] Y. Wanga, F.P. Kuhajdaa, J.N. Lia, E.S. Pizera, W.F. Hana, L.J. Sokolla, D.W. Chana, Cancer Lett. 167 (2001) 99.
[4] E. Benedettini, P. Nguyen, M. Loda, Diagn. Histopathol. 14 (2008) 195.
[5] S. Ogino, T. Kawasaki, A. Ogawa, G.J. Kirkner, M. Loda, C.S. Fuchs, Hum. Pathol. 38 (2007) 842.
[6] W. Zhou, W.F. Han, L.E. Landree, J.N. Thupari, M.L. Pinn, T. Bililign, E.K. Kim, A. Vadlamudi, S.M. Medghalchi, R.E. Meskini, G.V. Ronnett, C.A. Townsend, F.P. Kuhajda, Cancer Res. 67 (2007) 2964.

[7] E.S. Pizer, S.F. Lax, F.P. Kuhajda, G.R. Pasternack, R.J. Kurman, Cancer 83 (1998) 528.
[8] Claus Härtig, J. Chromatogr. A 1177 (2008) 159.
[9] N. Sánchez-Ávila, J.M. Mata-Granados, J. Ruiz-Jiménez, M.D. Luque de Castro, J. Chromatogr. A 1216 (2009) 6864.
[10] R. Kaliszan, Quantitative Structure–Chromatographic Retention Relationship, John Wiley & Sons, New York, 1987.
[11] H. Du, J. Wang, Z. Hu, X. Yao, Talanta 77 (2008) 360.
[12] K. Héberger, J. Chromatogr. A 1158 (2007) 273.
[13] O. Deeb, B. Hemmateenejad, A. Jaber, R. Garduno-Juarez, R. Miri, Chemosphere 67 (2007) 2122.
[14] M. Jalali-Heravi, M. Asadollahi-Baboli, P. Shahbazikhah, Eur. J. Med. Chem. 43 (2008) 548.
[15] Q. Shen, W.M. Shi, X.P. Yang, B.X. Ye, Eur. J. Med. Chem. Pharm. Sci. 28 (2006) 369.
[16] X.J. Yao, A. Panaye, J.P. Doucet, H.F. Chen, R.S. Zhang, B.T. Fan, M.C. Liu, Z.D. Hu, Anal. Chim. Acta 535 (2005) 259.
[17] J.Z. Li, H.X. Liu, X.J. Yao, M.C. Liu, Z.D. Hu, B.T. Fan, Chemometr. Intell. Lab. Syst. 87 (2007) 139.
[18] Z. Garkani-Nejad, B. Ahmadi-Roudi, Eur. J. Med. Chem. 45 (2010) 719.
[19] J. Hunger, G. Huttner, J. Comput. Chem. 20 (1999) 455.
[20] S. Ahmad, M.M. Gromiha, J. Comput. Chem. 24 (2003) 1313.
[21] U. Depczynski, V.J. Frost, K. Molt, Anal. Chim. Acta 420 (2000) 217.
[22] B.K. Alsberg, N. Marchand-Geneste, R.D. King, Chemometr. Intell. Lab. Syst. 54 (2000) 75.
[23] D. Jouanrimbaud, D.L. Massart, R. Leardi, O.E. deNoord, Anal. Chem. 67 (1995) 4295.
[24] O. Farkas, I.G. Zenkevich, F. Stout, J.H. Kalivas, K. Héberger, J. Chromatogr. A 1198 (2008) 188.
[25] P. Stefan, Niculescu, J. Mol. Struct. (Theochem) 622 (2003) 71.
[26] T.A. Masters, Practical Neural Network Recipes in C, Academic Press, San Diego, CA, 1993.
[27] MATLAB Version 7.1. Mathworks Inc., 2005. http://www.mathworks.com/products/matlab/.
[28] R. Leardi, R. Boggia, M. Terrile, J. Chemometr. 6 (1992) 267.
[29] Z. Daren, J. Comput. Chem. 25 (2001) 197.
[30] Hyperchem, Molecular Modeling System, Hyper Cube Inc., 1993.
[31] R. Todeschini, V. Consonni, A. Mauri, M. Pavan, Software Dragon, 2003, http://disat.unimib.it/chm/Dragon.htm.
[32] C.W. Karen, B.F. Alberico, Eur. J. Med. Chem. 43 (2008) 364.
[33] D.W. Osten, J. Chemometr. 2 (1998) 39.
[34] R. Todeschini, V. Consonni, Handbook of Molecular Descriptors, Wiley/VCH, Weinheim, 2000.
[35] M. Jalali-Heravi, M.H. Fatemi, J. Chromatogr. A 915 (2001) 177.
[36] V.K. Gupta, A. Rastogi, J. Hazard. Mater. 152 (2008) 407.
[37] A.K. Singh, V.K. Gupta, Barkha Gupta, Anal. Chim. Acta 585 (2007) 171.
[38] V.K. Gupta, A.K. Singh, Barkha Gupta, Anal. Chim. Acta 575 (2006) 198.
[39] V.K. Gupta, A. Mittal, L. Krishnan, J. Mittal, J. Colloid Interface Sci. 293 (2006) 16.
[40] V.K. Gupta, R. Ludwig, S. Agarwal, Anal. Chim. Acta 538 (2005) 213.
[41] A.K. Jain, V.K. Gupta, L.P. Singh, P. Srivastava, J.R. Raisoni, Talanta 65 (2005) 716.
[42] R. Prasad, V.K. Gupta, Azad Kumar, Anal. Chim. Acta 508 (2004) 61.
[43] V.K. Gupta, Rajni Mangla, S. Agarwal, Electroanalysis 14 (2002) 1127.
[44] V.K. Gupta, P. Kumar, Anal. Chim. Acta 389 (1999) 205.
[45] A.K. Jain, V.K. Gupta, L.P. Singh, Anal. Proc. Anal. Commun. 32 (1995) 263.
[46] A.K. Jain, V.K. Gupta, B.B. Sahoo, L.P. Singh, Anal. Proc. Anal. Commun. 32 (1995) 99.
[47] S.K. Srivastava, V.K. Gupta, M.K. Dwivedi, S. Jain, Anal. Proc. Anal. Commun. 32 (1995) 21.
[48] S.K. Srivastava, V.K. Gupta, S. Jain, Electroanalysis 8 (1996) 938.
[49] A.K. Jain, V.K. Gupta, U. Khurana, L.P. Singh, Electroanalysis 9 (1997) 857.
[50] V.K. Gupta, A.K. Jain, L.P. Singh, U. Khurana, Anal. Chim. Acta 355 (1997) 33.