Neural Network Prediction of the Solvatochromic Polarity/Polarizability Parameter π_2^H

Daniel Svozil and Jiří G. K Ševčík*

Department of Analytical Chemistry, Faculty of Science, Charles University, Albertov 2030, Prague CZ-128 40, Czech Republic

Vladimír Kvasnička

Department of Mathematics, Faculty of Chemical Technology, Slovak Technical University, Bratislava SK-81 237, Slovakia

Received July 30, 1996[⊗]

A three-layer feed-forward neural network was used for the prediction of the polarity/polarizability parameter π_2^H. A simulated annealing algorithm was used to minimize the error at the neural network output. Descriptors related to the structure of the compounds were calculated as the input vector. The Kohonen neural network was used to split the data set into training and testing sets. The results obtained from the neural network were compared with the MLRA results.

INTRODUCTION

One of basic approaches to a quantitative description of solute−solvent interactions is related to the solvatochromic approach of Abraham et al.[1] It holds that

$$\log \mathrm{SP} = \mathrm{SP}_0 + rR_2 + s\pi_2^H + a\alpha_2^H + b\beta_2^H + l \log L^{16} \quad (1)$$

where the dependent variable SP can be any retention parameter, e.g., the adjusted retention time t_R', gas−liquid partition coefficient K, etc.

Regression coefficients SP_0, r, s, a, b, and l in eq 1 are characteristic for the given stationary phase. Coefficient r corresponds to the ability of the stationary phase to interact with π- and NB-electron pairs (NB means nonbonding) of the solute, coefficient s describes participation of the stationary phase in dipole−dipole interactions, and coefficient a or b represents the basicity or acidity of the stationary phase, respectively. Coefficient l describes the dispersion and cavitation forces. The variables on the right-hand side of eq 1 are related to the solutes: R_2, excess molar refraction; α_2^H and β_2^H, effective acidity and basicity parameters, respectively; α_2^H, polarity/polarizability; log L^{16}, includes both dispersion interactions and the cavitation term. The regression coefficients in eq 1 can be found by using multiple linear regression analysis. The applicability of this equation depends on the availability of solute parameters.

It has been found that some parameters (e.g., R_2, log L^{16}) are fully additive and predictable for all common compounds,[1−3] but some other parameters (e.g., π_2^H) are not additive.[4] It has further been found that the additivity principle for the polarity/polarizability parameter π_2^H is only valid for very limited sets of compounds; e.g., satisfactory results have been obtained for aliphatic compounds.[5]

The most common way of obtaining solute parameters for an unknown compound is based on calculation from chromatographic data using the back-regression method. This method assumes that the retention and solute parameters are known for a wide set of compounds. The method is very hard and time consuming. We decided to use a neural network (NN) for prediction of the π_2^H parameter. The neural network method can provide a nonlinear relationship between the input parameters (descriptors) by "learning itself" and is sufficiently robust.

THEORY

The structural formula of an organic compound contains all the information concerning its physicochemical properties. It is possible to calculate these properties from one or more descriptive parameters (descriptors) related to the molecular structure. The main aim of QSRR (quantitative structure−retention relationship) analysis, a part of QSPR (quantitative structure−property relationship), is to find an appropriate mathematical relationship between a molecular structure and its analytical properties. Since the pioneering work of Hansch in 1964,[6] QSPR analyses have been carried out by using various mathematical methods, that of feed-forward neural networks recently attracting considerable attention.

Feed-Forward Neural Networks. Multilayer feed-forward (MLF) neural networks, trained with a back-propagation learning algorithm, are the most popular neural networks. They are applied to a wide variety of chemically related problems.[7] A neural network architecture with three layers, the input layer containing 17 neurons, the hidden one seven neurons, and the output one a single neuron, was used in this study (Figure 1). Each neuron in a particular layer was connected with all the neurons in the next layer. The connection between the ith and jth neurons was characterized by the weight coefficient ω_{ij} and the ith neuron by the threshold coefficient ϑ_i (Figure 1). The weight coefficient reflects the degree of importance of the given connection in the neural network. The output value (activity) of the ith neuron x_i is determined by eqs 2 and 3. It holds that

$$x_i = f(\xi_i) \quad (2)$$

$$\xi_i = \vartheta_i + \sum_j \omega_{ij} x_j \quad (3)$$

where ξ_i is the potential of the ith neuron and the function $f(\xi_i)$ is the so-called transfer function (the summation in eq

[⊗] Abstract published in *Advance ACS Abstracts,* February 15, 1997.

S0095-2338(96)00347-2 CCC: $14.00

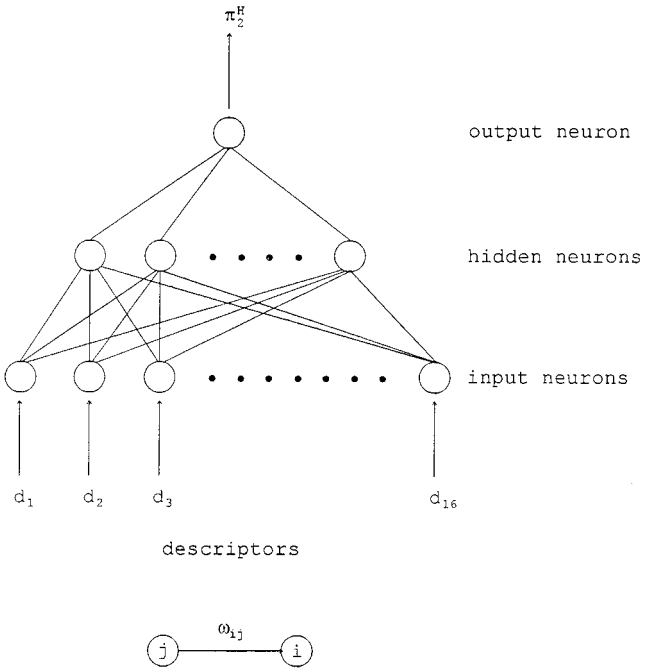

Figure 1. Architecture of the feed-forward neural network used.

3 is carried out over all neurons *j* transferring the signal to the *i*th neuron).

For the transfer function it holds that

$$f(\xi) = \frac{1}{1 + \exp(-\xi)} \tag{4}$$

The supervised adaptation process varies the threshold coefficients ϑ_i and weight coefficients ω_{ij} to minimize the sum of the squared differences between the computed and measured value. This is accomplished by minimization of the objective function *E*:

$$E = \frac{1}{2}(x_o - \widehat{x}_o)^2 \tag{5}$$

where x_o and $\widehat{x}_o$ are the computed and required activities of the output neuron.

For adjustment of the weight and threshold coefficients in the steepest-descent minimization it holds that

$$\omega_{ji}^{(k+1)} = \omega_{ji}^{(k)} - \lambda\left(\frac{\partial E}{\partial \omega_{ji}}\right) + \alpha\Delta\omega_{ji}^{(k)} \tag{6}$$

$$\vartheta_j^{(k+1)} = \vartheta_j^{(k)} - \lambda\left(\frac{\partial E}{\partial \vartheta_j}\right) + \alpha\Delta\vartheta_j^{(k)}$$

where λ is the rate of learning ($\lambda = \langle 0,1 \rangle$) and α is the so-called momentum factor ($\alpha = \langle 0,1 \rangle$) accelerating the steepest-descent method. The back-propagation method calculates the derivative of the objective function *E* and optimizes it for the minimum (the error between the computed and required activities is minimal). Unfortunately, the steepest-gradient method does not ensure that the adjusted weight and threshold coefficients correspond to the global minimum. The method of simulated annealing, based on the Metropolis algorithm, was thus used in our neural network.

EXPERIMENTAL SECTION

Data Set. A published data set of polarity/polarizability parameters π_2^H for 333 organic compounds[8] was used in our study. Aliphatic compounds and their derivatives were omitted during preliminary calculations. The original data set was further reduced by omitting all compounds with $\pi_2^H \geq 1$. The data set was split into the training and testing sets. The training set was used for adaptation of the neural network, whereas the testing set was used for verification of the adapted neural network.

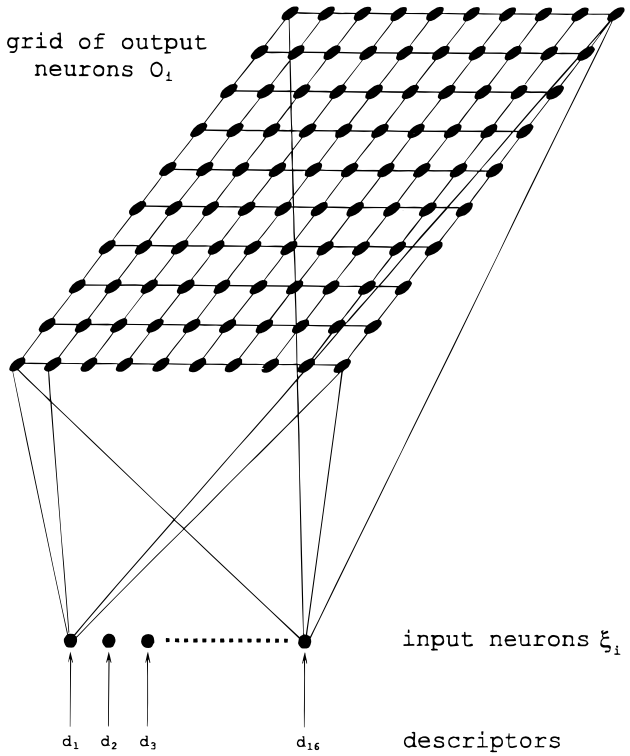

Figure 2. Kohonen neural network architecture.

The data splitting was carried out by using the Kohonen neural network (Figure 2). The architecture of this type of neural network is different from that of the feed-forward neural network described above. The Kohonen neural network is based on two layers of neurons. The first layer acts only as the input layer, through which the input signal is led into the second layer. Each neuron from the input layer (activity ξ_i) is connected with each neuron (O_i) in the output layer arranged as a two-dimensional grid (Figure 2). In the Kohonen algorithm, the input vector has *N* components $\xi_1 - \xi_N$, which define the point $\vec{\xi}$ in the *N*-dimensional input space. In our case this input vector consisted of the molecular descriptors for the given compound and the value of π_2^H. The output neurons *i* (with output values O_i) were arranged into a 10 × 10 square grid. Each input neuron was connected with all the output neurons and a weight coefficient of $\omega_{ij} \geq 0$ was assigned to each connection. A single output neuron was activated at a given moment (i.e., the output from this neuron was equal to 1, the neuron fired; the other neurons had zero outputs and were inactive). The firing output neuron is called winning neuron (*i**) and has the output activity $O_{i^*} = 1$. The winning neuron is the neuron with the highest potential h_i:

$$h_i = \sum_j \omega_{ij}\xi_j \tag{7}$$

The weight coefficients were corrected using the Kohonen algorithm that is described in detail in Hertz et al.'s textbook.[9]

Table 1. List of the Descriptors Used[a]

descriptors	citation
Constitutional Descriptors	
molecular weight, MW	
Electrostatic Descriptors	
sum of positive charges on the backbone, Q	
Quantum−Chemical Descriptors	
HOMO	
LUMO	
dipole moment, μ	
Topological Descriptors	
Wiener index W	14
polarity index P	15
Zagreb index of the 1st order M_1	16
Randiæ's indices of the 1st−5th orders $^{1-5}\chi_R$	17
Kier-Hall's indices of the 0th−5th orders $0^{-5}\chi_K$	18
Kappa indices of the 0th−3rd orders $^{0-3}\kappa$	19
Platt's index F	20
total topological index TTI	21
Shannon's (Bertz's) index SI	22
information content IC	22
redundancy index RI	21
Bonchev's normalized index BI	21

[a] Splitting into the formal groups was done according to Katritzky et al.[11]

The Kohonen training algorithm is an example of the unsupervized training algorithm (it does not require a "teacher", such as a back-propagation algorithm). Our Kohonen neural network was used with $\alpha = 0.2$ (learning constant), $d_0 = 10$ (initial size of neighborhood), and $T = 10\,000$ (number of learning steps). We employed the rectangular type of neighborhood, and the output activities were determined as L_1 (city blocks, Manhattan) distances between the input activities and the corresponding weights. After finishing the adaptation process, all objects were clustered so that each object activated only one output neuron on the rectangular grid, and some output neurons were never activated and/or were activated by one or more objects. The splitting of the objects through the grid of output neurons may thus be considered as clustering of the objects, each cluster being composed of one or more objects and specified by a single output neuron. Finally, the training set was created by shifting one object from each cluster to the training set and the remaining one to the testing set.

Descriptors. In this work, 29 descriptors were calculated. They can be grouped into the categories of topological, electrostatic, quantum−chemical, and constitutional descriptors.[10,11] The list of the descriptors used and the corresponding literature references are given in Table 1. The whole data set can be obtained on request.

CALCULATIONS

The graphical program ISIS Draw was used to sketch the structure of the compounds used in that data set. The structures were stored in the form of connection tables representing atoms and bond types and were used for the following calculations.

The topological indices (Table 1) were calculated using our own programs written in Borland C++ 3.1 and the Kier−Hall's program Molconn-X 2.0. Electrostatic, constitutional, and quantum−chemical descriptors (Table 1) were calculated with the program Mopac 7.0 which uses a semiempirical molecular orbital method (AM1) to compute the minimum energy conformation for each structure in the data set. Multilinear regression analysis (MLRA) and statistical analysis were performed using the program S-Plus for Windows 3.2. All neural network computations used our programs written in Borland C++ 3.1 and Turbo Pascal 7.0. For calculations of polarity/polarizability parameters π_2^H, the feed-forward neural network was used with the simulated annealing algorithm in the combination with the back-propagation training algorithm.

The weight and threshold coefficients were precalculated using the simulated annealing algorithm (trying to reach the best local minimum), and the process of training was then finished with the help of the back-propagation algorithm. The architecture of the neural network was (16,7,1), i.e., 16 input neurons, seven hidden neurons, and one output neuron. Seven hidden neurons were used because the results of MLRA (Table 2) indicated a very strong nonlinear relationship between the descriptors and the predicted variable π_2^H.

For the feed-forward neural network, the parameter α (momentum factor) was set to 0.5 and λ (learning rate) to a small positive number (between 0.01 and 0.05). The learning rate was optimized during the calculation (when the objective function in a particular iteration was larger than that in the previous iteration, the λ value was decreased by 9%). The initial values of the weight and threshold coefficients for the first iteration were randomly generated within the interval $\langle -1,1 \rangle$.

The quality of a prediction was evaluated by the coefficient of determination:

$$r^2 = 1 - \frac{\sum (x_{obs} - x_{calc})^2}{\sum (x_{obs} - \bar{x}_{calc})^2} \tag{8}$$

where $\bar{x}_{obs}$ is the mean value of the observed variable. For prediction of the polarity/polarizability parameter π_2^H, the method of multilinear regression analysis was also used and the results were compared with the NN calculations results. The value of π_2^H was determined in the MLRA method as a linear combination of all the 16 descriptors (d_i) plus a constant term (c_0):

$$\pi_2^H = c_0 + \sum_{i=1}^{16} c_i d_i \tag{9}$$

The MLRA calculations were performed with the same training and testing sets (obtained from the Kohonen neural network) as those used in the neural network calculations (Table 2).

RESULTS AND DISCUSSION

All the results are summarized in Table 2 and discussed below. The predictability of MLRA and NN was tested in parallel.

First, the data set was split into the training and testing sets. Then the MLRA analysis was performed with all 29 computed c_i, yielding $r^2_{train} = 0.940$, $r^2_{test} = 0.585$. The set of 29 descriptors was then reduced, excluding descriptors with variances 10 times larger than the value of the regression coefficient (Table 3).

Table 2. Results Obtained from the MLRA and NN Methods for Different Data Sets

				MLRA		NN		
compounds	data set	training set	testing set	r^2_{train}	r^2_{test}	r^2_{train}	r^2_{test}	pool of descriptors
A + B + Ph + Py	all	67	266	0.940	0.585			all
A + B + Ph + Py	all	69	264	0.741	0.450	0.908	0.537	1
B + Ph + Py	with outliers	66	182	0.858	0.588	0.941	0.784	2
B + Ph + Py	without outliers	66	170	0.898	0.643	0.928	0.836	2
B + Ph	with outliers	62	97	0.959	0.632	0.944	0.837	2
B + Ph	without outliers	62	90	0.959	0.778	0.979	0.932	2

Table 3. Pools of Descriptors Obtained via Two Different Statistical Methods

pool of descriptors no.	selection method	descriptors used
1	decriptors excluded on the basis of the MLRA results	W, M_1, $^0\kappa$, SI, MW, HOMO, μ
2	descriptors excluded on the basis of the cross-correlation matrix	W, P, M_1, $^5\chi_R$, $^2\kappa$, $^3\kappa$, F, SI, IC, RI, BI, MW, HOMO, LUMO, Q, μ

The new calculations were performed using the MLRA, resulting in $r^2_{train} = 0.741$, $r^2_{test} = 0.450$, and the NN method, resulting in $r^2_{train} = 0.980$, $r^2_{test} = 0.537$. The worse fit obtained for the MLRA indicates that the relationship between the input descriptors and the predicted variable π_2^H is strongly nonlinear. A neural network with seven hidden neurons was thus chosen, as such a neural network is capable of covering nonlinear relationships (in general, the higher the number of hidden neurons, the better the neural network ability to cover nonlinearities). However, neural networks with many hidden neurons exhibit a worse predictive ability. Even if the NN results were better then the MLRA results, they were still insufficient. Therefore, all the aliphatic hydrocarbons and their derivatives were omitted from the data set. To reduce the number of the descriptors used for the neural network and the MLRA calculations, the pairwise correlation (a cross-correlation matrix) was used. In the case of a high correlation ($r > 0.90$) between two descriptors, one of them was removed from the descriptor pool, even if it had been shown[12] that correlation results were better when using more intercorrelated descriptors then just a single, best one. In that way, a pool consisting of 16 descriptors (Table 3) was obtained. Then the MLRA and NN analysis calculations with the data sets were carried out comprising benzene, phenol, and pyridine derivatives and benzene and phenol derivatives (Table 2).

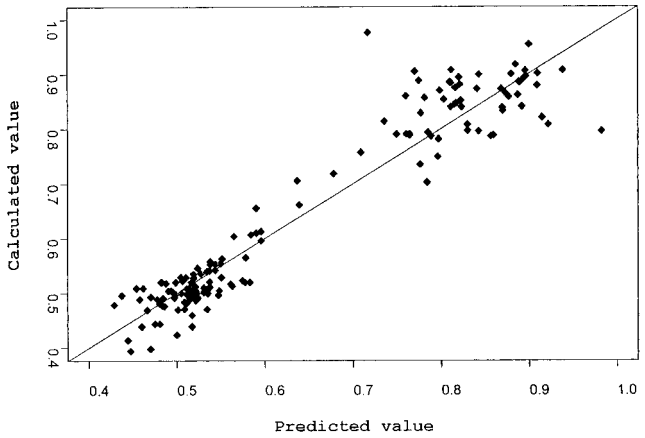

Figure 3. MLRA correlation plot of π_2^H. Predicted values are taken from ref 8; for calculated values see Table 4 (benzene + phenol, all objects, without outliers): $R^2 = 0.9099$; residual standard error, 0.054 58 on 136 degrees of freedom; F-statistic = 85.89 on 16 and 136 degrees of freedom.

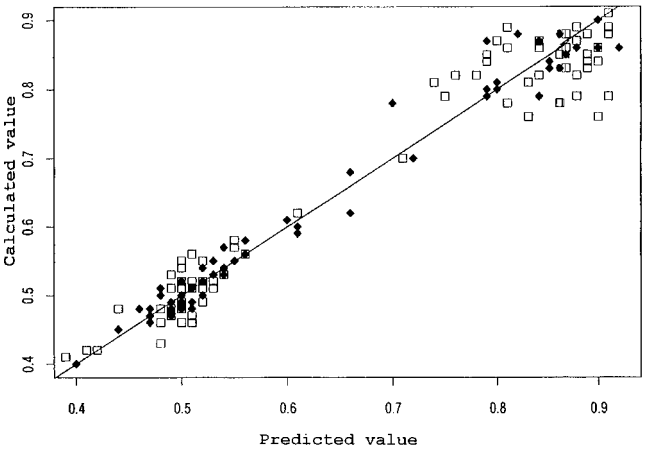

Figure 4. NN correlation plot of π_2^H. Predicted values are taken from ref 8; calculated values are received from NN computation ((◆) benzene + phenol, training set (62 objects); (□) testing set (90 objects); see Table 2).

CONCLUSIONS

The comparison of the results obtained from the MLRA (Figure 3) and NN (Figure 4) methods shows that the NN method tends to yield better predictions than the MLRA method. It is assumed that feed-forward neural networks cover nonlinear relationships between the dependent variables (calculated descriptors c_i) and independent variables (predicted value of π_2^H).

Table 4. MLRA Regression Coefficients and Their Standard Errors[a]

descriptor	W	P	M_1	$^5\chi_R$	$^2\kappa$	$^3\kappa$	F	SI	IC
reg coef	−0.161	0.136	−0.091	−0.045	−0.069	0.152	−0.113	−3.321	0.219
std error	0.053	0.040	0.048	0.047	0.047	0.047	0.064	1.242	0.077

descriptor	RI	BI	MW	HOMO	LUMO	Q	μ	intercept
reg coef	−2.225	−0.027	0.201	−0.015	0.196	0.761	0.236	2.919
std error	0.861	0.059	0.113	0.016	0.101	0.150	0.018	0.832

[a] Benzene + phenol, all objects, without outliers; $R^2 = 0.9099$; residual standard error, 0.054 58 on 136 degrees of freedom; F-statistic = 85.89 on 16 and 136 degrees of freedom.

The neural networks are very robust (i.e., very resistant against the input errors), and because of their "learning ability", they are capable of covering more relations between the data than the classical regression analysis. On the other hand, the training of neural networks is a time-consuming procedure, especially for complex systems. The processes taking place during the training of a network are not well interpretable, and this area is still under development.[13]

To construct a universal neural network that would make it possible to solve a given chemical problem for the widest range of the compounds seems to be impossible because of their large variability. We assume that for a general prediction of the parameter π_2^H neural networks trained for some subsets of compounds must be constructed and then selectively used. This problem is subject to further investigations.

ACKNOWLEDGMENT

This work was supported by the Grant Agency of the Czech Republic (Grant No. 203/95/0805).

REFERENCES AND NOTES

(1) Abraham, M. H.; Whiting, G. S.; Doherty, R. M.; Shuely, W. J. Hydrogen bonding. Part 13. A new Method for the characterization of GLC Stationary Phases - The Laffort Data Set. *J. Chem. Soc., Perkin Trans. 2* **1990**, 1451−1460.

(2) Havelec, P.; Ševčík, J. G. K. Concept of additivity for a non-polar solute−solvent criterion log(L^{16}). Non-aromatic compounds. *J. Chromatogr. A* **1994**, *677*, 319−329.

(3) Havelec, P.; Ševčík, J. G. K. Extended Additivity Model of Parameter log(L^{16}). *J. Phys. Chem. Ref. Data* **1996**, *25*, 1483−1493.

(4) Abraham, M. H.; Whiting, G. S. Hydrogen bonding. XXI. Solvation parameters for alkylaromatic hydrocarbons from gas-liquid chromatographic data. *J. Chromatogr.* **1992**, *594*, 229−241.

(5) Abraham, M. H.; Whiting, G. S.; Doherty, R. M.; Shuely, W. J. Hydrogen bonding. XVI. A new solute solvation parameter, π_2^H, from gas chromatographic data. *J. Chromatogr.* **1991**, *587*, 213−228.

(6) Hansch, C.; Fujita, T. ρ-σ-π Analysis. A Method for the Correlation of Biological Activity and Chemical Structure. *J. Am. Chem. Soc.* **1964**, *86*, 1616−1626.

(7) Zupan, J.; Gasteiger, J. *Neural Networks for Chemists*; VCH Publishers: New York, 1993.

(8) Abraham, M. H. Hydrogen bonding. XXVII. Solvation parameters for functionally substituted aromatic compounds and heterocyclic compounds, from gas-liquid chromatographic data. *J. Chromatogr.* **1993**, *644*, 95−139.

(9) Hertz, J.; Krogh, A.; Palmer, G. R. *Introduction to the Theory of Neural Computation*; Addison-Wesley Publishing Co.: Reading, MA, 1991.

(10) Gordeeva, E. V.; Katritzky, A. R.; Shcherbukhin, V. V.; Zefirov, N. S. Rapid Conversion of Molecular Graphs to Three-Dimensional Representation Using the MOLGEO Program. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 102−111.

(11) Katritzky, A. R.; Lobanov, V. S.; Karelson, M. QSPR: The Correlation and Quantitative Prediction of Chemical and Physical Properties from Structure. *Chem. Soc. Rev.* **1995**, *24*, 279−287.

(12) Randic, M. Resolution of Ambiquities in Structure-Property Studies by Use of Orthogonal Descriptors. *J. Chem. Inf. Comput. Sci.* **1991**, *31*, 311−320.

(13) Andrews, R.; Diedrich, J.; Tickle, A. B. *A Survey and Critiques For Extracting Rules from Trained Artificial Neural Networks*; internal printing of the Neurocomputing Research Centre, Queensland University of Technology, Brisbane, Australia.

(14) Wiener, H. Structural Determination of Paraffin Boiling Points. *J. Am. Chem. Soc.* **1947**, *69*, 17−20.

(15) Wiener, H. Correlation of Heats of Isomerization, and Differences in Heats of Vaporization of Isomers, Among the Paraffin Hydrocarbons. *J. Am. Chem. Soc.* **1947**, *69*, 2636−2638.

(16) Gutman, I.; Trinajstič, N. Graph Theory and Molecular Orbitals. Total π-Electron Energy of Alternant Hydrocarbons. *Chem. Phys. Lett.* **1972**, *17*, 535−538.

(17) Randič, M. On Characterization of Molecular Branching. *J. Am. Chem. Soc.* **1975**, *97*, 6609−6615.

(18) Kier, L. B.; Hall, L. H. Molecular connectivity VII: Specific Treatment of Heteroatoms. *J. Pharm. Sci.* **1976**, *65*, 1806−1809.

(19) Kier, L. B. A shape index from molecular graphs. *Quant. Struct.-Act. Relat.* **1985**, *4*, 109−112.

(20) Platt, J. R. Prediction of Isomeric Differences in Paraffin Properties. *J. Phys. Chem.* **1952**, *56*, 328−336.

(21) Kier, L. B.; Hall, L. H. *User's Guide for Molconn-X 2.0*, Hall Associates Consulting, Quincy, MA.

(22) Todeschini, R.; Cazar, R.; Collina, E. The chemical meaning of topological indices. *Chemom. Intel. Lab. Syst.* **1992**, *15*, 51−59.

CI960347E