k Nearest Neighbors QSAR Modeling as a Variational Problem: Theory and Applications

Peter Itskowitz and Alexander Tropsha*

Laboratory for Molecular Modeling, School of Pharmacy, University of North Carolina at Chapel Hill, North Carolina 27599-7360

Received December 9, 2004

Variable selection *k* Nearest Neighbor (kNN) QSAR is a popular nonlinear methodology for building correlation models between chemical descriptors of compounds and biological activities. The models are built by finding a subspace of the original descriptor space where activity of each compound in the data set is most accurately predicted as the averaged activity of its *k* nearest neighbors in this subspace. We have formulated the problem of searching for the optimized kNN QSAR models with the highest predictive power as a variational problem. We have investigated the relative contribution of several model parameters such as the selection of variables, the number (*k*) of nearest neighbors, and the shape of the weighting function used to evaluate the contributions of *k* nearest neighbor compound activities to the predicted activity of each compound. We have derived the expression for the weighting function which maximizes the model performance. This optimization methodology was applied to several experimental data sets divided into the training and test sets. We report a significant improvement of both the leave-one-out cross-validated R^2 (q^2) for the training sets and predictive R^2 of the test sets in all cases. Depending on the data set, the average improvements in the prediction accuracy (prediction R^2) for the test sets ranged between 1.1% and 94% and for the training sets (q^2) between 3.5% and 118%. We also describe a modified computational procedure for model building based on the use of relational databases to store descriptors and calculate compounds' similarities, which simplifies calculations and increases their efficiency.

INTRODUCTION

Recent trends in Quantitative Structure−Activity Relationship (QSAR) studies have focused on the development of optimal models through variable selection. This procedure selects only a subset of available chemical descriptors—those that are most meaningful and statistically significant in terms of correlation with biological activity. The optimum selection of variables is achieved by combining stochastic search methods with either linear or nonlinear correlation methods such as multiple linear regression (MLR), partial least squares (PLS) analysis, or artificial neural networks (ANN).[1−6] Most of the time, these methods employ generalized simulated annealing,[2] genetic algorithms,[3] or evolutionary algorithms[4−7] as the stochastic optimization tool. Several fitting functions have been applied to improve the effectiveness and convergence of these algorithms.[4,5] It has been demonstrated that variable selection effectively improves QSAR models (reviewed in ref 8).

Several nonlinear QSAR methods have been proposed in recent years. Most of these methods are based on either ANN[9−16] (both back-propagation (BP-ANN) and counterpropagation (CP-ANN)[17] approaches) or machine learning techniques.[18−21] Since model building involves the optimization of many parameters, the speed of the analysis is relatively slow although some progress in improving the computational efficiency of nonlinear QSAR modeling was reported. Thus, Hirst developed a simple and fast nonlinear QSAR method,[22] in which the activity surface was generated from the activities of training set compounds based on some predefined mathematical function.

k Nearest Neighbor (kNN) QSAR is an example of a nonlinear approach that our group has been actively developing and applying to many data sets for several years.[23−27] It is based on the active analogue approach to the analysis of structure−activity relationships, which implies that structurally similar compounds are expected to have similar biological activities. Consequently, the activity of a compound can be estimated as an average of the activities of its *k* ($k = 1$, 2, etc...) nearest neighbors in the descriptor space. Further, we take into account that the perception of structural similarity is relative and should always be considered in the context of a particular biological target. Since the physicochemical characteristics of the receptor binding site vary from one target to another, the structural features that can best explain the observed biological similarities between compounds are different for different properties of interest. The optimal choice of relevant structural descriptors is achieved by the "bioactivity driven" variable selection, i.e., by searching for a subset of molecular descriptors that afford a highly predictive kNN-QSAR model. Since the number of all possible combinations of descriptors is huge, an exhaustive analysis of all combinations is not possible. Thus, a stochastic optimization algorithm, e.g., simulated annealing (SA), is employed for the efficient sampling of the multidimensional descriptor space.[23]

In this paper, we formulate the process of kNN QSAR model development as a variational problem. This affords a systematic way to identify factors that influence the perfor-

* Corresponding author e-mail: tropsha@email.unc.edu.

10.1021/ci049628+ CCC: $30.25
Published on Web 04/15/2005

mance of the model in terms of its internal accuracy, i.e., the leave-one-out (LOO) R^2 (q^2) for the training set. Concurrently, we improve the computational efficiency of the model development using relational databases to store chemical descriptors and calculate compound similarities. This modified method is significantly faster and much simpler as compared to present approaches. We show that the application of the modified kNN QSAR approach to several experimental data sets afforded robust models with improved both internal and external prediction accuracy as compared to models generated with the previous implementation of this approach.

THEORY

Background: A Brief Overview of the kNN QSAR Approach. The kNN QSAR method employs the kNN classification principle[28] and a variable selection procedure. Briefly, a subset of *nvar* (number of selected variables) descriptors is selected randomly in the beginning. The *nvar* is set to different values to obtain the best q^2 possible, which may be dependent not only on the choice but also on the number of descriptors as well. The selection is optimized by LOO cross-validation, where each compound is eliminated from the training set and its activity is predicted as the average activity of k most similar molecules ($k = 1$ to 5 in the original implementation[23]). The similarity is characterized by the Euclidean distance between compounds in multidimensional space of normalized (range-scaled) descriptors. A method of simulated annealing with the Metropolis-like acceptance criteria is used to optimize the variable selection. Further details of the kNN method implementation including the description of the simulated annealing procedure used for stochastic sampling of the descriptor space are given elsewhere.[23]

The original kNN method[23] was enhanced recently by using weighted molecular similarity.[24] In the original method, the activity of each compound was predicted as the algebraic average activity of its k nearest neighbor compounds in the training set. However, in general, the Euclidean distances in the descriptor space between a compound and each of its k nearest neighbors are not the same. Thus, the neighbor with the smaller distance from a compound was given a higher weight in calculating the predicted activity as $y^{pred} = \sum_{k\ nearest\ neighbors} w_i y_i$, where $w_i = \rho(d_i)$ is the weight for every individual nearest neighbor; $\rho(d_i)$ is some normalized function of the Euclidean distance d_i between the compound and its k nearest neighbors; y_i is the actual activity value for nearest neighbor i; and y^{pred} in is the predicted activity value.

Another recent area of improvement with respect to the computational efficiency of the method was the introduction of the chemical similarity cutoff (this option is available in our current implementation of kNN QSAR). Variable selection and model validation process are very time-consuming, because one needs to calculate the similarity between all pairs of compounds based on each trial subset of descriptors for every cycle of model development. In our current implementation, for each trial subset variables, we have to calculate at least $N \times (N\text{-}1)/2$ distances for a training set that contains N compounds. Indeed, for each compound, its k nearest neighbors should be identified from the remaining N-1 compounds. Furthermore, before we obtain the optimal subset of variables, the SA-driven variable selection procedure has to be repeated many times. When N is relatively large, the calculations take a very long time because the algorithm scales as N^2. To alleviate this problem, we introduced a restricted similarity cutoff approximation. We made the assumption that k nearest neighbors of each compound in any subspace of the original descriptor space (i.e., any space of selected variables) are always included in a large subset (but much smaller than the total number of compounds) of nearest neighbors of this compound in the original descriptor space. This implies that prior to the beginning of kNN optimization, we introduce a restricted set of neighbors for each compound based on some similarity threshold in the original descriptor space and continue to select k nearest neighbors of each compound from this restricted set of neighbors (N_{rs}). Hence, in the SA-driven variable selection process, we just calculate the distances between each compound and its restricted set of neighbors in order to find out its k nearest neighbors, instead of considering all remaining N-1 compounds. Because we identify the restricted neighbors with $N_{rs} \ll N$, the calculation time to search for k nearest neighbors of each compound is significantly reduced.

In summary, the kNN-QSAR algorithm generates both an optimum k value and an optimal *nvar* subset of descriptors, which afford a QSAR model with the highest value of q^2 for the training set. Although this approach has been applied successfully to many experimental data sets and the modifications of the original method described above have improved its computational efficiency and accuracy, we sought more drastic and less artificial improvements of the method than using the restricted similarity cutoff. We now introduce several concepts pertaining to revisiting the kNN model development process and assessing the kNN model optimization as the variational optimization problem.

kNN QSAR Modeling Process. We first discuss the chemical descriptor space in order to introduce several necessary definitions. Assume that we have K descriptors for each compound and that these descriptors are normalized to have similar scales of values (we typically use range scaling as discussed in ref 23), linearly independent and orthogonal. (If the descriptors are not orthogonal, principal component analysis (PCA)[29] could be applied first.) Next, assume that the K descriptors form a complete set in a sense that any property of the system, e.g., physical property or biological activity, can be expressed as some function of the descriptor values. We shall denote this complete space of all K descriptors as σ_F. Thus, the target property (activity) can then be written as

$$y = A(\vec{x}) \tag{1}$$

where y is the activity value and $\vec{x} = (x_1, x_2, ..., x_K)$ represents a vector in the descriptor space. Note, that the function $A(\vec{x})$ is not necessarily dependent upon all descriptors in σ_F. Indeed, the complete set may include descriptors that are formally different for different compounds yet do not improve or even may add noise to the correlation with the target property. Formally, for such descriptors

$$\frac{\partial A}{\partial x_i} = 0 \tag{2}$$

for all values of x_l. We will denote the subspace formed by all descriptors that contribute to $A(\vec{x})$ (or, in other words, for which condition (2) is not satisfied) as σ_A. Evidently, σ_A is a subspace of the complete space σ_F. Finally, we shall define the model subspace σ_M, which corresponds to *nvar* described above in the context of the kNN QSAR modeling approach. σ_M is a set of descriptors used to build a particular model and it is also a subspace of σ_F.

Suppose, that we have a set of data points *i*, for which $y_i = A(\vec{x})$. Using the kNN QSAR approach, we develop a model for the activity predictions using the distance weighted activity values of the closest neighbors[23,24]

$$y_i^{pred} = \sum_{j \neq 1}^{M} y_j \rho(d_{ij}) \tag{3}$$

where $\rho(d_{ij})$ is a weighting function dependent only on distance between points *i* and *j* in the subspace σ_M. At each point *i* this function must be constrained by the normalization condition

$$\sum_{j \neq 1}^{M} \rho(d_{ij}) = 1 \tag{4}$$

where *M* is the number of nearest neighbors. The measure of the model performance is the value of the leave-one-out cross-validated R^2 (q^2) defined as follows

$$q^2 = 1 - \frac{\sum_{i}^{N} (y_i^{actual} - y_i^{pred})^2}{\sum_{i}^{N} (y_i^{actual} - \bar{y})^2} \tag{5}$$

where $\bar{y}$ is the average value of *y* and *N* is the total number of data points.

Transformation to Hyperspherical Coordinates in the Model Subspace. We now assume that we can substitute the summation in the above eqs 3 and 4 with the integration (this follows from the conventional assumption that the activity is a continuous function of compound descriptors). The predicted value of the activity at the point $\vec{x}$ will then be expressed as

$$y^{pred}(\vec{x}) = \int_{V_{\sigma_F}} A(\vec{x}') \rho(|\vec{x}' - \vec{x}|_{\sigma_M}) d\vec{x}' \tag{6}$$

where the integration is performed over the hypersphere $V_{\sigma F}$ of some radius *R* in the full space σ_F centered at the point $\vec{x}$, with the point $\vec{x}$ itself and its close surrounding excluded from the integration volume. Below we shall discuss the meaning of the radius of this hypersphere and the need for an exclusion of the point $\vec{x}$ in more details. Note also that the distance $|\vec{x} - \vec{x}|_{\sigma_M}$ in (6) between the points $\vec{x}'$ and $\vec{x}$ is calculated in the model subspace σ_M rather than in the full space σ_F.

Now consider the following coordinate transformation: set the origin at the point $\vec{x}$ and make a transformation to hyperspherical coordinates[30] in the model subspace σ_M, while the rest of the descriptors remain intact

$$\begin{aligned}
x'_1 &= r \cos \varphi_1 \\
x'_2 &= r \sin \varphi_1 \cos \varphi_2 \\
x'_3 &= r \sin \varphi_1 \sin \varphi_2 \cos \varphi_3 \\
&\cdots \\
x'_{n-1} &= r \sin \varphi_1 \sin \varphi_2 \ldots \sin \varphi_{n-2} \cos \varphi_{n-1} \\
x'_n &= r \sin \varphi_1 \sin \varphi_2 \ldots \sin \varphi_{n-2} \sin \varphi_{n-1} \\
x'_{n+1} &= x_{n+1} \\
x'_{n+2} &= x_{n+2} \\
&\cdots \\
x'_K &= x_K
\end{aligned} \tag{7}$$

where *n* is the dimensionality of σ_M and *K* is the dimensionality of the full space σ_F.

Then the predicted value at the point $\vec{x}$ can be written as follows

$$y^{pred}(\vec{x}) = \int_{r_0}^{R} r^{n-1} dr \rho(r) \bar{A}_{\varphi}(\vec{x}, r) \tag{8}$$

where $\bar{A}_{\varphi}(\vec{x}, r)$ is defined as

$$\bar{A}_{\varphi}(\vec{x}, r) = \int_0^{\pi} \sin^{n-2} \varphi_1 d\varphi_1 \int_0^{\pi} \sin^{n-3} \varphi_2 d\varphi_2 \ldots \int_0^{\pi} \sin \varphi_{n-2} d\varphi_{n-2} \int_0^{2\pi} d\varphi_{n-1} \frac{1}{V_{n+1}} \int dx_{n+1} \ldots \frac{1}{V_K} \int dx_K \cdot A(\vec{x}) \tag{9}$$

The distance weighting function $\rho(r)$ is constrained by the normalization condition

$$\int_{V_{\sigma_F}} \rho(r) d\vec{x} = \frac{2\pi^{n/2}}{\Gamma\left(\frac{n}{2}\right)} \cdot \int_{r_0}^{R} r^{n-1} dr \rho(r) \tag{10}$$

where *n* is the dimensionality of σ_M and $\Gamma(\alpha)$ is the Euler Gamma-function.

Note, that the meaning of the integration limits r_0 and *R* in (8) and (10) is quite different. The lower limit r_0 reflects the fact that we have a discrete, but dense, set of points and that the point $\vec{x}$ and its immediate vicinity must be excluded from the integration. If it were not the case and we were dealing with a continuous data, the choice of the weighting function in the form of $\rho(|\vec{x}' - \vec{x}|) = \delta(\vec{x}' - \vec{x})$, where $\delta(\vec{x}' - \vec{x})$ is the Dirac Delta-function, would always yield the true value of the activity. Consequently, the value of σ_M is a property of the data set rather than that of the developed model. Its value can be related to the closest distance between points in a chosen subspace σ_M and, thus, to the density of the data set in the model subspace.

The upper limit of the integration *R* is, on the contrary, the parameter that can be chosen so that the best performance of the model is achieved. It is easy to see that there exists a

correspondence between R and the number of nearest neighbors M from formula (3).

Formulation of the Variational Problem. In the further development we shall use a measure of the model performance ω of the following form

$$\omega = \int_{V_F}(A(\vec{x}) - y^{pred}(\vec{x}))^2 d\vec{x} \tag{11}$$

where $y^{pred}(\vec{x})$ is defined by (8). Evidently, the requirement that ω has a minimum value is equivalent to the requirement of the closeness of q^2 in eq 5 to 1. After simple calculations we obtain the following expression for ω

$$\omega = \omega[\sigma_M, R, \rho(r)] = (\bar{A})^2 - 2\int_{r_0}^{R} r^{n-1} dr \rho(r) u(r) + \int_{r_0}^{R} r_1^{n-1} dr_1 \rho(r_1) \int_{r_0}^{R} r_2^{n-1} dr_2 (r_2) v(r_1, r_2) \tag{12}$$

where

$$u(r) = \int_{V_F} A(\vec{x}) \bar{A}_{\varphi}(\vec{x}, r) d\vec{x} \tag{13}$$

and

$$v(r_1, r_2) = \int_{V_F} \bar{A}_{\varphi}(\vec{x}, r_1) \bar{A}_{\varphi}(\vec{x}, r_2) dx \tag{14}$$

where $\bar{A}$ is an average value of $A(\vec{x})$ and $\bar{A}_{\varphi}(\vec{x}, r)$ is defined in eq 9.

We can now formulate a variational problem: among all possible models, based on the prediction in the form given by eq 8, we must find those that minimize ω subject to the normalization constraint (10). These models will have the highest accuracy. Thus, eq 12 with the constraint (10) is our basic equation which allows us to analyze the process of the model generation and discuss the possible ways to improve the model performance.

Let us consider the parameters of the models that influence ω. The variable selection or, in other words, the choice of the model subspace σ_M, is the first of them. Evidently, the closer σ_M is to the "activity-defined" subspace σ_A, the better performance of the model can be expected. Indeed, the more descriptors from σ_A and the fewer descriptors which do not belong to σ_A are present in σ_M, the more accurate the weighting process becomes, and, consequently, the closer our prediction in the eq 8 is to the actual value. Formally we can write the above requirement as

$$\frac{\delta \omega}{\delta \sigma_M} = 0 \tag{15}$$

In practice the Monte Carlo simulated annealing technique is used in order to search for the best performing descriptor subspaces. The methods based on the direct solution of the eq 15 are still to be developed.

The second parameter is the number of nearest neighbors used to evaluate the activity of any given compound. In our approach, it is the radius of the integration hypersphere R that effectively defines the number of the nearest neighbors used for the prediction. In previous calculations[23] this number is optimized by simply setting it to 1, 2, etc. until the best performance is achieved.

Finally, ω is a functional of the weighting function $\rho(r)$. Solving the equation

$$\frac{\delta \omega}{\delta \rho} = -2u(r) + \int_{r_0}^{R} r'^{n-1} dr' \rho(r') v(r, r') - \mu = 0 \tag{16}$$

for $\rho(r)$ in a given subspace σ_M will minimize the value of ω and, thus, yield the best model performance. μ in (16) is a Lagrange multiplier for the constraint (10).

We now can return to a discrete representation (summation) in order to illustrate how the above minimization procedure can be applied to a real data set. Suppose, that we chose a subspace σ_M and indexed all the pair distances between the data points in this subspace using the subscript k. Our purpose is to obtain the values $\rho_k = \rho(d_k) = \rho(d_{ij})$, where i and j are any two data points from the data set. Using eq 16, we can get a system of linear equations with respect to ρ_k

$$-2u_k + \sum_{k'} \rho_{k'} v_{kk'} - \mu = 0 \tag{17}$$

where

$$u_k = \sum_{i=1}^{N} (\bar{y}_{\varphi})_{ik} \tag{18}$$

and

$$v_{kk'} = \sum_{i=1}^{N} (\bar{y}_{\varphi})_{ik} \cdot (\bar{y}_{\varphi})_{ik} \tag{19}$$

where $(\bar{y}_{\varphi})_{ik}$ is the average activity value calculated over all points located at distance d_k from the point i

$$(\bar{y}_{\varphi})_{ik} = \sum_{j} y_j \tag{20}$$

where

$$d_{ij} = d_k$$

Solution of the system of equations (17) together with the condition (4) will define the values of ρ_k which will minimize ω or, alternatively, yield the highest q^2 for the chosen set of descriptors.

IMPLEMENTATION AND COMPUTATIONAL DETAILS

Model Development. To illustrate the advantages of the variational approach to kNN we have applied the minimization procedure described above to the problem of finding of predictive models for several experimental data sets as described below. A simulated annealing technique was used to optimize the choice of descriptor subspaces with the "temperature" decreased every time a higher value of q^2 was obtained. For each descriptor subspace we optimized the q^2 value with respect to the number of nearest neighbors (from 1 to 7 nearest neighbors considered) and with respect to the

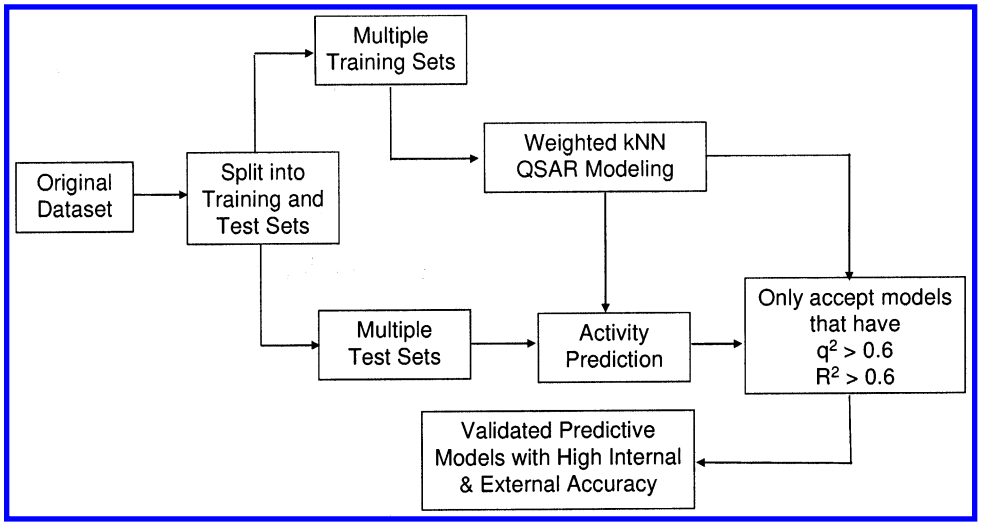

Figure 1. The workflow of the kNN QSAR modeling.

shape of the weighting function $\rho(d_{ij})$. We chose the weighting function in the form

$$\rho(d_{ij}) = \frac{e^{-\alpha d_{ij}/D}}{\sum_{j}^{M} e^{-\alpha d_{ij}/D}} \tag{21}$$

where $\bar{D}$ is the average of the sum of M shortest distances between the points in the model subspace

$$D = \frac{1}{N}\sum_{i}^{N}\sum_{j}^{M} d_{ij} \tag{22}$$

and α is a variational parameter, which was used to maximize q^2. Note, that the normalization (4) has already been applied to the weighting function $\rho(d_{ij})$ by introducing the denominator in formula (21).

For each model the calculation involved a solution of the equation

$$\frac{d(q^2)}{d\alpha} = \sum_{i}^{N}(y_i - y_i^{pred})\cdot\frac{1}{\sum_{k}^{M} e^{-\alpha d_{ik}/\bar{D}}}\cdot\sum_{j}^{M}(y_i - y_i^{pred})\cdot d_{ij}\cdot e^{-\alpha d_{ik}/\bar{D}} = 0 \tag{23}$$

with respect to α and then finding the corresponding value of q^2. We also calculated the reference q^2 value at $\alpha = 1$. We expect that a numerical solution of eq 23 takes only a small fraction of the time required for the calculation of q^2 in a given descriptor subspace. For each considered model an additional analysis of the second derivative of q^2 with respect to α was performed in order to make sure that the obtained extreme corresponded to the maximum of q^2.

Descriptors and Model Validation. All chemical structures were generated using SYBYL software.[31] Multiple descriptors derived from 2D molecular topology were used. Molecular topological indices[31,32] were generated with the MolConnZ program (MZ descriptors).[33] Overall, MolConnZ produces over 400 different descriptors. Most of these descriptors characterize chemical structure but several depend on the arbitrary numbering of atoms in a molecule and are introduced solely for bookkeeping purposes. In our study, only 312 chemically relevant descriptors were initially calculated. Depending on a particular data set, the number of descriptors used in model development ranged between 112 and 188 (after deleting descriptors with zero value or zero variance). MZ descriptors were range-scaled prior to distance calculations since the absolute scales for MZ descriptors can differ by orders of magnitude (see ref 23 for details). Accordingly, our use of range scaling avoided giving descriptors with significantly higher ranges a disproportional weight upon distance calculations in multidimensional MZ descriptor space. The overall modeling process is illustrated in Figure 1. It includes splitting the data set into the training and test set, building kNN QSAR models for the training set, and selecting validated models using several criteria. The data sets were divided following the procedure developed earlier[25,34,35] that maximizes the diversity and chemical space coverage of both training and test sets. Our approach to model validation is described in detail elsewhere.[25,35] We considered a model successful if it satisfied several criteria as follows: (i) both the q^2 value for the training set and the R^2 value for the corresponding test set exceeded 0.6; (ii) coefficients of determination (predicted versus observed activities R_0^2 and observed versus predicted activities $R'_0{}^2$) were similar to R^2, i.e., $|R^2 - R_0^2|/R^2 < 0.1$; and (iii) the slopes of regression lines of predicted vs observed k and observed vs predicted k' values were between 0.85 and 1.15.

Use of Relational Databases To Improve the Computational Efficiency. For every data set we created a relational database which contained not only the tables with descriptor and activity values but also a table with precalculated values of the squares of "one-dimensional" descriptor differences, needed for the calculations of Euclidean distances in the model subspaces. Although such tables were very large (the number of records in the table is approximately the number of compounds squared times the number of descriptors divided by 2), we found that the computational efficiency to obtain the optimal q^2 was two to three times better compared to the previous version of kNN QSAR. The time required to generate one model (one q^2 calculation for fixed

Table 1. Examples of Acceptable Models for the Artificial Data Set of 100 Compounds Characterized by 100 Descriptors[a]

model ID	no. of neighbors	no. of descr	list of used descr	α optimized	q^2 (α)	q^2 (α=1)	$[q^2(\alpha)-q^2(1)]/q^2(1)\times 100\%$	R^2 (α)	R^2 (α=1)	$[R^2(\alpha)-R^2(1)]/R^2(1)\times 100\%$	size, test set
560	7	5	1,2,4,5,6	27.016	0.732	0.570	28.4	0.828	0.769	7.6	17
567	7	6	1,2,4,5,6,7	35.733	0.693	0.563	23.2	0.682	0.680	0.3	17
559	6	5	1,2,4,5,6	20.611	0.740	0.605	22.3	0.836	0.809	3.3	20
392	7	5	1,2,3,4,5	31.070	0.758	0.643	17.8	0.633	0.633	0.1	17
590	2	4	1,2,3,5	12.638	0.627	0.535	17.3	0.744	0.582	27.9	17
755	6	5	1,2,3,4,5	36.725	0.768	0.692	11.0	0.705	0.548	28.6	20
602	7	5	1,2,3,5,6	49.226	0.756	0.650	16.2	0.811	0.614	32.1	9
391	6	5	1,2,3,4,5	25.444	0.757	0.653	15.9	0.631	0.646	−2.3	17

[a] The activity value was generated using the analytical formula (24) which involved descriptors 1 through 5.

Table 2. Examples of Successful Models for the Toxicity Data Set[36] of 250 Compounds with 173 Descriptors

model ID	no. of neighbors	no. of descr	α optimized	q^2 (α)	q^2 (α=1)	$[q^2(\alpha)-q^2(1)]/q^2(1)\times 100\%$	R^2 (α)	R^2 (α=1)	$[R^2(\alpha)-R^2(1)]/R^2(1)\times 100\%$	size, test set
2421	7	18	37.36	0.742	0.686	8.2	0.715	0.650	10.0	31
2183	7	10	31.09	0.757	0.700	8.1	0.745	0.700	6.4	31
2582	7	22	36.62	0.765	0.708	8.1	0.743	0.688	7.9	31
211	7	28	31.44	0.727	0.675	7.7	0.704	0.655	7.6	42
2581	6	22	28.87	0.764	0.712	7.3	0.743	0.684	8.5	31
2420	6	18	30.18	0.739	0.691	7.0	0.718	0.667	7.8	31
1163	7	22	29.97	0.749	0.703	6.6	0.711	0.628	13.2	42
624	7	40	44.72	0.775	0.728	6.4	0.720	0.638	12.8	50
210	6	28	25.05	0.728	0.688	5.8	0.705	0.664	6.1	42
2421	7	18	37.36	0.742	0.686	8.2	0.715	0.650	10.0	31
2183	7	10	31.09	0.757	0.700	8.1	0.745	0.700	6.4	31
1887	5	36	23.89	0.768	0.731	5.1	0.763	0.769	−0.9	31

set of descriptors and fixed number of nearest neighbors) ranged from 9 ms (8 ms without optimization) for small data sets to 25 ms (23 ms without optimization) for larger data sets on a Pentium IV Windows PC. In addition, the coding was greatly simplified due to the use of database aggregate and sorting functions. All the intermediate and final results of the calculations, such as a list of descriptors, q^2 and R^2 values, optimized α values and the reference q^2 and R^2 values at $\alpha = 1$ were stored in a separate table so that no calculation had to be done more than once. The same table was later used for the analysis of the resulting models. We concluded that the time needed to generate these tables was negligible.

RESULTS

We have applied our modified kNN QSAR approach to several data sets. All calculations reported below started by deriving models using conventional calculations, i.e., fixed weighting function and predefined number of k nearest neighbors. We then sought model improvement by the means of the modified procedure as described in the Implementation and Computational Details.

Artificial (Toy) Set. This artificially generated set included 100 compounds characterized by 100 descriptors; the descriptors were generated as random numbers. The activity was calculated using an analytical formula, which simulated the gravitational potential created by spheres of various density and diameter at the origin

$$y_i = y_i(x_1^i, x_2^i, x_3^i, x_4^i, x_5^i) = \frac{100 \cdot x_1^i \cdot (x_2^i)^3}{\sqrt{(x_3^i)^2 + (x_4^i)^2 + (x_5^i)^2}} \qquad (24)$$

where the descriptors x_1 and x_2 signified respectively the density and radius of the sphere, centered at the point with coordinates (x_3, x_4, x_5).

For this data set we have generated all possible models, rather than using stochastic descriptor sampling, in the subspaces containing up to 9 descriptors. We have identified 15 descriptor subspaces which produced 64 successful models for 3 different training/test set divisions. Three models had one nearest neighbor and, thus, could not be improved. The highest observed improvements of q^2 and R^2 were 28.4% and 44.9%, respectively. The average improvement of q^2 and R^2 was 6.7% and 5.9%, respectively. Of these 64 models, only one showed the decrease of the R^2 value comparing to the case when $\alpha = 1$, whereas 60 models (3 models with one nearest neighbors could not be improved) showed an improvement for both q^2 and R^2 values. The typical results of our calculations are shown in Table 1.

Toxicity Data Set. This data set[36] contains 250 compounds with known toxicity values characterized by 173 MolconnZ descriptors. We have identified 163 descriptor subsets which afforded 444 successful models for 3 different training/test set divisions. 49 of these models had one nearest neighbor and, thus, could not be improved. The highest observed improvement of q^2 and R^2 was 16.7% and 19.3%, respectively. The average improvement of q^2 and R^2 was 3.4% and 3.9%, respectively. Of 395 models, 358 showed the improvement of R^2 after the optimization, whereas 37 models showed the decrease of the R^2 value. Typical models are presented in Table 2.

Anticonvulsant Data Set. This experimental data set[37] contains 48 compounds with known anticonvulsant activity. 188 MolconnZ descriptors were available for each compound. We have identified 142 descriptor subsets, which afforded 170 acceptable models for 3 different training/test set divisions. 8 of these models had one nearest neighbor and, thus, could not be improved. The highest observed improvement of R^2 was more than 4-fold (from 0.157 at $\alpha = 1$ to 0.635), and the highest observed R^2 improvement

Table 3. Examples of Successful Models for the Anticonvulsant Data Set[37] of 48 Compounds with 188 Descriptors per Compound

model ID	no. of neighbors	no. of descr	α optimized	q^2 (α)	q^2 (α=1)	$[q^2(\alpha)-q^2(1)]/q^2(1)\times 100\%$	R^2 (α)	R^2 (α=1)	$[R^2(\alpha)-R^2(1)]/R^2(1)\times 100\%$	size, test set
363302	6	18	87.51	0.635	0.157	304.8	0.606	0.485	24.9	7
421991	7	24	93.76	0.625	0.172	264.1	0.653	0.389	67.8	10
425028	6	24	87.51	0.618	0.211	193.5	0.638	0.500	27.5	7
424448	7	24	81.27	0.632	0.237	166.8	0.667	0.395	68.7	9
413325	7	24	93.76	0.610	0.23	165.1	0.765	0.545	40.3	7
424504	7	24	81.27	0.610	0.231	163.6	0.749	0.336	123.3	7
424426	6	24	62.54	0.624	0.238	162.5	0.643	0.466	37.9	7
421758	5	24	53.17	0.601	0.273	119.9	0.692	0.521	32.7	9
426454	4	24	43.81	0.603	0.288	109.2	0.750	0.411	82.7	7
426453	3	24	12.59	0.623	0.394	58.2	0.774	0.624	23.9	7
424348	5	24	53.17	0.621	0.258	140.2	0.665	0.669	-0.6	7

Table 4. Examples of Successful Models for the Dopamine D1 Antagonists Data Set[38] of 29 Compounds with 112 Descriptors per Compound

model ID	no. of neighbors	no. of descr	α optimized	q^2 (α)	q^2 (α=1)	$[q^2(\alpha)-q^2(1)]/q^2(1)\times 100\%$	R^2 (α)	R^2 (α=1)	$[R^2(\alpha)-R^2(1)]/R^2(1)\times 100\%$	size, test set
22026	7	6	51.61	0.738	0.351	110.0	0.647	0.344	88.0	6
22306	7	6	68.78	0.720	0.363	98.6	0.652	0.444	46.7	7
22305	6	6	25.07	0.722	0.426	69.4	0.641	0.439	45.8	7
22304	5	6	31.32	0.716	0.451	58.8	0.654	0.533	22.6	7
22025	6	6	42.25	0.734	0.488	50.3	0.667	0.383	74.0	7
22024	5	6	28.2	0.735	0.525	40.0	0.623	0.395	57.6	7
22366	4	6	16.88	0.711	0.583	21.9	0.615	0.539	14.0	7
22022	3	6	16.49	0.704	0.583	20.8	0.691	0.390	77.1	7
20272	3	6	50.05	0.742	0.655	13.3	0.604	0.434	39.0	6
20271	2	6	50.05	0.733	0.653	12.3	0.616	0.531	16.0	7

Table 5. Examples of Successful Models for the Protein Complexes of 264 Compounds with 100 Descriptors[39]

model ID	no. of neighbors	no. of descr	α optimized	q^2 (α)	q^2 (α=1)	$[q^2(\alpha)-q^2(1)]/q^2(1)\times 100\%$	R^2 (α)	R^2 (α=1)	$[R^2(\alpha)-R^2(1)]/R^2(1)\times 100\%$	size, test set
664	7	15	50.84	0.621	0.515	20.6	0.605	0.515	17.4	50
663	6	15	40.39	0.621	0.524	18.5	0.603	0.519	16.2	44
164155	5	20	26.83	0.634	0.546	16.3	0.603	0.597	1.0	38
138689	5	20	28.64	0.622	0.539	15.3	0.604	0.544	10.9	38
157366	6	20	34.34	0.620	0.540	14.8	0.607	0.565	7.4	35
148081	3	20	11.75	0.638	0.593	7.5	0.604	0.593	1.9	38
141494	3	20	11.85	0.641	0.596	7.5	0.606	0.590	2.8	38
892	4	50	6.96	0.659	0.656	0.4	0.607	0.585	3.9	50
206	5	60	18.30	0.644	0.618	4.2	0.701	0.711	−1.5	50

was 123% (from 0.336 to 0.749). The average improvement of q^2 and R^2 was 118.3% and 93.5%, respectively. Of these 162 models, 157 showed the improvement of R^2 after the optimization, whereas 5 models showed the decrease of the R^2 value. Typical models are presented in Table 3.

Dopamine D1 Antagonists Data Set. This data set[38] contained 29 compounds with known activity values characterized by 112 MolconnZ descriptors. We have identified 18 descriptor subsets which afforded 40 acceptable models for 2 different training/test set divisions. The highest observed improvement of q^2 and R^2 was 110% and 88.0%, respectively. The average improvement of q^2 and R^2 was 48.1% and 62.7%, respectively. Out of 40 models 37 showed the improvement of R^2 after the optimization, whereas 3 models showed the decrease of the R^2 value. Typical models are presented in Table 4.

Protein−Ligand Complexes. This data set includes 264 diverse protein−ligand complexes with known X-ray characterized geometry and known binding constants. Earlier, we have developed chemical geometrical descriptors characterizing protein−ligand interfaces.[39] We have identified 60 descriptor subsets which produced 67 successful models for 4 different training/test set divisions. None of these models had one nearest neighbor. The highest observed improvements of q^2 and R^2 were 20.6% and 17.4%, respectively. The average improvements of q^2 and R^2 were 3.5% and 1.1%, respectively. 40 of these 67 models showed the improvement of R^2 after the optimization, whereas 27 models showed the decrease of the R^2 value. Typical models are presented in Table 5.

DISCUSSION

In this paper we have focused on the optimization of several factors that control the performance of kNN QSAR model development such as the choice of descriptors, the number of nearest neighbors used by the model, and the weighting function to evaluate contributions of nearest neighbors of a compound to the predicted value of its activity. We started from the analysis of the results for the toy set where the activity formula is known. We found that the best performing models that resulted from the calculations indeed included all the significant descriptors (i.e., those that were used in eq 24 to calculate the "activity" of the training set). There were very few, if any, of other descriptors included in these models. Indeed, as can be seen from the transformation (7), the "angular" averaging in eq 9 yields best results if it includes the contributions from the descriptors of the

true activity space σ_A, whereas the inclusion of irrelevant descriptors could only increase the noise.

The analysis of all results allows us to conclude that models with the high number of nearest neighbors are subject to the greatest improvement due to the optimization of α values (evidently, the models with only one nearest neighbor could not be improved, since $\rho(d_{ij})$ for them is equal to 1). This can be explained by considering the formalism developed above. Indeed, to get a reasonable estimate of the activity value at the point $\vec{x}$ in (8), we must average over all "angular" coordinates (see eq 9). Thus, in a successful model we expect to see contributions from points located in different directions from the point $\vec{x}$. This is achieved by increasing the number of nearest neighbors M. On the other hand, since for large M not all of the points are sufficiently close to $\vec{x}$, the distance weighting function $\rho(d_{ij})$ becomes more narrow (this corresponds to high α values) to compensate for the effect of the inclusion of more distant points.

CONCLUSIONS

We have developed a formalism allowing us to consider the kNN QSAR modeling as a variational problem. We have obtained a quantitative measure of the model performance ω (see eq 12). We have demonstrated that in the ideal case when the data set is very dense and the generated descriptors are independent and form a complete space, the model performance is determined by three factors: the variable selection, the number of nearest neighbors, and the shape of the distance weighting function. We also provided the equations for the determination of the shape of the weighting function that maximizes the model performance.

We have applied this approach to several experimental data sets and concluded that we could achieve a significant improvement of q^2 (improving its values by 0.1 to 0.7) for all data sets. It is remarkable that the improvement in q^2 values almost certainly leads to the improvement of the predictive R^2 values compared to the nonoptimized case. Less than 5% of all successful models exhibited the decrease of the R^2 values. The worst decrease in R^2 values never exceeded 6% after the optimization. The optimization with respect to the shape of the weighting function yields a considerable improvement of predictions in all cases and is relatively inexpensive (it takes only a small fraction of the time needed to calculate q^2).

We have also demonstrated that the new computational procedure based on the use of relational databases significantly increased the speed of calculations and greatly simplified the coding due to the use of the optimized built-in aggregate and sorting functions. The storage of all previously calculated models also improved the performance since no calculation had to be done more than once.

We conclude that the approach and procedures introduced in this paper afford better models due to their optimization with respect to the distance weighting function and the use of relational databases. The results of this study suggest that model optimization should be included as a standard component in kNN QSAR modeling. The modified kNN QSAR software is available from the authors upon request.

ACKNOWLEDGMENT

The authors thank Dr. A. Golbraikh for his interest to this work and valuable discussions. This research was supported by the NIH research grant GM066940. The authors also acknowledge Tripos, Inc. for the software grant.

REFERENCES AND NOTES

(1) Sutter, J. M.; Dixon, S. L. Automated Descriptor Selection for Quantitative Structure−Activity Relationships Using Generalized Simulated Annealing. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 77−84.
(2) Rogers, D.; Hopfinger, A. J. Application of Genetic Function Approximation to Quantitative Structure−Activity Relationships and Quantitative Structure−Property Relationships. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 854−866.
(3) Kubinyi, H. Variable Selection in QSAR Studies. I. An Evolutionary Algorithm. *Quant. Struct.−Act. Relat.* **1994**, *13*, 285−294.
(4) Kubinyi, H. Selection in QSAR Studies. II. A Highly Efficient Combination of Systematic Search and Evolution. *Struct.−Act. Relat.* **1994**, *13*, 393−401.
(5) Luke, B. T. Evolutionary Programming Applied to the Development of Quantitative Structure−Activity Relationships and Quantitative Structure−Property Relationships. *Chem. Inf. Comput. Sci.* **1994**, *34*, 1279−1287.
(6) So, S. S.; Karplus, M. Evolutionary Optimization in Quantitative Structure−Activity Relationship: an Application of Genetic Neural Networks. *J. Med. Chem.* **1996**, *39*, 1521−1530.
(7) Baroni, M.; Costantino, G.; Cruciani, G.; Riganelli, D.; Valigi, R.; Clementi, S. Generating Optimal Linear PLS Estimations (GOLPE): An Advanced Chemometric Tool for Handling 3D-QSAR Problems. *Quant. Struct.−Act. Relat.* **1993**, *12*, 9−20.
(8) Tropsha, A. Recent Trends in Quantitative Structure−Activity Relationships. In *Burger's Medicinal Chemistry and Drug Discovery*; Abraham, D., Ed.; John Wiley & Sons: New York, 2003; Vol. 1.
(9) Andrea, T. A.; Kalayeh, H. Applications of Neural Networks in Quantitative Structure−Activity Relationships of Dihydrofolate Reductase Inhibitors. *J. Med. Chem.* **1991**, *34*, 2824−2836.
(10) So, S. S.; Richards, W. G. Application of Neural Networks: Quantitative Structure−Activity Relationships of the Derivatives of 2,4-Diamino-5-(Substituted-Benzyl) Pyrimidines As DHFR Inhibitors. *J. Med. Chem.* **1992**, *35*, 3201−3207.
(11) Ajay, A. Unified Framework for Using Neural Networks to Build QSARs. *J. Med. Chem.* **1993**, *36*, 3565−3571.
(12) Hirst, J. D.; King, R. D.; Sternberg, M. J. Quantitative Structure−Activity Relationships by Neural Networks and Inductive Logic Programming. I. The Inhibition of Dihydrofolate Reductase by Pyrimidines. *J. Comput.-Aided Mol. Des.* **1994**, *8*, 405−420.
(13) Hirst, J. D.; King, R. D.; Sternberg, M. J. Quantitative Structure−Activity Relationships by Neural Networks and Inductive Logic Programming. II. The Inhibition of Dihydrofolate Reductase by Triazines. *J. Comput.-Aided Mol. Des.* **1994**, *8*, 421−432.
(14) Tetko, I. V.; Tanchuk, V. Yu.; Chentsova, N. P.; Antonenko, S. V.; Poda, G. I.; Kukhar, V. P.; Luik, A. I. HIV-1 Reverse Transcriptase Inhibitor Design Using Artificial Neural Networks. *J. Med. Chem.* **1994**, *37*, 2520−2526.
(15) Manallack, D. T.; Ellis, D. D.; Livingstone, D. J. Analysis of Linear and Nonlinear QSAR Data Using Neural Networks. *J. Med. Chem.* **1994**, *37*, 3758−3767.
(16) Maddalena, D. J.; Johnston, G. A. Prediction of Receptor Properties and Binding Affinity of Ligands to Benzodiazepine/GABAA Receptors Using Artificial Neural Networks. *J. Med. Chem.* **1995**, *38*, 715−724.
(17) Peterson, K. L. Quantitative Structure−Activity Relationships in Carboquinones and Benzodiazepine Using Counter-Propagation Neural Networks. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 896−904.
(18) Bolis, G.; Pace, L.; Fabrocini, F. A. Machine Learning Approach to Computer-Aided Molecular Design. *J. Comput.-Aided Mol. Des.* **1991**, *5*, 617−628.
(19) King, R. D.; Muggleton, S.; Lewis, R. A.; Sternberg, M. J. Drug Design by Machine Learning: The Use of Inductive Logic Programming to Model the Structure-Activity Relationships of Trimethoprim Analogues Binding to Dihydrofolate Reductase. *Proc. Natl. Acad. Sci. U.S.A.* **1992**, *89*, 11322−11326.
(20) King, R. D.; Muggleton, S.; Srinivasan, A.; Sternberg, M. J. Structure−Activity Relationships Derived by Machine Learning: The Use of Atoms and Their Bond Connectivities to Predict Mutagenicity by Inductive Logic Programming. *Proc. Natl. Acad. Sci. U.S.A.* **1996**, *93*, 438−442.
(21) Jain, A. N.; Dietterich, T. G.; Lathrop, R. H.; Chapman, D.; Critchlow, R. E., Jr.; Bauer, B. E.; Webster, T. A.; Lozano-Perez, T. Shape-Based Machine Learning Tool for Drug Design. *J. Comput.-Aided Mol. Des.* **1994**, *8*, 635−652.
(22) Hirst, J. D. Nonlinear Quantitative Structure−Activity Relationship for the Inhibition of Dihydrofolate Reductase by Pyrimidines. *J. Med. Chem.* **1996**, *39*, 3526−3532.

(23) Zheng, W.; Tropsha, A. A Novel Variable Selection QSAR Approach Based on the K-Nearest Neighbor Principle. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 185−194.
(24) Shen, M.; LeTiran, A.; Xiao, Y.; Golbraikh, A.; Kohn, H.; Tropsha, A. Quantitative Structure−Activity Relationship Analysis of Functionalized Amino Acid Anticonvulsant Agents Using k Nearest Neighbor and Simulated Annealing PLS Methods. *J. Med. Chem.* **2002**, *45*, 2811−2823.
(25) Golbraikh, A.; Tropsha, A. Beware of Q2! *J. Mol. Graph. Model.* **2002**, *20*, 269−276.
(26) Shen, M.; Xiao, Y.; Golbraikh, A.; Gombar, V. K.; Tropsha, A. Development and Validation of K−Nearest-Neighbor QSPR Models of Metabolic Stability of Drug Candidates. *J. Med. Chem.* **2003**, *46*, 3013−3020.
(27) Kovatcheva, A.; Golbraikh, A.; Oloff, S.; Xiao, Y. D.; Zheng, W.; Wolschann, P.; Buchbauer, G.; Tropsha, A. Combinatorial QSAR of Ambergris Fragrance Compounds. *J. Chem. Inf. Comput. Sci*. **2004**, in press.
(28) Sharaf, M. A.; Illman, D. L.; Kowalski, B. R. *Chemometrics*; John Wiley & Sons: New York, 1986.
(29) McPherson, G. *Applying and Interpreting Statistics*; Springer-Verlag: 1990.
(30) Fichtengol'tz, G. M. *Differencial and Integral Calculus*; Nauka: Moscow, Russia, 1969.
(31) The program Sybyl is available from Tripos Associates, St. Louis, MO.
(32) Randic, M. On Characterization of Molecular Branching. *J. Am. Chem. Soc.* **1975**, *97*, 6609−6615.
(33) EduSoft. MolconnZ. [**4.05**]. 2003.
(34) Golbraikh, A.; Shen, M.; Xiao, Z.; Xiao, Y. D.; Lee, K. H.; Tropsha, A. Rational Selection of Training and Test Sets for the Development of Validated QSAR Models. *J. Comput.-Aided Mol. Des.* **2003**, *17*, 241−253.
(35) Golbraikh, A.; Tropsha, A. Predictive QSAR Modeling Based on Diversity Sampling of Experimental Datasets for the Training and Test Set Selection. *J. Comput.-Aided Mol. Des.* **2002**, *16*, 357−369.
(36) Cronin, M. T.; Aptula, A. O.; Duffy, J. C.; Netzeva, T. I.; Rowe, P. H.; Valkova, I. V.; Schultz, T. W. Comparative Assessment of Methods to Develop QSARs for the Prediction of the Toxicity of Phenols to Tetrahymena Pyriformis. *Chemosphere* **2002**, *49*, 1201−1221.
(37) Shen, M.; Beguin, C.; Golbraikh, A.; Stables, J. P.; Kohn, H.; Tropsha, A. Application of Predictive QSAR Models to Database Mining: Identification and Experimental Validation of Novel Anticonvulsant Compounds. *J. Med. Chem.* **2004**, *47*, 2356−2364.
(38) Hoffman, B.; Cho, S. J.; Zheng, W.; Wyrick, S.; Nichols, D. E.; Mailman, R. B.; Tropsha, A. Quantitative Structure−Activity Relationship Modeling of Dopamine D(1) Antagonists Using Comparative Molecular Field Analysis, Genetic Algorithms-Partial Least-Squares, and K Nearest Neighbor Methods. *J. Med. Chem* **1999**, *42*, 3217−3226.
(39) Zhang S.; Golbraikh A.; Tropsha, A. Development of Novel Geometrical Chemical Descriptors and Their Application to the Prediction of Ligand−Receptor Binding Affinity. *Proceedings, 227-th National Meeting of the American Chemical Society, COMP-76* 2004.

CI049628+