

# Support Vector Regression Based QSPR for the Prediction of Retention Time of Peptides in Reversed-Phase Liquid Chromatography

Hassan Golmohammadi · Zahra Dashtbozorgi ·  
Yvan Vander Heyden

Received: 15 September 2014 / Revised: 18 November 2014 / Accepted: 26 November 2014 / Published online: 9 December 2014  
© Springer-Verlag Berlin Heidelberg 2014

**Abstract** In the present study, Quantitative Structure–Property Relationship (QSPR) models were developed to investigate the retention times ( $t_R$ ) of various peptides in seven reversed-phase liquid chromatography systems using Partial Least Squares (PLS), Artificial Neural Network (ANN) and Support Vector Machine (SVM) techniques. Different types of molecular descriptors were calculated to represent the molecular structures of the various compounds studied. Important descriptors were selected by a Genetic Algorithm-Partial Least Square (GA-PLS) method. The four descriptors selected using GA-PLS were used as inputs for PLS, ANN and SVM to build models to predict the retention times. Our study reveals that the relation between the chemical properties and retention time is a nonlinear phenomenon and that the PLS method is not capable to properly model it. The results obtained demonstrate that, for all seven data sets, the  $t_R$  values estimated by SVM were in good agreement with the experimental data, and the performances of the SVM models were comparable or superior to those of PLS and ANN.

**Electronic supplementary material** The online version of this article (doi:10.1007/s10337-014-2819-1) contains supplementary material, which is available to authorized users.

H. Golmohammadi (✉)  
Young Researchers and Elite Club, Shahr-e-Rey Branch,  
Islamic Azad University, Tehran, Iran  
e-mail: Hassan.gol@gmail.com

Z. Dashtbozorgi  
Young Researchers and Elite Club, Central Tehran Branch,  
Islamic Azad University, Tehran, Iran

Y. Vander Heyden  
FABI, Department of Analytical Chemistry and Pharmaceutical  
Technology, Pharmaceutical Institute Vrije Universiteit Brussel  
(VUB), Laarbeeklaan 103, 1090 Brussels, Belgium

**Keywords** Reversed-phase liquid chromatography ·  
Peptides · Retention time · Quantitative structure–property  
relationship · Support vector machine

## Introduction

Proteins belong to the most significant biologically active substances. Acting as hormones, neurotransmitters, immunomodulators, coenzymes, enzyme substrates and inhibitors, receptor ligands, drugs, toxins, and antibiotics they play an important role in controlling and regulating many critically essential procedures in living organisms. Additionally, to understand living cell functioning, an inclusive exploration of the whole protein set of a cell will be necessary [1, 2]. As a result, degradation of proteins to peptides, separation and analysis of peptides are becoming progressively more important in proteomics.

High-performance liquid chromatography (HPLC) is now powerfully established as the foremost technique for the analysis and separation of an extensive range of molecules. Especially, HPLC in its different modes has become the fundamental technique in the description of peptides and proteins and peptide separation and has, consequently, played a key role in the fast advances in the biological and biomedical sciences over the last years [3].

Reversed-phase liquid chromatography (RPLC) is possibly the most regularly used mode of separation for peptides, although ion-exchange (IEC) and size-exclusion chromatography (SEC) also find applications. The three-dimensional structure of proteins can be perceptive to the often cruel conditions employed in RPLC, and consequently, RPLC is utilized less for the isolation of proteins where it is important to recover the protein in a biologically active form [4]. RPLC is a very influential method

for the analysis of peptides and proteins for a number of reasons that include: (1) the excellent resolution that can be attained under a wide range of chromatographic conditions for very intimately related molecules as well as structurally quite distinct molecules; (2) the experimental simplicity with which chromatographic selectivity can be influenced through changes in mobile phase characteristics; (3) the generally high recoveries and, hence, high productivity; and (4) the excellent reproducibility of repetitive separations carried out over a long period of time, which is caused partially by the stability of the sorbent materials under a wide range of mobile-phase conditions [5].

RPLC involves the separation of peptides on the basis of hydrophobicity. The separation depends on hydrophobic binding of the solute molecule from the mobile phase to the immobilized hydrophobic ligands attached to the stationary phase. The mobile phase composition and the pressure are two essential factors which influenced the separation of peptides. The retention pattern of a peptide changes as the mobile phase composition and the column pressure change. The retention of peptides ( $\log K$ ) do not vary linearly with the mobile phase but do follow a quadratic relationship [6].

Regardless of the ever increasing usage of HPLC for the separation and analysis of peptides and proteins, selection of the chromatographic conditions is still found by time-consuming trial-and-error methods. A priori knowledge of the retention time of a given peptide on a given chromatographic system would help in the selection of proper chromatographic conditions. Currently, prediction of the retention behavior of peptides is mainly rooted in the amino acid composition [7–10]. However, using this technique, some experiments for the standard samples must be achieved to derive the group retention coefficients of the amino acid in the given conditions, which is still time-consuming and is difficult to generalize the calculated results.

Quantitative Structure–Property Relationship (QSPR) studies, which relate descriptors of the molecular structure to properties of chemical compounds, have proved to be successful in predicting retention times of peptides [11]. The advantage of this technique over other predictive methods lies in the fact that the descriptors used can be computed exclusively from structural considerations and do not rely on experimental properties as input parameters. Once the structure of given compound is known, one can compute a larger number of different molecular and geometric descriptors. Therefore, once a reliable model is derived, one can use the model to estimate the property of a compound, whether or not the compound already has been synthesized [12]. In the actual study, closeness between predicted and experimental retention times will help in the future identification of peptides.

Although QSPR methods have been effectively used to forecast many physicochemical properties, only a small

number of research groups have investigated the quantitative correlation between the structural parameters and the chromatographic retention of peptides; This might be due to the problematic optimization of the peptides structures which is very time-consuming because in most of the cases, the size of the peptides is rather large. Liu et al. [13] developed a QSPR model for the prediction of the capacity factors of 75 peptides based on Support Vector Machine and the Heuristic Method. Petritis and co-workers [14] used the Genetic Algorithm and Artificial Neural Network (ANN) techniques for the prediction of peptide liquid chromatography elution times in proteome analyses. Ma et al. [15] predicted electrophoretic mobilities of peptides in capillary zone electrophoresis (CZE) using the Linear Heuristic Method (HM) and a Nonlinear Radial Basis Function Neural Network (RBFNN). Shinoda et al. [16] developed a computational method to predict the retention times of peptides in HPLC using Multiple Linear Regression (MLR) and ANN. Du et al. [17] generated Quantitative Structure–Retention Relationship (QSRR) models to correlate retention times of peptides in reversed-phase liquid chromatography to their structures based on linear and non-linear modeling methods. They used MLR for a linear QSRR model and Radial Basis Function Neural Networks (RBFNN) and Projection Pursuit Regression (PPR) for the nonlinear modeling. Put et al. [18] estimated the retention times of a set of peptides based on PLS regression and Uninformative Variable Elimination PLS (UVE-PLS) models.

Vapnik and Cortes have worked on a new computational classification method called Support Vector Machine (SVM) [19, 20]. SVM has been extended to solve regression problems, and has shown great performance in QSPR studies due to its remarkable ability to interpret the nonlinear relationships between molecular-structure descriptions and properties [21–26].

In this work, SVM was performed for modeling and predicting the retention times of various peptides using different kinds of molecular descriptors. The main goal was to generate a QSPR model that could be employed for the prediction of  $t_R$  of a diverse set of peptides from their molecular structures and to show the flexible modeling ability of SVM and at the same time, to seek the important structural features related to the retention times of peptides. PLS and ANN methods were also employed to generate quantitative linear and nonlinear models to compare with those obtained by SVM.

SVM feature mapping technique was used for the prediction of retention time values of a large set of peptides with different molecular structures. This is a simple, sensitive and inexpensive method that can accurately predict the chemical property such as retention time. The proposed model could identify and provide some insight into what

calculated descriptors related to retention time. SVMs-based modeling methods could produce more accurate QSPR models compared to linear regression methods, since they have the ability to handle the possible nonlinear relationships during the training process.

## Materials and methods

### Data Set

The data set of retention times of 93 peptides with known amino acid composition was extracted from the values reported by Put and Vander Heyden [11]. The retention times of the peptides were measured on seven RP chromatographic systems (CS1–CS7) [27]. The following columns were selected: CS1, XTerra MS C18 (Waters, Millford, MA, USA; 15.0 × 0.46 cm id); CS2, LiChrospher RP-18 (Merck, Darmstadt, Germany; 25.0 × 0.46 cm id); CS3, LiChrospher CN (Merck; 10.0 × 0.46 cm id); CS4, Discovery HS F5-3 (Supelco, Bellefonte, PA, USA; 15.0 × 0.46 cm id) with a silica-based pentafluorophenylpropylsilane stationary phase; CS5, Discovery RP-Amide C16 (Supelco; 15.0 × 0.46 cm<sup>2</sup> id); CS6, Chromolith (Merck; 10.0 × 0.46 cm id), a monolithic silica column; CS7, PLRP-S (Polymer Laboratories, Amherst, MA, USA; 15.0 × 0.41 cm id) with a crosslinked polystyrene/divinylbenzene stationary phase. The retention times of all molecules included in the data set were obtained under the same conditions. For all systems, the operating temperature was constant at 40 °C; the flow rate 1 mL min<sup>-1</sup> and the detection wavelength 223 nm. The data set was randomly divided into three separate sections, the training, test and validation sets, consisting of 55, 19, and 19 members, respectively. The training set was used to adjust the parameters of models, the test set to prevent the network on model from over-fitting and the external validation set to evaluate the prediction abilities of the constructed models.

### Descriptor Calculation and Reduction

The primary step to acquire a QSPR model is to encode the structural features of molecules, which are named molecular descriptors. The QSPR model performance and the accuracy of the results are strongly dependent on the way the structural representation is carried out. In the first step, all structures were drawn with the HyperChem (Ver. 7.0) program [28] and then pre-optimized using MM + molecular mechanics force field. A more precise optimization is then done with the semiempirical PM6 method in Mopac (2009) [29]. All calculations were carried out at the restricted Hartree–Fock level with no configuration interaction. The molecular structures were optimized using the

Polak–Ribiere algorithm until the root–mean–square gradient was 0.001. In a next step, the Hyperchem and Mopac output files were used by the Dragon package (Version 3) to calculate molecular descriptors [30]. Overall more than 1,400 theoretical descriptors were calculated for each molecule by this software. These descriptors can be categorized into several groups, 0D, constitutional descriptors; 1D, functional groups, atom-centered fragments, empirical descriptors and molecular properties; 2D, topological descriptors, molecular walk counts, BCUTs descriptors, Galvez topological charge indices, 2D autocorrelations; 3D, aromaticity indices, Randic molecular profiles from the geometry matrix, geometrical, RDF, 3D-MORSE, WHIMs, and GETAWAYs descriptors.

The calculated descriptors were first analyzed for the existence of constant or near constant variables. The detected ones were then removed. Besides, to decrease the redundancy existing in the descriptors data matrix, the descriptors' correlation with each other and with the property of the molecules was examined and the collinear descriptors (i.e.  $R > 0.9$ ) were detected. Among the collinear descriptors, the one presenting the highest correlation with the property was retained and the others removed from the data matrix. A total of 158 out of 521 descriptors showed high correlation and were removed from the next generation. Subsequently genetic algorithm-partial least squares (GA-PLS) variable subset selection method was used for selection of important descriptors.

### GA-PLS Variable Selection

One of the problems in selecting the set of molecular descriptors is their collinearity even though the most collinear are already removed. Second, models based on a reduced number of descriptions are simpler and better. To overcome these problems some approaches join the feature-selection technique Genetic Algorithms with Partial Least Squares [31–33]. GA-PLS consists of three basic steps. (1) Construction of a preliminary population of chromosomes in which each chromosome is a binary bit string by which the existence of a variable is symbolized; (2) Assessment of fitness of each chromosome in the population by the internal predictivity of PLS. (3) Reproduction of the population of chromosomes in the next generation. The operations of selection, cross-over and mutation of chromosomes, are made in this step. Then, steps 2 and 3 are repeated until the number of the repetitions has reached the designated number of generations.

In this paper, we use Leardi's GA-PLS method [34]. The values of empirical parameters affecting the performance of GA-PLS were defined as in Table 1. To obtain more reliable results, the GA process was repeated at least ten times. If some variables are present only in one model, it can be

**Table 1** Parameters of the GA-PLS approach

Population size	30 Chromosomes
Regression method	PLS
Maximum number of variables selected in the same chromosome	30
Maximum number of components	The optimal number
Response	Cross-validated % explained variance
Probability of mutation	0.01
Probability of cross over	0.5
Number of evaluations	200
Number of runs	100

concluded that they have been selected just by chance, and consequently, they can be disregarded in the final model.

### Partial Least Squares Regression

The general principle of a linear regression method is to quantify the relationship between several independent or predictive variables and a dependent variable. Independent or predictive variables could be diverse physicochemical descriptors of the molecules, their principal components or other latent variables. The Partial Least Squares method is used to establish relationships between the dependent variable of the  $y$  vector and the descriptors of the  $X$  matrix [35]. PLS can analyze data with collinear, noisy, and numerous variables in both  $X$  and  $y$  [36]. PLS decreases the dimension of the predictor variables by extracting factors or latent variables that are correlated with  $y$  while capturing a large amount of the variation in  $X$ . This means that PLS maximizes the covariance between  $X$  and  $y$ . In PLS, the scaled matrices  $X$  and  $y$  are decomposed into score vectors ( $t$  and  $u$ ), loading vectors ( $p$  and  $q$ ), and residual error matrices ( $E$  and  $F$ ):

$$X = tp^T + E \quad (1)$$

and

$$Y = uq^T + F \quad (2)$$

The PLS algorithm used in this study was the singular value decomposition (SVD)-based PLS. This algorithm was proposed by Lorber et al. in 1987 [37]. A discussion of the SVD-based PLS algorithm can be found in the literature [38–40]. The program of PLS modeling based on SVD was in-house written in MATLAB 7 [41].

### Artificial Neural Networks

ANN can be defined as structures comprised of tightly interconnected adaptive simple processing elements or units that are able of performing especially parallel

calculations for data processing and knowledge representation. A detailed description of the theory behind a neural network has been adequately described elsewhere [42–44]. Therefore, only the items/points relevant to this work are described here. An ANN consists of some connected neurons and process information. A network consists of one input layer, one output layer and may also contain some hidden layers. Each layer contains some neurons connected to other neurons in previous and/or next layers. A neuron has an input, an output and a transfer function. The Sigmoidal transfer function,  $f(x)$ , is one of the performed functions, expressed as the following equation:

$$f(x) = \frac{1}{(1 + e^{-x})} \quad (3)$$

The output of node  $j$ ,  $O_j$ , is given by Eq. (4):

$$O_j = f \left[ \sum w_{ij} O_i + b_j \right] \quad (4)$$

where  $O_i$  is the output of  $i$ th neuron from the previous layer,  $w_{ij}$  presents the weights applied to the connection of neurons  $i$ th and  $j$ , and  $b_j$  is a bias term.

A feed-forward neural network consists of three layers. The first layer (input layer) includes nodes and acts as an input buffer for the data. Signals introduced to the network, with one node per element in the sample data vector, pass through the input layer to the layer called the hidden layer. Each node in this layer sums the weighted inputs and forwards them through a transfer function to the output layer. In the output layer, the processes of summing and transferring are repeated. The output of this layer now represents the calculated value for the node  $k$  of the network.

An ANN is an adaptive network that changes its structure based on external or internal information that flows through the network during the learning (training) phase. Training is performed by repeatedly presenting the network with identified inputs and outputs, and adjusting the connection weights and biases between the individual nodes. This process is repeated until the output nodes of the network match the preferred outputs to a stated degree of accuracy. Training can, for instance, be done using the back-propagation algorithm. To train the network using the back-propagation algorithm, the differences between the ANN output and its desired value are calculated after each iteration.

In the present work, an in-house ANN program was written in MATLAB 7. This network was feed-forward fully connected and has three layers with sigmoidal transfer functions. Descriptors selected by the GA-PLS method were used as inputs to the network and the output signal represents the retention times of the peptides. Thus, this network has four nodes in the input layer and one node in the output layer. The output of the sigmoid function is

in the range between 0 and 1 (dynamic range). Therefore, the value of each input (description) value was divided by the mean description value to bring them into the dynamic range of the sigmoidal transfer function of the network. The initial values of the weights were randomly selected from a uniform distribution that ranged between  $-0.3$  and  $+0.3$  and the initial values of the biases were set to be 1. During training, the network parameters are optimized. These parameters are: number of nodes in the hidden layer, weights and biases learning rates and the momentum. To evaluate the performance of ANN, standard error of training (SET) and standard error of prediction (SEP) were used. Then the network was trained using the training set by back-propagation strategy for optimization of the weights and biases values. It should be noted that it is common to plot the SET versus the number of iterations for optimization of ANN parameters.

### Support Vector Machine

The Support Vector Machine is an algorithm developed by the machine learning community. Owing to its unexpected generalization performance, the SVM has attracted attention and obtained a broad application range, such as pattern recognition problems [45, 46], drug design [47], Quantitative Structure–Activity Relationship [48], and QSPR analysis [49].

Support Vector Machines were developed by Vapnik, and the method is becoming more broadly known because of its many attractive features and promising empirical performance [18, 19, 50, 51]. The methodology discloses the Structural Risk Minimization (StRM) principle [52, 53] which has been exposed to be better than the conventional Empirical Risk Minimization (ERM) principle, employed by conservative neural networks.

A training set of  $m$  compounds with known properties or activities  $y_i$  and structurally consequent descriptors  $x_i$  are represented as  $\{(x_i, y_i)\}_{i=1}^m$  where correlations between structure and properties or activities are defined by  $y_i = f(x_i)$ . The term  $f(x_i)$  can be characterized by a linear function of the form:

$$f(x_i) = \langle w_i, x_i \rangle + b \tag{5}$$

where  $w$  identifies the weight vector of the linear function and  $b$  communicates to the threshold coefficient. SVM approximates the set of data with a linear function that is formulated in the high-dimensional feature space with the following function:

$$y = \sum_{i=1}^m w_i \phi(x_i) + b \tag{6}$$

where  $\{\phi_i(x_i)\}_{i=1}^m$  represents the features of input variables subjected to kernel transformation, while  $\{w_i\}_{i=1}^m$  and  $b$  are coefficients.

SVM is essentially a linear learning approach that was initially proposed for classification problems. However, it is also suitable to regression problems through the use of the  $\varepsilon$ -insensitive loss function. SVM can manage data possessing non-linear relationships by means of the so-called kernel trick. Kernel transformation is fundamentally a projection of the descriptor matrix from the input space into the higher-dimensional feature space. This can be achieved by the following equation:

$$k(x, y) = \langle \phi(x) \times \phi(y) \rangle \tag{7}$$

where  $k$  is a kernel function and  $\phi$  is a mapping from input space  $X \in x$  to the feature space  $F$ . A number of kernel functions are accessible for non-linear transformation of the input space. Popular kernel functions used in SVM include the variance–covariance-based linear and polynomial kernels, and the Euclidean distance-based radial basis function kernels.

A radial basis function kernel as illustrated by the following equation was employed to perform the non-linear mapping:

$$K(x, y) = \exp(-\gamma \|x - y\|^2) \tag{8}$$

After kernel transformation, the new feature space permits the data to be linearly distinguishable by hyperplanes where the hyperplane that maximizes the distance between the data samples was selected by the algorithm as the maximal hyperplane.

Minimization of the regularized risk function (Eq. 9) realizes two important properties of SVMs by means of estimating coefficients  $w$  and  $b$ : (i) identify regression assessment by performing risk minimization regarding the  $\varepsilon$ -insensitive loss function, and (ii) perform risk minimization derived from the StRM principle in which elements of the structure are defined by the disparity  $\|w\|^2 \leq \text{constant}$ .

The regularized risk function is defined as:

$$R(C) = C \frac{1}{N} \sum_{i=1}^N L_\varepsilon(y, f(x_i, w)) + \frac{1}{2} \|w\|^2 \tag{9}$$

where  $C \frac{1}{N} \sum_{i=1}^N L_\varepsilon(y, f(x_i, w))$  is the empirical error (risk) and  $\frac{1}{2} \|w\|^2$  is a measure of function flatness. The empirical error is measured by the  $\varepsilon$ -insensitive loss function  $(y, f(x_i, w))$  in which errors below  $\varepsilon$  would not be penalized. The punishment parameter  $C$  is a regularized constant responsible for determining the trade-off between the empirical error and the model complexity.

The estimation performance of SVM regression models is determined by the  $\varepsilon$ -insensitive loss function as follows:

$$L_{\varepsilon}(y, f(x, w)) = \begin{cases} |y - f(x, w)| - \varepsilon & \text{for } |y - f(x, w)| \geq \varepsilon \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

The parameter  $\varepsilon$  is referred to as the tube size, and it is defined as the approximation accuracy placed on the training data points. Basically, the purpose of support vector regression is to decide a function  $f(x)$  such that there is at most  $\varepsilon$  deviation from the actual value  $y_i$  for all training data while being as flat as possible. In other words, the loss function ignores errors as long as it is less than  $\varepsilon$  but would not accept considerable deviations from it.

#### Estimation of the Predictive Ability of a QSPR Model

For the estimation of the fitting and predictive abilities of a QSPR model, often the Fischer's ( $F$ ) test, the correlation coefficient of the experimental versus fitted/predicted properties ( $R$ ), the root mean squared error of calibration (RMSEC), the root mean squared error of prediction (RMSEP) and the root-mean squared error of cross-validation (RMSECV) are used. The latter are calculated using the following equations.

$$\text{RMSEC} = \sqrt{\frac{\sum_{i=1}^c (y_{\text{pred}_i} - y_{\text{obs}_i})^2}{c}} \quad (11)$$

where  $y_{\text{obs}_i}$  is the observed property (retention time) of a calibration (training) set object,  $y_{\text{pred}_i}$  is the predicted property of a calibration set object, and  $c$  is the number of samples in the calibration set.

$$\text{RMSEP} = \sqrt{\frac{\sum_{i=1}^t (y_{\text{pred}_i} - y_{\text{obs}_i})^2}{t}} \quad (12)$$

where  $y_{\text{obs}_i}$  is the observed property of a test set object,  $y_{\text{pred}_i}$  is the predicted property of a test set object, and  $t$  is the number of samples in the test set.

The root mean squared error of cross validation (RMSECV) is defined as in Eq. (13),

$$\text{RMSECV} = \sqrt{\frac{\sum_{i=1}^v (y_{\text{valid}_i} - y_{\text{obs}_i})^2}{v}} \quad (13)$$

where  $y_{\text{obs}_i}$  is the observed property of a validation set object,  $y_{\text{valid}_i}$  is the predicted property of a validation set object and  $v$  is the number of samples in the validation set.

The predictive ability of the calibration models from samples that were not used to generate the calibration equation was recorded as RMSECV. The RMSECV is regarded as the indication of the accuracy of calibration models when there are sufficient validation samples.

## Results and Discussion

### Descriptor Selection with GA-PLS

The GA-PLS procedure was performed on the data set to choose the most favorable set of descriptors. Because the GA is principally a stochastic algorithm, the results of various GA applications would accordingly be a little different. With the purpose of obtaining more consistent results (more reliable models); the GA process is repeated several times. In the present study, for each data set of the GA process was repeated 100 times and the selection of the variables were based on their frequency of incidence in the models, with maximal Cross-validated explained variance (C.V. %) attained for each operation. In this procedure, the chromosome and its fitness in the species correspond to a set of variables and internal prediction of the derived PLS model, respectively. Selection of useful variables is based on their frequency of occurrence in the best models obtained for each program. The frequency was calculated by the following equation:

$$\text{Frequency } (i) = \frac{\text{the total number of descriptor } (i) \text{ selected by GA-PLS}}{\text{the times of operation using GA-PLS}} \quad (14)$$

where  $i$  is the  $i$ th descriptor. The fitness of the individuals indicates the prediction power of the selected descriptors. The final model is picked via a stepwise regression, and the variables are selected in terms of their frequency. The descriptors with a high frequency were considered as more essential in describing the molecular structural properties which have the most imperative contribution to the overall retention times. Descriptors with a frequency above 90 % in 10 operations were selected. Parameters of the Genetic Algorithm for the generation of GA-PLS are shown in Table 1. With this approach, a set of four descriptors (see Table 2) was chosen for each data set and used to create the PLS, ANN and SVM models. For all of the seven datasets, the same descriptors were selected by GA-PLS. Their differences were only in their coefficients of regression. The specification of each model was described in Table 2.

### PLS Modeling

Table S1 gives the retention times on all seven CSs for all molecules. The PLS predicted values of the retention times for all peptides are shown in Table S2. Table 2 shows the regression coefficients of the four descriptors for the best PLS models. The optimum number of latent variables to be included in the model was three. The four descriptors in the model are: structural information content (neighborhood symmetry of 3-order) (SIC3), Geary autocorrelation

**Table 2** The partial least squares regression coefficients

CS	Coefficients				
	SIC3	GATS2P	BELM1	NCP	Constant
CS1	-18.66	-10.62	31.01	0.30	-18.26
CS2	-15.91	-12.44	31.10	0.32	-20.34
CS3	-20.14	-11.64	27.33	0.22	-23.15
CS4	-14.60	-13.31	34.46	0.34	-23.74
CS5	-18.54	-12.26	29.58	0.25	-21.73
CS6	-16.08	-14.46	29.25	0.20	-16.10
CS7	-15.94	-13.59	30.60	0.24	-17.13

CS chromatographic system, SIC3 structural information content (neighborhood symmetry of 3-order), GATS2P Geary autocorrelation lag 2/weighted by atomic polarizabilities, BELM1 lowest eigenvalue n. 1 of Burden matrix/weighted by atomic masses, NCP number of total primary carbons

**Table 3** Correlation matrix for the four descriptors

Descriptors	Descriptors			
	SIC3	GATS2P	BELM1	NCP
SIC3	1	0.085	-0.201	-0.501
GATS2P		1	-0.359	0.140
BELM1			1	0.228
NCP				1

lag 2/weighted by atomic polarizabilities (GATS2P), lowest Eigenvalue n. 1 of Burden matrix/weighted by atomic masses (BELM1) and number of total primary carbons (NCP). Table 3 represents the correlation matrix for these descriptors. As shown in this table there is not any significant correlation between the selected descriptors.

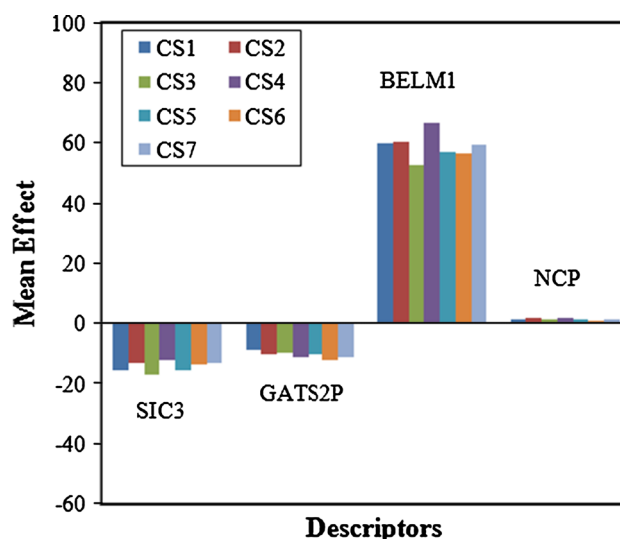
For evaluation of the relative significance and contribution of each descriptor in the models, the mean effect (ME) value was calculated for each descriptor by the following equation:

$$ME_j = \frac{\beta_j \sum_{i=1}^n d_{ij}}{\sum_j^m \beta_j \sum_{i=1}^n d_{ij}} \quad (15)$$

where  $ME_j$  is the mean effect for the considered descriptor  $j$ ,  $\beta_j$  is the coefficient of descriptor  $j$ ,  $d_{ij}$  is the value of descriptors for each molecule, and  $m$  is the number of descriptors in the model. The calculated ME values are plotted in Fig. 1.

#### The Influence of Each Descriptor on Retention Time

It is well-known that the chromatographic retention time can be considered as a chemical structure-dependent parameter, which is constant for any peptide in a defined

**Fig. 1** Plot of descriptor's mean effects

separation conditions including mobile phase composition, stationary phase, pH, temperature. At the constant separation condition, amino acid composition, peptide chain length and sequence, (generally structure of peptide) play essential role on retention time. Therefore, we focus on the descriptors which encode the structural features of peptides.

The first descriptor, according to its mean effect, is the lowest eigenvalue n. 1 of Burden matrix/weighted by atomic masses (BELM1). This BCUT descriptor is an expansion of parameters initially developed by Burden [54]. The Burden parameters are derived from a combination of the atomic number for each atom and a description of the nominal bond-type for adjacent and nonadjacent atoms. They may include connectivity information and atomic properties that are relevant to intermolecular interactions. The BCUT descriptors expand a number and types of atomic features that can be considered, and also supply a diversity of proximity measures and weighting schemes. These descriptors can be generated, depending on the choices of connectivity and atomic information, and on the scaling factors controlling the relative balance of these two kinds of information. It can capture sufficient structural features of molecules to yield useful measurement of molecular diversity. These descriptors designed to encode atomic properties that govern intermolecular interactions. The positive coefficient for BELM1 in PLS model indicates that an increase in the value of this descriptor leads to increase in the value of retention time. The increase of this descriptor reduce the solute-solvent interactions and therefore increasing the dispersive interactions with the stationary phase and consequently increasing the value of  $t_R$ .

The second descriptor is the structural information content (neighborhood symmetry of 3-order) (SIC3). This topological descriptor represents a measure of the graph complexity and is calculated as follows [55]:

$$\text{SIC}_r = \frac{\text{IC}_r}{\log_2 A} \quad (16)$$

where  $A$  is the number of atoms and  $\text{IC}_r$  is the information content index (neighborhood symmetry of 1-order), which is calculated as follows:

$$\text{IC}_r = - \sum_{g=1}^G \frac{A_g}{A} \times \log_2 \frac{A_g}{A} = - \sum_{g=1}^G P_g \times \log_2 P_g \quad (17)$$

where  $g$  runs over the  $G$ .  $G$  is number of equivalence classes (i.e. the number of different amino acid residues),  $A_g$ , is the cardinality of the  $g$ th equivalence class,  $A$  is the total number of atoms, and  $P_g$  is the probability of randomly selecting a vertex of the  $g$ th class. This descriptor gives us information on how many atoms with a similar connectivity pattern we have in the molecule. The descriptor is dependent on the number of atoms involved in the molecule, and it arranges the molecules in the order of rising chain length and number of the substituents of peptides. This descriptor describes the difference of the hydrophobicity and steric property of the solute comprehensively. As the hydrophobic and steric interaction is the main interaction between the solute and the stationary phase, this descriptor plays an important role in the elution process and has positive correlation with the  $t_R$ .

The next descriptor is the Geary autocorrelation lag 2/ weighted by atomic polarizabilities (GATS2P). This 2D Autocorrelation descriptor in general explains how the considered property is distributed along the topological structure and is defined as:

$$\text{GATS2P}(d) = \frac{\frac{1}{2\Delta} \times \sum_{i=1}^A \sum_{j=1}^A \delta_{ij} \times (w_i - w_j)^2}{\frac{1}{(A-1)} \times \sum_{i=1}^A (w_i - \bar{w})^2} \quad (18)$$

where  $w_i$  is an atomic property, and  $\bar{w}$  its average value on the molecule,  $A$  is the number of atoms,  $d$  the considered topological distance (i.e. the lag in the autocorrelation terms),  $\delta_{ij}$  a Kronecker delta ( $\delta_{ij} = 1$  if  $d_{ij} = d$ , zero otherwise) and  $\Delta$  is the sum of the Kronecker deltas, i.e. the number of vertex pairs at distance equal to  $d$  [56]. Autocorrelation descriptor calculated for molecular geometry are based on interatomic distances collected in the geometry matrix and the property function is defined by the set of atomic properties. The 2D Autocorrelation descriptors in general explain how the considered property is distributed along the topological structure. Increase of this descriptor will enhance the polarizability and the interaction of unsaturated molecules with the mobile phase and therefore,

favors the elution process. Furthermore, GATS2P encodes the hydrophobicity of the compound, thus, an increase in this descriptor strengthens the hydrophobicity of the molecule, enhances the interaction between the solute and stationary phase, and then disfavors the elution process. Both these interactions can lead to a decrease in the value of  $t_R$  on the whole.

The last descriptor is the total number of primary carbons (NCP). This constitutional descriptor depends on the atomic constitution of the chemical structure (molecule). This descriptor is insensitive to any conformational change and does not distinguish among isomers. This constitutional descriptor encodes the size, shape, and degree of branching in the compound, also relates to the dispersion interaction among molecules. The larger the molecular size is, the stronger the dispersion interaction becomes. Thus, in some sense, it has some correlation with the hydrodynamic friction. So the larger the value of the NCP is, the longer the retention times of the molecule.

From the above discussion, it can be concluded that all descriptors involved in the QSPR model have some physical meaning, and that they account for structural features influencing the retention (times) of the molecules. We can conclude that the retention mechanism of RPLC mainly correlates with the factors as mentioned above, dispersive interactions, steric interaction between the solute and stationary phase and hydrodynamic friction among the peptides and the stationary phase and the mobile phase.

### ANN Modeling

The next step in our study was the generation, optimization and training of the ANN. Table 4 shows the architecture and specifications of the optimized ANN's parameters. After the optimization of the ANN's parameters, the network was trained using the training set for the adjustment of weights and bias values. It is known that neural networks can become over-trained. An over-trained network has generally learned absolutely the stimulus pattern it has seen but cannot give accurate prediction for unseen stimuli, and it is no longer capable to generalize, i.e. the network also has modeled the experimental error in the training set. There

**Table 4** Architecture and specifications of an optimized ANN model

Number of nodes in the input layer	4
Number of nodes in the hidden layer	6
Number of nodes in the output layer	1
Weights learning rate	0.2
Biases learning rate	0.3
Momentum	0.2
Transfer function	Sigmoid



are numerous methods for overcoming this problem. One method is to use a test set to assess the prediction power of the network throughout its training. In this method, after each 1,000 training iterations, the network was used to calculate  $t_R$  of molecules included in the test set. To preserve the predictive power of the network at an enviable level, training was stopped when the value of root mean squared error for the test set started to increase. Results obtained showed overtraining began after 42,000 iterations. Since the test set error is not a good estimation of the generalization error, the prediction potential of the model was evaluated on a third set of data, named validation set. Compounds in the validation set were not used during the training process and were reserved only to evaluate the predictive power of the generated ANN.

Table S3 lists the ANN estimated values of retention times of all seven CSs for the training, test and validation sets. The statistical parameters obtained by ANN model for these sets are summarized in Table 5. Comparison between

the statistical parameters in Table 5 reveals the superiority of the ANN model over PLS one; the  $R$  values systematically higher the errors smaller. The key strength of neural networks, unlike regression analysis, is their ability to flexible mapping of the selected features by manipulating their functional dependence implicitly.

### SVM Modeling

SVM is used to generate another non-linear model based on the same subset of descriptors. The performance of SVM for regression depends on the combination of several parameters: the capacity parameter  $C$ ,  $\varepsilon$  of the  $\varepsilon$ -insensitive loss function, and  $\gamma$  controlling the amplitude of the Gaussian function.  $C$  is a regularization parameter that controls the tradeoff between maximizing the margin and minimizing the training error. If  $C$  is too small, then inadequate strain will be placed on fitting the training data. If  $C$  is too large, then the algorithm will over-fit the training data. To

**Table 5** Statistical parameters obtained using the PLS, ANN and SVM models

CS	PLS			ANN			SVM		
	$R$	RMSE	$F$	$R$	RMSE	$F$	$R$	RMSE	$F$
CS1									
Training	0.953	1.26 <sup>a</sup>	525	0.984	0.67	1,580	0.992	0.46	3,467
Test	0.937	1.25 <sup>b</sup>	121	0.976	0.74	342	0.991	0.46	907
Validation	0.945	1.24 <sup>c</sup>	143	0.984	0.59	513	0.993	0.40	1,137
CS2									
Training	0.940	1.36	404	0.982	0.74	1,411	0.991	0.54	2,847
Test	0.937	1.41	122	0.974	0.85	311	0.990	0.52	862
Validation	0.953	1.22	167	0.987	0.59	666	0.993	0.44	1,247
CS3									
Training	0.934	2.32	364	0.985	0.71	1,707	0.995	0.40	5,432
Test	0.920	1.56	94	0.979	0.81	392	0.992	0.47	1,018
Validation	0.965	1.47	222	0.986	0.63	616	0.998	0.30	4,254
CS4									
Training	0.932	1.57	350	0.982	0.76	1,451	0.994	0.44	4,202
Test	0.922	1.52	97	0.975	0.88	332	0.994	0.50	1,320
Validation	0.946	1.21	146	0.985	0.66	582	0.995	0.36	1,762
CS5									
Training	0.944	0.99	435	0.983	0.57	1,557	0.994	0.32	4,746
Test	0.959	1.75	193	0.979	0.55	395	0.992	0.36	1,029
Validation	0.939	0.90	128	0.987	0.42	626	0.995	0.26	1,868
CS6									
Training	0.935	1.09	369	0.980	0.61	1,253	0.992	0.38	3,256
Test	0.931	1.06	111	0.977	0.66	359	0.992	0.47	1,115
Validation	0.934	1.02	116	0.985	0.50	552	0.994	0.33	1,523
CS7									
Training	0.939	1.10	392	0.983	0.60	1,488	0.993	0.38	3,964
Test	0.935	1.08	118	0.975	0.63	324	0.989	0.41	779
Validation	0.945	1.00	142	0.982	0.64	473	0.993	0.36	1,182

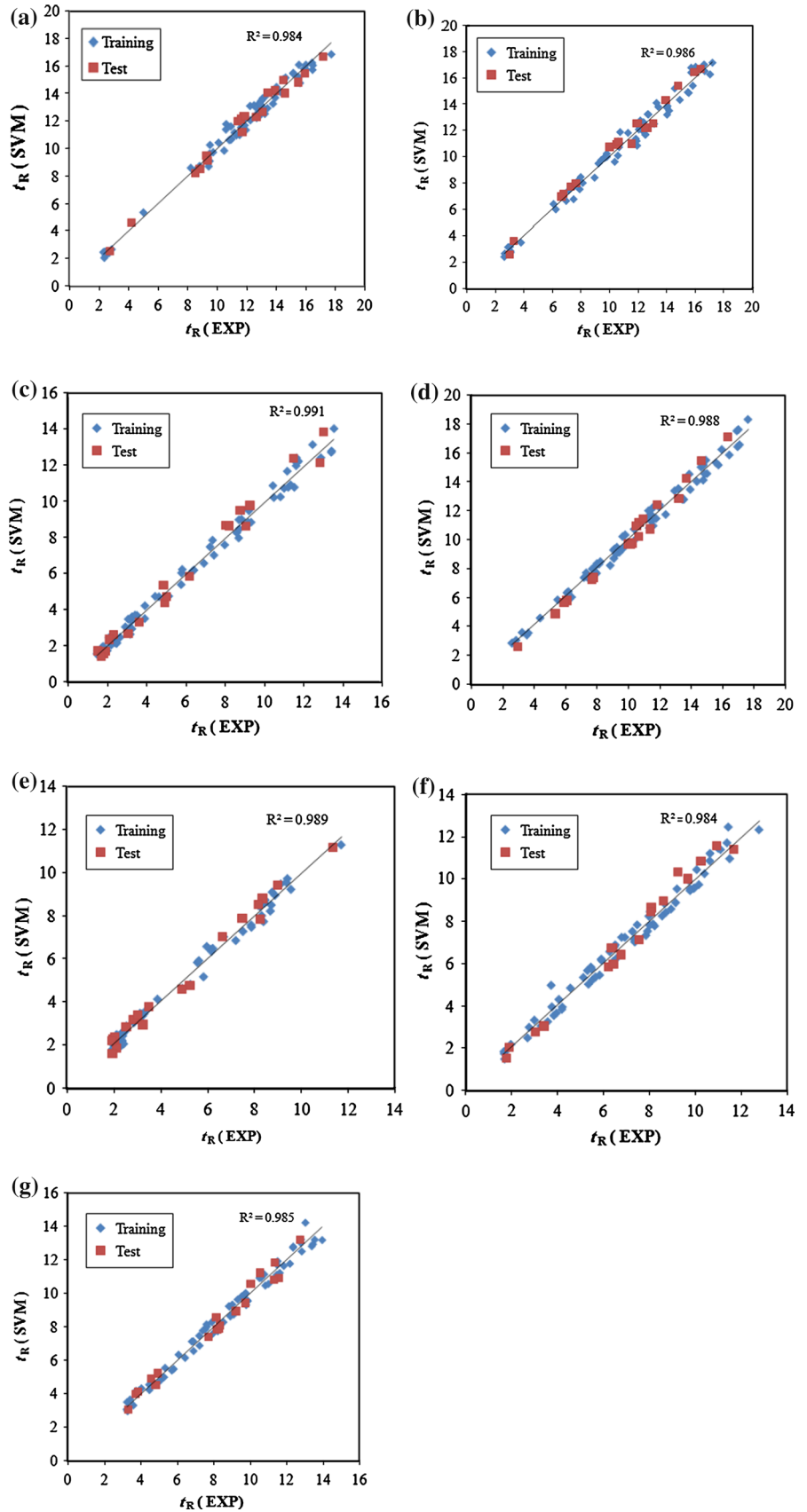
$R$  the correlation coefficient,  $RMSE$  root mean squared error

<sup>a</sup> Root mean squared error of calibration (RMSEC)

<sup>b</sup> Root mean squared error of prediction (RMSEP)

<sup>c</sup> Root mean squared error of cross validation (RMSECV)

**Fig. 2** Plot of SVM estimated versus experimental retention times for **a** CS1, **b** CS2, **c** CS3, **d** CS4, **e** CS5, **f** CS6 and **g** CS7



make the learning process steady, a large value should be set up for  $C$ . The kernel type is another significant factor. For regression errands, the Gaussian RBF kernel is generally utilized. The Gaussian RBF function is represented as follows:

$$\exp(-\gamma|u - v|^2)$$

where  $\gamma$  is a constant, parameter of the kernel,  $u$  and  $V$  are two independent variables.  $\gamma$  controls the amplitude of the Gaussian RBF function and consequently, controls the generalization ability of SVM. The best value for  $\varepsilon$  depends on the type of noise present in the data, which is usually unidentified. Even if sufficient knowledge of the noise is reachable to choose an optimal value for  $\varepsilon$ , there is the practical consideration of the number of resulting support vectors.  $\varepsilon$  insensitivity avoids the whole training set meeting border conditions and therefore, authorizes for the option of scattering in the dual formulation's solution. Thus, selecting the appropriate value of  $\varepsilon$  is mandatory. Consequently, these parameters should be optimized to acquire the best results. To select proper values for these parameters, diverse values were tried; the set of values with the best leave-one-out cross-validation performance was selected as the optimal. From the above process, the  $\gamma$ ,  $\varepsilon$  and  $C$  were fixed to 5, 0.04 and 300, respectively, when the support vector number was 45. The predicted results from the optimal SVM are shown in Table 5 for all seven CSs. The SVM model has higher correlation coefficient ( $R$ ) and Fisher values ( $F$ ) and lower RMSE for all three sets compared to the PLS and ANN models. The statistical parameters tabulated in Table 5 reveal the high accuracy and predictive ability of the model. Figure 2 shows the plot of the SVM predicted versus experimental values for the retention times of all molecules in the data set. (Divided over training, test and validation sets)

#### Comparison of the Results Obtained by Different QSPR Approaches

From the results of the QSPR models for modeling the retention time (Table 5), it can be seen that results obtained using SVM are comparable or superior to those by ANN and PLS. In fact, as a universal machine learning method, SVM is rooted in the structural risk minimization principle, which minimizes an upper bound of the generalization error rather than minimizing the training error. SVM thus has a better generalization performance than PLS and ANN. Moreover, compared to ANN, once corresponding parameters are specified, the solution of SVM is definite and reproducible, which is clearly better.

By performing model validation, it can be concluded that the presented model is a valid model and can be successfully employed to predict the  $t_R$  of peptides with an

**Table 6** Comparison of the obtained SVM model with MARS, TMARS, MLR, PLS and UVE-PLS

Method	System	RMSEC	RMSEP
SVM	CS1	0.46	0.40
	CS2	0.54	0.44
	CS3	0.40	0.30
	CS4	0.44	0.36
	CS5	0.32	0.26
	CS6	0.38	0.33
	CS7	0.38	0.36
MARS	CS1	1.05	0.82
	CS2	0.85	1.31
	CS3	1.34	1.43
	CS4	1.01	1.21
	CS5	1.54	1.11
	CS6	0.50	0.59
	CS7	1.59	1.17
TMARS	CS1	0.77	1.11
	CS2	0.57	0.70
	CS3	1.30	1.24
	CS4	1.00	0.95
	CS5	1.45	1.50
	CS6	0.57	0.75
	CS7	0.86	0.78
MLR	CS1	0.79	0.88
	CS2	0.93	0.90
	CS3	1.38	1.18
	CS4	1.28	1.16
	CS5	1.53	1.22
	CS6	0.80	0.71
	CS7	0.87	0.61
PLS	CS1	0.36	0.49
	CS2	0.54	0.57
	CS3	0.60	0.87
	CS4	0.57	0.75
	CS5	1.04	0.87
	CS6	0.48	0.47
	CS7	1.38	1.19
UVE-PLS	CS1	0.45	0.45
	CS2	0.55	0.48
	CS3	0.50	0.78
	CS4	0.55	0.62
	CS5	1.20	1.01
	CS6	0.35	0.40
	CS7	1.08	0.86

accuracy within the confidence limits from the experiential  $t_R$  determination. It can be logically accomplished that the proposed model will properly predict  $t_R$  for new peptides. In addition, the presented method could also recognize

and provide some insight into what structural features are related to the  $t_R$  of peptides.

### Comparison with Other QSPR Models

Put and Vander Heyden [11] developed a QSRR based on multivariate adaptive regression splines (MARS), two-step MARS (TMARS), PLS, uninformative variable elimination partial least squares (UV-PLS) and MLR for prediction of the retention times of the set of 98 peptides on the seven chromatographic systems. The comparison of statistics of each CS of our SVM model with other QSPR models is shown in Table 6. Comparison of the RMSEs of the present study with those from previous work shows the superiority of our SVM model.

Revelli, Mutelet and Jaubert [57] developed a linear solvation energy relationship (LSER) for predicting gas-to-ionic liquid partition coefficient ( $\log K_L$ ) and water to-ionic liquid partition coefficient ( $\log P$ ) of various organic compounds. The solute descriptors they used in their LSER models were: the excess molar refraction  $E$ ; the dipolarity/polarizability  $S$ ; the hydrogen bond acidity and basicity  $A$  and  $B$ , respectively, the gas–liquid partition coefficient on  $n$ -hexane at 298 K and McGown volume  $V$ . The squared correlation coefficient ( $R^2$ ) of the model for prediction set was 0.997 and 0.996, for  $\log K_L$  and  $\log P$ , respectively which are comparable with our results in Table 6.

### Conclusions

In the present work, applying the Support Vector Machine, QSPR models have been developed for predicting the  $t_R$  of a set of peptides from same of their molecular description values. The outcome of our computations indicates that while the GA-PLS method allows proper selection of important descriptors, the introduction of a SVM gives a substantial improvement in prediction quality. The calculated statistical parameters of these models expose the superiority of SVM over PLS and ANN models. The SVM reveals a better performance because it applies the structural risk minimization principle, which has been disclosed to be better than the conservative empirical risk minimization principle, employed by the usual machine learning techniques. SVM has the advantage over the other techniques of converging to the global optimum, and not to a local optimum that depends on the initialization and parameters affecting rate of convergence.

**Acknowledgments** T. Van Mulders for logistic support.

### References

1. Scriba GKE, Psurek A (2008) Separation of peptides by capillary electrophoresis. Capillary electrophoresis methods In Molecular biology 384: 483–506
2. Kasicka V (2006) Recent advances in capillary electrophoresis and capillary electrochromatography of peptides. Electrophoresis 27:142–175
3. Morisaka H, Kirino A, Kobayash K, Ueda M (2012) Two-dimensional protein separation by the HPLC system with a monolithic column. Biosci Biotechnol Biochem 76:585–588
4. Gorka J, Rohmer M, Bornemann S, Papatotiriou DG, Baeumlisberger D, Arrey TN, Bahr U, Karas M (2012) Perfusion reversed-phase high-performance liquid chromatography for protein separation from detergent-containing solutions: an alternative to gel-based approaches. Anal Biochem 424:97–107
5. Mant CT, Chen Y, Yan Z, Popa TV, Kovacs JM, Mills JB, Tripet BP, Hodges RS (2007) HPLC analysis and purification of peptides. Methods Mol Biol 386:3–55
6. Marchetti N, Guiochon G (2005) Separation of peptides from myoglobin enzymatic digests by RPLC. Influence of the mobile-phase composition and the pressure on the retention and separation. Anal chem 77:3425–3430
7. Gilar M, Jaworski A (2011) Retention behavior of peptides in hydrophilic-interaction chromatography. J Chromatogr A 1218:8890–8896
8. Baczek T, Kalisz R (2009) Predictions of peptides' retention times in reversed-phase liquid chromatography as a new supportive tool to improve protein identification in proteomics. Proteomics 9:835–847
9. Perlova TY, Goloborodko AA, Margolin Y, Pridatchenko ML, Tarasova IA, Gorshkov AV, Moskovets E, Ivanov AR, Gorshkov MV (2010) Retention time prediction using the model of liquid chromatography of biomacromolecules at critical conditions in LC-MS phosphopeptide analysis. Proteomics 19:3458–3468
10. Dimer F, Hubbuch J (2010) 3D structure-based protein retention prediction for ion-exchange chromatography. J Chromatogr A 1217:1343–1353
11. Put R, Vander Heyden Y (2007) The evaluation of two-step multivariate adaptive regression splines for chromatographic retention prediction of peptides. Proteomics 7:1664–1677
12. Puzyn T, Leszczynski J, Cronin MT (2010) Recent advances in qsar studies: methods and applications. Springer, Dordrecht
13. Liu HX, Xue CX, Zhang RS, Yao XJ, Liu MC, Hu ZD, Fan BT (2004) Quantitative prediction of  $\log k$  of peptides in high-performance liquid chromatography based on molecular descriptors by using the heuristic method and support vector machine. J Chem Inf Comput Sci 44:1979–1986
14. Petritis K, Kangas LJ, Ferguson PL, Anderson GA, Pasa-Tolic L, Lipton MS, Auberry KJ, Strittmatter EF, Shen YF, Zhao R, Smith RD (2003) Use of artificial neural networks for the accurate prediction of peptide liquid chromatography elution times in proteome analyses. Anal Chem 75:1039–1048
15. Ma W, Luan F, Zhang H, Zhang X, Liu M, Hu Z, Fan B (2006) Accurate quantitative structure–property relationship model of mobilities of peptides in capillary zone electrophoresis. Analyst 131:1254–1260
16. Shinoda K, Sugimoto M, Yachie N, Sugiyama N, Masuda T, Robert M, Soga T, Tomita M (2006) Prediction of liquid chromatographic retention times of peptides generated by protease digestion of the escherichia coli proteome using artificial neural networks. J Proteome Res 5:3312–3317
17. Du H, Wang J, Zhang X, Yao X, Hu Z (2008) Prediction of retention times of peptides in RPLC by using radial basis function

- neural networks and projection pursuit regression. *Chemom Intell Lab Sys* 92:92–99
18. Put R, Daszykowski M, Baczek T, Vander Heyden Y (2006) retention prediction of peptides based on uninformative variable elimination by partial least squares. *J Proteome Res* 5:1618–1625
  19. Vapnik VN (1997) *Statistical learning theory*. Wiley, New York
  20. Cortes C, Vapnik VN (1995) *Support vector networks*. *Mach Learn* 20:273–297
  21. Lima PC, Golbraikh A, Oloff S, Xiao Y, Tropsha A (2006) Combinatorial QSAR modeling of *P*-glycoprotein substrates. *J Chem Inf Model* 46:1245–1254
  22. Fatemi MH, Gharaghani S (2007) A novel QSAR model for prediction of apoptosis inducing activity of 4-aryl-4-H-chromenes based on support vector machine. *Bioorg Med Chem* 15:7746–7754
  23. Fatemi MH, Gharaghani S, Mohammadkhani S, Rezaie Z (2008) Prediction of selectivity coefficients of univalent anions for anion-selective electrode using support vector machine. *Electrochim Acta* 53:4276–4282
  24. Niazi A, Jameh-Bozorghi S, Nori-Shargh D (2008) Prediction of toxicity of nitrobenzenes using ab initio and least squares support vector machines. *J Hazard Mater* 151:603–609
  25. Pan Y, Jiang JC, Wang R, Cao HY (2008) Advantages of support vector machine in QSPR studies for predicting auto-ignition temperatures of organic compounds. *Chemom Intell Lab Syst* 92:169–178
  26. Pan Y, Jiang JC, Wang R, Cao HY, Zhao JB (2008) Quantitative structure–property relationship studies for predicting flash points of organic compounds using support vector machines. *QSAR Comb Sci* 27:1013–1019
  27. Baczek T, Wiczling P, Marszall M, Vander Heyden Y, Kaliszan R (2005) Prediction of peptides retention at different HPLC conditions from multiple linear regression models. *J Proteome Res* 4:555–563
  28. Hyperchem re 4 for Windows (1995) Autodesk, Sansalito, CA
  29. Mopac for Windows (2009) Stewart computational chemistry
  30. Todeschini R, Consonni V, Pavan M, Pisani V (2001) *Dragon software version 3.0*, Milano, Italy
  31. Zou X, Zhao J, Mao H, Shi J, Yin X, Li Y (2010) Genetic algorithm interval partial least squares regression combined successive projections algorithm for variable selection in near-infrared quantitative analysis of pigment in cucumber leaves. *Appl Spectrosc* 64:786–794
  32. Chen W, Dai P, Chen Y, Chen D, Jiang Z (2012) Feature selection method based on the adaptive genetic algorithm-kernel partial least squares for high dimensional data. *Adv Mat Res* 468:1762–1766
  33. Sraththaphut L, Phechkrajang C (2012) Genetic algorithms-based approach for wavelength selection in spectrophotometric determination of paracetamol and chlorzoxazone in tablet preparation by partial least-squares. *Indian J Pharm Edu Res* 46:62–68
  34. Leardi R, Gonzales AL (1998) Genetic algorithms applied to feature selection in PLS regression: how and when to use them. *Chemom Intell Lab Syst* 41:195–207
  35. Alma OG, Bulut E (2012) Genetic algorithm based variable selection for partial least squares regression using ICOMP criterion. *Asian J Math Stat* 5:82–92
  36. Samistraro G, Muniz GIB, Peralta-Zamora P, Cordeiro GA (2009) Estimation of physical properties of kraft paper by near infrared spectroscopy an partial least squares regression. *Quim Nova* 32:1422–1428
  37. Lorber A, Wangen L, Kowalsky BR (1987) A theoretical foundation for the PLS algorithm. *J Chemom* 1:19–31
  38. Witten JM, Park S, Myers KJ (2010) Partial least squares: a method to estimate efficient channels for the ideal observers. *IEEE Trans Med Imaging* 29:1050–1058
  39. Lê Cao KA, Boitard S, Besse P (2011) Sparse PLS discriminant analysis: biologically relevant feature selection and graphical displays for multiclass problems. *BMC Bioinform* 12:253–260
  40. Min H, Qi-bing Z (2011) Feature extraction of hyperspectral scattering image for apple mealiness based on singular value decomposition. *Spectrosc Spectr Anal* 31:767–770
  41. MATLAB 7.0, The mathworks, Natick, MA, USA. <http://www.mathworks.com>
  42. Jalali-Heravi M (2008) Neural networks in analytical chemistry. *Methods Mol Biol* 458:81–121
  43. Cartwright HM (2008) Artificial neural networks in biology and chemistry—the evolution of a new analytical tool. *Methods Mol Biol* 458:1–13
  44. Peterson KL (2007) In: Lipkowitz KB, Boyd DB (eds) *Artificial neural networks and their use in chemistry*. Wiley, Hoboken
  45. Byvatov E, Fechner U, Sadowski J, Schneider G (2003) Comparison of support vector machine and artificial neural network systems for drug/nondrug classification. *J Chem Inf Comput Sci* 43:1882–1889
  46. Liu HX, Zhang RS, Luan F, Yao XJ, Liu MC, Hu ZD, Fan BT (2003) Diagnosing breast cancer based on support vector machines. *J Chem Inf Comput Sci* 43:900–907
  47. Burbidge R, Trotter M, Buxton B, Holden S (2001) Drug design by machine learning: support vector machines for pharmaceutical data analysis. *Comput Chem* 26:5–14
  48. Darnag R, Mostapha Mazouz EL, Schmitzer A, Villemin D, Jarid A, Cherqaoui D (2010) Support vector machines: development of QSAR models for predicting anti-HIV-1 activity of TIBO derivatives. *Eur J Med Chem* 45:1590–1597
  49. Khosrokhavar R, Ghasemi JB, Shiri F (2010) 2D Quantitative structure–property relationship study of mycotoxins by multiple linear regression and support vector machine. *Int J Mol Sci* 11:3052–3068
  50. Burges CJC (1998) Tutorial on support vector machines for pattern recognition. *Data Min Knowl Discov* 2:1–47
  51. Vapnik V (1982) *Estimation of dependences based on empirical data*. Springer, Berlin
  52. Basak D, Pal S, Patranabis DC (2007) Support vector regression. *Neural Inf Process Lett Rev* 11:203–224
  53. Vapnik V (1995) *The nature of statistical learning theory*. Springer, New York
  54. Burden FRJ (1989) Molecular identification number for substructure searches. *Chem Inf Comput Sci* 29:225–227
  55. Sarkar RK, Roy AB, Sarkar PK (1978) Topological information content of genetic molecules. *Math Biosci* 39:299–312
  56. Geary RC (1954) The contiguity ratio and statistical mapping. *Incorp Statist* 5:115–145
  57. Revelli AL, Mutelet F, Jaubert JN (2010) Prediction of partition coefficient of organic compounds in ionic liquids: use of a linear solvation energy relationship with parameters calculated through a group contribution method. *Ind Eng Chem Res* 49:3883–3892