Adaptive Neuro-Fuzzy Inference System: An Instant and Architecture-Free Predictor for Improved QSAR Studies

Yannis L. Loukas*

Department of Pharmaceutical Chemistry, School of Pharmacy, University of Athens, Panepistimiopolis, Zografou, 157 71 Athens, Greece

Received May 30, 2000

The application of an adaptive neuro-fuzzy inference system (ANFIS) has been developed for obtaining sufficient quantitative structure−activity relationships (QSAR) with high accuracy. To this end, a data set of 68 pyrimidines derivatives as DHFR inhibitors, described first in the excellent independent studies of Hansch et al. (*J. Med. Chem.* **1982**, *25*, 777−784 and *J. Med. Chem.* **1991**, *34*, 46−54) and later by So and Richards (*J. Med. Chem.* **1992**, *35*, 3201−3207), was examined. The ANFIS system, first time applied in the literature to QSAR studies, was trained using a hybrid algorithm consisting of back-propagation and least-squares estimation while the optimum number and shape of membership functions were obtained through the subtractive clustering algorithm. Prior to the development and evaluation of the ANFIS system, geometry optimization of the examined compounds was performed, deriving a series of diverse descriptors from which the best subset was selected by using a hybrid genetic algorithm system. The predictive abilities of the resulting models compared to those produced from classical multivariate regression such as linear and nonlinear (quadratic) partial least squares regression (PLS and QPLS, respectively). The ANFIS method outperformed both the PLS models as well as the published results, leading to substantial gain in both the prediction ability and the computation speed (almost instant training).

I. Introduction

Classical quantitative structure−activity relationship (QSAR) studies are used for both the selection of principal physicochemical characteristics (descriptors) and relating them to biological activities and the derivation of mathematical models that involve these multivariate data in order to be used for predictive purposes in drug design. The generation of a wide range of molecular and substituent descriptors based on molecular modeling techniques has become a routine part of QSAR studies. This has created a challenging problem in the selection of the best variables to use for the QSARs. In many situations, typical of the drug design process, QSAR applications are trying to solve an underdetermined problem in which there are more variables (descriptors) than objects (compounds). Moreover, the underlying physical properties of the drugs that are correlated with their biological responses are often unknown so that a priori feature selection is not possible in most cases. This difficulty has been summarized by Kubinyi[1] in suggesting that "selection of variables is time-consuming, difficult and, despite many different statistical criteria for the evaluation of the resulting models, a highly subjective and ambiguous procedure".

In this study we examine a method which represents one of the most successful combinations of feature selection and feature mapping tools. This includes the combination of genetic algorithms (GA, a method which deals with high-dimensional data sets, i.e., a large number of descriptors and often a limited number of compounds) for the selection of descriptors with adaptive neuro-fuzzy inference system (ANFIS) for the correlation of the selected descriptors with activity. GA/ANFIS should be considered superior to the combination of genetic algorithms with regression analysis (GA/RA). Although GA/RA descriptors give an optimal regression model, they cannot capture descriptors for a nonlinear model.[2]

Over the past decade or so, significant advances have been made in two distinct technological areas: fuzzy logic (FL) and neural networks (NNs). The synergism of FL systems and NN has produced a functional system capable of learning, high-level thinking, and reasoning. It is an improved tool for determining the behavior of imprecisely defined complex systems. The purpose of a neuro-fuzzy system is to apply neural learning techniques to identify and tune the parameters and/or structure of neuro-fuzzy systems. These neuro-fuzzy systems can combine the benefits of these two powerful paradigms into a single capsule. They have several features, which make them suitable for a wide range of scientific applications. These strengths include fast and accurate learning, good generalization capabilities, excellent explanation facilities in the form of semantically meaningful fuzzy rules, and the ability to accommodate both data and existing expert knowledge about the problem under consideration.

The goal of ANFIS[3] is to find a *model* or *mapping* that will correctly associate the inputs (descriptors) with the target (activity). Fuzzy inference system (FIS)[4] is a knowledge representation where each fuzzy rule describes a local behavior of the system. If we view a FIS as a feedforward network structure where the primary inputs and intermediate results are being sent around to compute the final output, then we can apply the same back-propagation principle in neural networks. The

* Phone: +301 7274 224. Fax: +301 6130 285. E-mail: loukas@pharm.uoa.gr.

10.1021/jm000226c CCC: $20.00
Published on Web 07/21/2001

network structure that implements FIS is referred to as ANFIS and employs hybrid learning rules to train a Sugeno-style FIS[5] with linear rule outputs (see later).

Among various combinations of methodologies in soft computing, the one that has highest visibility at this juncture is that of fuzzy logic and neurocomputing, leading to so-called neuro-fuzzy systems. Within fuzzy logic, such systems play a particularly important role in the induction of rules from observations. It can be a very powerful tool for dealing quickly and efficiently with imprecision and nonlinearity as happens, for instance, in QSAR studies. The basic idea behind these neuro-adaptive learning techniques works similarly to that of neural networks. These techniques provide a method for the fuzzy modeling procedure to learn information about a data set, in order to compute the membership function parameters that best allow the associated fuzzy inference system to track the given input/output data. A network-type structure similar to that of a neural network, which maps inputs through input membership functions and associated parameters, and then through output membership functions and associated parameters to outputs, can be used to interpret the input/output map. This eliminates the disadvantage of a normal feedforward multilayer network, which is difficult for an observer to understand or to modify. In the following we shall explain how to use ANFIS to train a fuzzy inference system that predicts the biological activity of a set of pyrimidines derivatives.

II. Methods

A. Basic Definitions and Terminology. Let X be a space of objects and x be a generic element of X. A classical set $A \subseteq X$ is defined as a collection of elements or objects $x \in X$ such that each x can either belong or not belong to the set A. By defining a *characteristic function* for each element x in X, we can represent a classical set A by a set of ordered pairs $(x, 0)$ or $(x, 1)$ which indicates $x \notin A$ or $x \in A$, respectively. Unlikely, a fuzzy set expresses the degree to which an element belongs to a set. Hence the characteristic function of a fuzzy set is allowed to have values between 0 and 1, which denotes the degree of membership of an element in a given set. So a fuzzy set A in X is defined as a set of ordered pairs: $A = \{(x, \mu_A(x)) | x \in X\}$, where $\mu_A(x)$ is called the *membership function* (MF) for the fuzzy set A. The MF maps each element of X to a membership grade (or value) between 0 and 1. Usually X is referred to as the *universe of discourse* or simply the *universe*.

In practice, when the universe of discourse X is a continuous space we usually partition X into several fuzzy sets whose MFs cover X in a more or less uniform manner. These fuzzy sets, which usually carry names such as "high", "middle", or "low" are called linguistic values. Suppose that X represents the molecular weight (MW) of the pyrimidine derivatives, thus we can define three linguistic fuzzy sets (three bell MFs) for MW as displayed in Figure 1a. Similarly, we define three bell MFs (not shown) for all the input variables (the examined descriptors) either manually (grid partition) or by applying a clustering procedure (see later). In Table 1 the parameters associated with the three bell MFs, describing the six inputs, are presented.

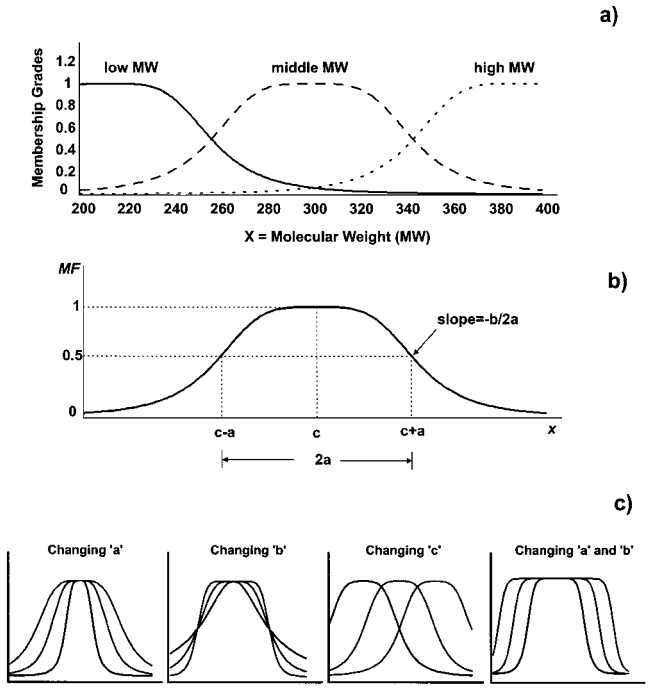

Figure 1. (a) Typical bell MFs of linguistic values "low", "middle", and "high" for the molecular weight MW of the pyrimidine derivatives; (b) physical meaning of parameters in a generalized bell MF; (c) effects of changing parameters in bell MFs.

Table 1. Parameters $\{a, b, c\}$ Associated with Each of the Three Bell MFs per Input Having the Linguistic Labels "Low", "Middle", and "High" [a]

	$\{a, b, c\}$ parameters before training ANFIS		
	low	middle	high
MW_mol	{47.08, 2, 200.3}	{47.08, 2, 294.4}	{47.08, 2, 388.6}
Vol_mol	{42.29, 2, 135.2}	{42.29, 2, 219.8}	{42.29, 2, 304.4}
MW_sub_3[b]	{57.81, 2, 1.01}	{57.81, 2, 116.6}	{57.81, 2, 232.3}
p_ch_2[c]	{0.02263, 2, 0.1708}	{0.02263, 2, 0.216}	{0.02263, 2, 0.2613}
p_ch_4[c]	{0.0199, 2, 0.1764}	{0.0199, 2, 0.2164}	{0.0199, 2, 0.2564}
pc_h_sum[d]	{0.1425, 2, −1.983}	{0.1425, 2, −1.698}	{0.1425, 2, −1.41}

[a] For the physical meaning of these parameters, see Methods and Figure 1. [b] MW of the substituents at position 3 of the benzyl ring. [c] Partial charges of atoms 2 and 4 (Figure 5). [d] Sum of partial charges of heavy atoms n ($n = 1-15$, Figure 5).

There are several classes of MFs with each one to be characterized from a set of parameters. The most widely used MF is the generalized bell MF (or bell MF), which is specified by three parameters $\{a, b, c\}$

$$\text{bell}(x;a,b,c) = \frac{1}{1 + |\frac{x - c}{a}|^{2b}}$$

where the parameter b is usually positive. A desired bell MF can be obtained by a proper selection of the parameter set $\{a, b, c\}$. Each of these parameters has a physical meaning: c determines the center of the corresponding membership function; a is the half width; and b (together with a) controls the slopes at the crossover points (where MF value is 0.5). Figure 1b shows these concepts while Figure 1c further illustrates the effects of changing each parameter. Further to the bell MF should be used the triangular, trapezoidal, and Gaussian MFs. Figure 2 represents the shape and the associated parameters for each one of these MFs.[6] The initial values of premise parameters are set in such a

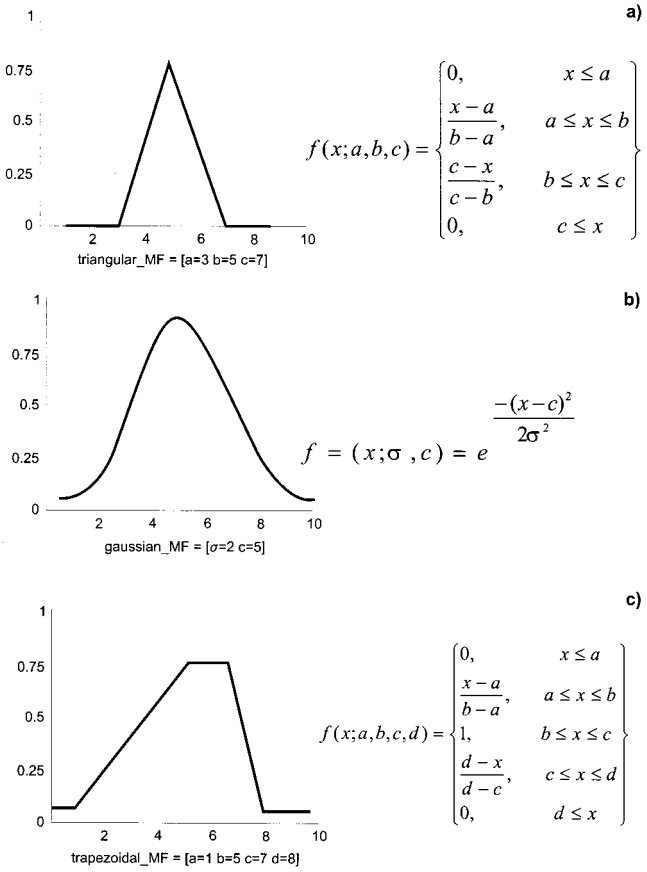

Figure 2. Examples of three membership functions and their associated parameters: (a) triangular, (b) Gaussian, and (c) trapezoidal.

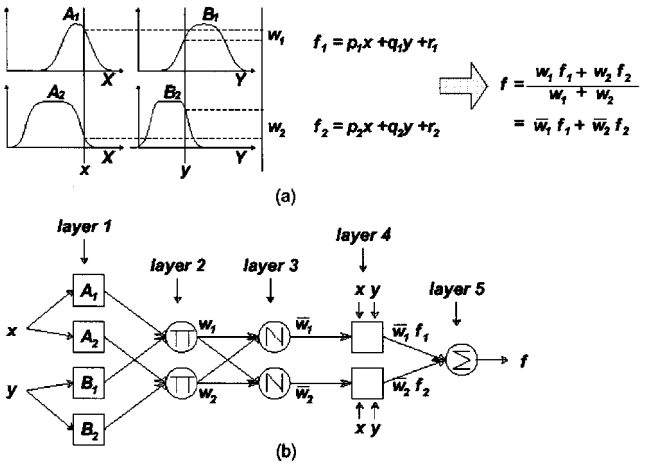

Figure 3. (a) A two-input first-order Sugeno fuzzy model with two rules; (b) equivalent ANFIS architecture.

way that the MF's are equally spaced along the operating range of each input variable. Furthermore, MFs are defined in a way that the fuzzy inference system can provide smooth transition and sufficient overlapping from one linguistic label to another.[7,8] During the learning phase of ANFIS, these parameters are changing continuously in order to minimize the error function between the target output values (activities) and the calculated ones.

B. ANFIS Architecture. The proposed neuro-fuzzy model in ANFIS is a multiplayer neural network-based fuzzy system. Its topology is shown in Figure 3, and the system has a total of five layers. In this connectionist structure, the input and output nodes represent the descriptors and the activity, respectively, and in the hidden layers, there are nodes functioning as membership functions (MFs) and rules. This eliminates the disadvantage of a normal feedforward multilayer network, which is difficult for an observer to understand or to modify. For simplicity, we assume that the examined fuzzy inference system has two inputs x and y (suppose the molecular weight MW and the molecular volume Vol of the pyrimidines derivatives) and one output, the activity. For a first-order Sugeno fuzzy model,[9,10] a common rule set with two fuzzy if-then rules is the following:

Rule 1: If MW is A_1 and Vol is B_1,
then $f_1 = p_1x + q_1y + r_1$

Rule 2: If MW is A_2 and Vol is B_2,
then $f_2 = p_2x + q_2y + r_2$

Figure 3 illustrates the reasoning mechanism for this Sugeno model and the corresponding equivalent ANFIS architecture, where nodes of the same layer have similar functions (the output of the ith node in layer l is denoted as $O_{l,i}$).

Layer 1: Every node i in this layer is an adaptive node with a node function

$$O_{1,i} = \mu_{A_i}(x), \quad \text{for } i = 1, 2$$

where x is the input to node i, and A_i is the linguistic label (low, high, etc.) associated with this node function. In other words, $O_{1,i}$ is the membership function of A_i, and it specifies the degree to which the given x satisfies the quantifier A_i. Usually we choose $\mu_{A_i}(x)$ to be bell-shaped with maximum equal to 1 and minimum equal to 0, such as the generalized bell function defined above. As the values of the parameters $\{a_i, b_i, c_i\}$ change, the bell-shaped functions vary accordingly, thus exhibiting various forms of membership functions on linguistic label A_i. Parameters in this layer are referred to as *premise parameters*.

Layer 2: Every node in this layer is a fixed node labeled Π (Figure 3), whose output is the product of all the incoming signals:

$$O_{2,i} = w_i = \mu_{A_i}(x) \times \mu_{B_i}(y), \quad \text{for } i = 1, 2$$

Each node output represents the *firing strength* of a rule.

Layer 3: Every node in this layer is a fixed node labeled N. The ith node calculates the ratio of the ith rule's firing strength to the sum of all rules' firing strengths:

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1, 2$$

For convenience, outputs of this layer are called **normalized firing strengths.**

Layer 4: Every node i in this layer is an adaptive node with a node function

$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i(p_i x + q_i y + r_i)$$

where $\bar{w}_i$ is a *normalized firing strength* from layer 3 and $\{p_i, q_i, r_i\}$ is the parameter set of this node. Parameters in this layer are referred to as *consequent parameters*.

Layer 5: The single node in this layer is a fixed node labeled $\sum$, which computes the overall output as the summation of all incoming signals:

$$\text{overall output} = O_{5,i} = \sum_i \bar{w}_i f_i = \frac{\sum_i \bar{w}_i f_i}{\sum_i w_i}$$

Thus we have constructed an ANFIS system that is functionally equivalent to a first-order Sugeno fuzzy model, which it will be used in the present QSAR study due to its transparency and efficiency.

C. Hybrid Learning Algorithm. From the proposed ANFIS architecture (Figure 3), it is observed that given the values of premise parameters, the overall output can be expressed as linear combinations of the consequent parameters. More precisely, the output f in Figure 3 can be rewritten as

$$f = \frac{w_1}{w_1 + w_2} f_1 + \frac{w_2}{w_1 + w_2} f_2 = \bar{w}_1(p_1 x + q_1 y + r_1) + \bar{w}_2(p_2 x + q_2 y + r_2)$$

$$= (\bar{w}_1 x)p_1 + (\bar{w}_1 y)q_1 + (\bar{w}_1)r_1 + (\bar{w}_2 x)p_2 + (\bar{w}_2 y)q_2 + (\bar{w}_2)r_2$$

which is linear in the consequent parameters p_1, q_1, r_1, p_2, q_2, and r_2. To train the above ANFIS system, the following error measure will be used:

$$E = \sum_{k=1}^{n} (f_k - \hat{f}_k)^2$$

where f_k and $\hat{f}_k$ are the kth desired and estimated outputs, and n is the total number of pairs (inputs-outputs) of data in the training data set. The learning algorithms of ANFIS consist of the following two parts: (a) the learning of the premise parameters by back-propagation and (b) the learning of the consequence parameters by least-squares estimation. More specifically, in the forward pass of the hybrid learning algorithm, functional signals go forward till layer 4 and the consequent parameters are identified by the least squares estimate. In the backward pass, the error rates propagate backward, and the premise parameters are updated by the gradient descent. During the learning process, the parameters associated with the membership functions will change. The computation of these parameters (or their adjustment) is facilitated by a gradient vector, which provides a measure of how well the fuzzy inference system is modeling the input/output data for a given set of parameters (see Appendix). Figure 4 represents three bell MFs for the input p_ch_2 (Table 2) and their parameters before and after training. Although all the input MFs undergo changes during learning, the p_ch_2 presented the most pronounced ones.

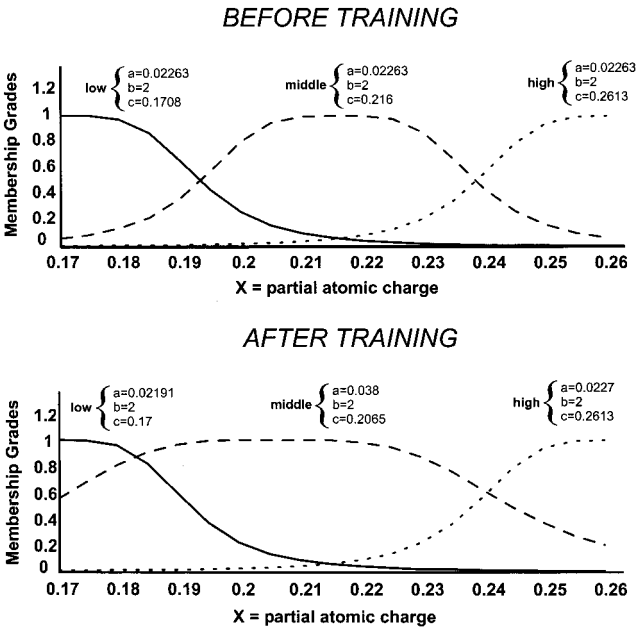

Figure 4. Three bell MFs and their parameters for the input p_ch_2 before and after training.

III. Experimental Section

A. Data Set. In the present paper we have performed a QSAR study for the inhibition of DHFR by 2,4-diamino-5-(substituted-benzyl)pyrimidines (Figure 5) using the data in Table 2. The first step in developing the model was the numerical description of the molecular structures by defining descriptors. To calculate the descriptors, the strain energy of the molecular structures must first be minimized. The initial conformation was based on the geometry of the benzylpyrimidine moiety (the common pharmacophore of the pyrimidine derivatives) obtained from the crystallographic analysis[11] of the complex of *Escherichia coli* with trimethoprim (compound number **58** in Table 2). For the flexible benzyl substituents, a geometry optimization was performed using the semiempirical calculations based on the AM1 Hamiltonian. The QSAR+ module of Cerius2[12] was used to calculate a total of 38 properties (Table 3) for each of the compounds. These properties, which were chosen to reflect the structural diversity in the data set, included 10 whole-molecule properties and 28 properties for individual atoms or substituents. The resulting data set with 38 properties and the activity values for each molecule were used for analysis.

B. Future Selection. Genetic algorithms[13] are general purpose search algorithms based upon the principles of evolution observed in nature. Genetic algorithms combine selection, crossover, and mutation operators with the goal of finding the best solution to a problem. Genetic algorithms search for this optimal solution until a specified termination criterion is met. The solution to a problem is called a chromosome. A chromosome is made up of a collection of genes, which are simply the parameters (the descriptors in the present study) to be optimized. For example, if there are six original input variables, the chromosome 001101 indicates that the first, second, and fifth variables should be discarded, and the third, fourth, and sixth ones kept. A genetic algorithm creates randomly an initial population (a collection of chromosomes), evaluates this population, and then evolves the population through multiple generations (using the genetic operators selection, crossover, mutation) in the search for a good solution for the problem at hand.

Selection is a genetic operator that chooses a chromosome from the current generation's population for inclusion in the next generation's population. Before making it into the next generation's population, selected chromosomes may undergo crossover and/or mutation (depending upon the probability of crossover and mutation) in which case the offspring chromo-

Table 2. Structures, Physicochemical Parameters, and Observed Inhibitory Activities of the Pyrimidine Derivatives[a]

	X	MW_mol	Vol_mol	MW_3	p_ch_2	p_ch_4	p_ch_sum	activity
				Training Subset				
1	4-$O(CH_2)_5CH_3$	300.450	234.252	1.010	0.174	0.179	−1.730	6.070
2	4-$O(CH_2)_6CH_3$	314.480	247.527	1.010	0.173	0.179	−1.730	6.100
3	4-NO_2	245.270	169.442	1.010	0.192	0.200	−1.759	6.200
4	3-$O(CH_2)_7CH_3$	328.510	262.161	129.250	0.174	0.180	−1.736	6.250
5	3-CH_2OH	230.300	168.110	31.040	0.176	0.191	−1.817	6.280
6	4-NH_2	215.290	156.848	1.010	0.173	0.189	−1.789	6.300
7	3,5-$(CH_2OH)_2$	260.330	187.395	31.040	0.177	0.189	−1.739	6.310
8	3-$O(CH_2)_6CH_3$	314.480	247.088	115.220	0.171	0.179	−1.732	6.390
9	4-$CH_2CH_2OCH_3$	274.360	202.957	1.010	0.174	0.179	−1.725	6.400
10	4-OH	216.270	155.715	1.010	0.172	0.177	−1.730	6.450
11	3,4-$(OH)_2$	232.270	162.391	17.010	0.177	0.194	−1.611	6.460
12	3-$OCH_2CH_2OCH_3$	274.360	203.024	75.100	0.172	0.178	−1.727	6.530
13	3-OCH_2CONH_2	273.330	197.877	74.070	0.178	0.182	−1.734	6.570
14	4-OCF_3	284.270	191.998	1.010	0.181	0.194	−1.693	6.570
15	3-CH_3	214.300	161.048	15.040	0.176	0.192	−1.830	6.700
16	4-$N(CH_3)_2$	243.350	183.503	1.010	0.172	0.189	−1.768	6.780
17	3-$O(CH_2)_3CH_3$	272.390	208.130	73.130	0.174	0.180	−1.735	6.820
18	4-Br	279.160	166.804	1.010	0.178	0.184	−1.848	6.820
19	3-OH, 4-OCH_3	246.300	176.208	17.010	0.176	0.195	−1.606	6.840
20	3-$O(CH_2)_5CH_3$	300.450	234.263	101.190	0.174	0.180	−1.734	6.860
21	4-$NHCOCH_3$	257.330	189.002	1.010	0.179	0.193	−1.731	6.890
22	4-$O(CH_2)_3CH_3$	272.390	208.328	1.010	0.173	0.179	−1.730	6.890
23	4-$OCH_2C_6H_5$	306.400	227.160	1.010	0.174	0.180	−1.727	6.890
24	3-OCH_3	230.300	168.988	31.040	0.176	0.194	−1.754	6.930
25	4-C_6H_5	276.370	208.075	1.010	0.179	0.194	−1.779	6.930
26	3-NO_2, 4-$NHCOCH_3$	302.330	208.283	46.010	0.182	0.186	−1.570	6.970
27	3-$OCH_2C_6H_5$	306.400	220.328	107.140	0.175	0.180	−1.730	6.990
28	3-CF_3	268.270	185.457	69.010	0.183	0.197	−1.807	7.020
29	3,5-$(CH_3)_2$	228.330	173.659	15.040	0.175	0.191	−1.562	7.040
30	3,4-OCH_2O	244.280	169.976	232.270	0.179	0.194	−1.580	7.130
31	3,5-$(OCH_3)_2$, 4-$O(CH_2)_7CH_3$	388.570	304.401	31.040	0.172	0.180	−1.426	7.200
32	3-I	326.160	161.638	126.900	0.210	0.218	−1.983	7.230
33	3-OCH_2CH_3,4-$OCH_2C_6H_5$	350.460	262.850	45.070	0.174	0.179	−1.562	7.350
34	3,5-$(OC_3H_7)_2$	316.450	241.090	59.100	0.174	0.180	−1.604	7.410
35	3-OCH_3, 4-OH	246.300	175.090	31.040	0.176	0.195	−1.600	7.540
36	3,5-$(OCH_2CH_3)_2$, 4-pyrryl	368.480	277.371	45.070	0.176	0.194	−1.528	7.660
37	3,5-$(OCH_2CH_3)_2$	288.390	215.327	45.070	0.173	0.179	−1.603	7.690
38	3-OC_2H_5, 5-OC_3H_7	302.420	227.672	45.070	0.173	0.179	−1.603	7.690
39	3-CF_3, 4-OCH_3	298.300	204.632	69.010	0.181	0.197	−1.675	7.690
40	3,5-$(OCH_3)_2$, 4-$N(CH_3)_2$	303.410	225.370	31.040	0.180	0.195	−1.496	7.710
41	3,5-$(OCH_3)_2$	260.330	188.877	31.040	0.175	0.195	−1.626	7.710
42	3-OCH_3, 4-$OCH_2CH_2OCH_3$	304.390	224.573	31.040	0.174	0.179	−1.561	7.770
43	3-OSO_2CH_3, 4-OCH_3	324.390	224.373	95.100	0.173	0.183	−1.508	7.800
44	3,4,5-$(CH_2CH_3)_3$	284.450	227.912	29.070	0.174	0.191	−1.689	7.820
45	3-OCH_3, 4-OSO_2CH_3	324.390	225.800	31.040	0.178	0.184	−1.569	7.940
46	3,5-$(OCH_3)_2$, 4-SCH_3	306.420	220.353	31.040	0.174	0.179	−1.653	8.070
47	3,5-$(OCH_3)_2$, 4-$C(CH_3)=CH_2$	300.400	226.370	31.040	0.173	0.178	−1.495	8.120
48	3,5-$(OCH_3)_2$, 4-$O(CH_2)_2OCH_3$	334.420	247.077	31.040	0.177	0.195	−1.435	8.350
				Validation Subset				
49	4-F	218.260	142.589	1.010	0.209	0.217	−1.742	6.350
50	4-Cl	234.710	163.399	1.010	0.175	0.180	−1.774	6.450
51	4-CH_3	214.300	148.062	1.010	0.208	0.216	−1.868	6.480
52	3-$CH_2O(CH_2)_3CH_3$	286.420	222.556	87.160	0.177	0.197	−1.817	6.550
53	3-CH_2OCH_3	244.330	181.558	45.070	0.176	0.190	−1.816	6.590
54	4-OSO_2CH_3	294.360	204.315	1.010	0.172	0.176	−1.667	6.600
55	4-OCH_3	230.300	168.799	1.010	0.175	0.192	−1.749	6.820
56	3,4-$(OCH_2CH_2OCH_3)_2$	348.450	259.744	75.100	0.174	0.179	−1.560	7.220
57	3-OCH_3, 4-$OCH_2C_6H_5$	336.430	250.619	31.040	0.174	0.179	−1.562	7.530
58	3,4,5-$(OCH_3)_3$	290.360	211.683	31.040	0.173	0.181	−1.413	8.080
				Test Subset				
59	H	200.270	135.243	1.010	0.208	0.216	−1.930	6.180
60	3-F	218.260	142.534	19.000	0.210	0.217	−1.748	6.230
61	3-OH	216.270	141.908	17.010	0.209	0.217	−1.799	6.470
62	3-C1	234.710	151.013	35.450	0.210	0.218	−1.831	6.650
63	3-OSO_2CH_3	294.360	194.705	95.100	0.261	0.256	−1.712	6.920
64	3-Br	279.160	155.582	79.900	0.210	0.218	−1.898	6.960
65	3-$O(CH_2)_7CH_3$, 4-OCH_3	358.540	284.236	129.250	0.173	0.178	−1.567	7.160
66	3-$OCH_2C_6H_5$, 4-OCH_3	336.430	252.409	107.140	0.174	0.179	−1.567	7.660
67	3,4-$(OCH_3)_2$	260.330	188.696	31.040	0.175	0.195	−1.594	7.720
68	3,5-$(OCH_3)_2$, 4-Br	339.220	211.141	31.040	0.175	0.182	−1.562	8.180

[a] The training, validate, and testing subsets are derived from the D-optimal design and the Kohonen self-organized map.

some(s) are actually the ones that make it into the next generation's population. Crossover is a genetic operator that combines (mates) two chromosomes (parents) to produce a new chromosome (offspring). The idea behind crossover is that the new chromosome may be better than both of the parents if it takes the best characteristics from each of the parents. Crossover occurs during evolution according to a user-definable crossover probability. This probability should usually be set fairly high (0.9 is a good first choice). Mutation is a genetic

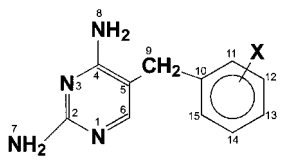

Figure 5. Structure of 2,4-diamino-5-(substituted-benzyl)-pyrimidines.

Table 3. Properties Calculated with the QSAR+ Module of Cerius2

(a)Whole-Molecule Properties	
MME	molecular mechanics energy (kcal)
MW_mol	molecular weight (atomic units)
DIPOLE	dipole moment (De)
Vol_mol	total molecular volume (Å^3)
SUR_mol	total surface area of molecule (Å^2)
HOMO	energy of HOMO (AM1) (eV)
LUMO	energy of LUMO (AM1) (eV)
ELEC	total electronic energy (AM1) (eV)
HEAT	heat of formation (AM1) (kcal)
logP_mol	logP of the whole molecule
(b) Atom and Substituent Properties	
MW_n	molecular weight of benzyl ring substituents at positions 3, 4, and 5
p_ch_n	partial atomic charges of heavy atoms ($n = 1-15$) (au) (see Figure 5)
p_ch_sum	sum of atomic charges of heavy atoms (1−15) (au)
VOL_	total substituents volumes at positions 3, 4, and 5 of benzyl ring (Å^3)
SUR_	total substituents surface area at positions 3, 4, and 5 of benzyl ring (Å^2)
logP_	π values of substituents at positions 3, 4, and 5 of benzyl ring

operator that alters one or more gene values in a chromosome from its initial state. This can result in entirely new gene values being added to the gene pool. With these new gene values, the genetic algorithm may be able to arrive at a better solution than was previously possible. Mutation is an important part of the genetic search as helps to prevent the population from stagnating at any local optima. Mutation occurs during evolution according to a user-definable mutation probability. This probability should usually be set fairly low (0.01 is a good first choice). If it is set too high, the search will turn into a primitive random search. Over a period of generations, successively better chromosomes are produced. In the present study, the size of the population was 50, the probability of crossover was 0.9, the probability of mutation was 0.01, and the number of evolution generations was 100. For each set of data, 200 runs were performed.

All the produced chromosomes were evaluated through an ANFIS system comprised of either two bell MFs or two Gaussian MFs, trained with five epochs each one. The chromosome with the best predictive ability was selected, which consisted of the six descriptors appearing in Table 2. Once the significant descriptors were isolated, the system was ready for analysis with the ANFIS system.

C. Training−Validation−Testing Subsets. Once the data set and the input variables were selected, the next step was the division of the data set in three subsets, namely the training, validation, and test subsets. The main requirement during training is the data representativity, meaning that the samples in the data set should be (evenly) spread over the expected range of data variability. To avoid the risk of not selecting representative samples during training, we evaluated two different strategies of training set design suggested by Massart,[14] namely the D-optimal design and the Kohonen self-organizing map approach.

Briefly, D-optimal designs are performed whenever the classical symmetrical designs cannot be used, because the experimental region is not regular in shape or the number of experiments selected by a classical design is too large. The principle of this method is to select the experimental points to maximize the determinant of the information matrix |X′X|. This matrix is equal to the variance covariance matrix when X is defined as a matrix with n' objects and m' variables after centering (where n' is the number of samples to be selected). The determinant of this matrix is maximal when the selected objects span the space of the whole data, in other words, when applied it aims to select the most influential samples (maximal spread). We apply Fedorov's algorithm[14] with the initial points selected by the Sequential or Dykstra method,[15] starting with an empty design, searching through the candidate list of samples, and choose in each step the one that maximizes the chosen criterion. There are no iterations involved, and the requested number of points will simply be picked sequentially. The D-optimality method selects the samples for the linear model $y = \sum b_i x_i + e$ where x_i is the input variable i. From this procedure, the samples indicated in Table 2 were chosen and were included in the training data set.

Next, a projective technique, the Kohonen network, was adopted to select the training cases. Simon[16] found that Kohonen self-organizing maps performed best in a similar task. The main goal of the Kohonen neural network is to map objects from n-dimensional into two-dimensional space. Objects with similar properties in the original space will map to the same node. In the present study a (4 × 4) Kohonen network was chosen containing 16 nodes. The learning rate was above 0.1 at the beginning and was linearly decreased to reach 0.01 at the end. The neighborhood size was also decreased linearly to reach a minimum of 1 after half of the training cycles and to remain 1 for the rest of the training. After stabilization of the network, it was observed that the samples chosen above were spread in all of the activated nodes denoting again the representation of the whole space. Accordingly, the samples for the validation and test subsets (Table 2) were selected in the same unbiased way in order to evaluate the predictive ability of ANFIS in the whole experimental space.

D. Linear and Nonlinear Multivariate Regression. The predictive ability of the examined ANFIS was compared further to that of classical multivariate regression.[17] A popular technique for multivariate regression is the partial least squares (PLS) regression with cross-validation as an important concept to identify the appropriate number of factors (or latent variables, *lv*) to use. Generally, PLS can be used to develop regression models that relate a number of independent predictor variables (X-block) to one or more dependent or predicted variables (Y-block). It relies on a decomposition of the X-block (the four descriptors in the present study) based on covariance criteria. PLS finds factors (latent variables) that are descriptive of X-block variance and are correlated with the Y-block (activities). PLS is advantageous to ordinary multiple linear regression (MLR) since it examines for collinearities in the predictor variables (i.e., some variables are linear combinations of other variables). The PLS models converges to MLR solution if all latent variables are included in the model. There are several ways to calculate PLS models, with the most commonly used being the noniterative partial least squares (NIPALS) and the SIMPLS algorithms, both of them giving exactly the same results for univariate y (as in the present study). The computational approaches for these two algorithms is well described in textbooks, and it is beyond the scope of the present study. The polynomial PLS model works just like the linear PLS using the same algorithms, except that once a pair of latent vectors is calculated, a polynomial of specific degree n is used to calculate the inner relation, replacing the b scalar for each latent variable with a **b** vector of polynomial coefficients. In the outputs of the function, **b** is a matrix ($n + 1$ by *lv*). In the present study it was confirmed that a degree of 2 generalizes better than the higher degree polynomials.

IV. Results and Discussion

We have performed a QSAR study for the inhibition of DHFR by 2,4-diamino-5-(substituted-benzyl)pyrimidines (Figure 5) using the data in Table 2. This particular set of compounds has been studied by other groups,[18−20] and it is ideal for the purpose of comparison. The fuzzy logic system was simulated using Matlab Fuzzy Logic Toolbox and run on a Pentium II platform. Training continued until there was no further decrease in validation error, resulting in average training time from a couple of minutes to a few seconds in some cases (see later), compared to 2 h training using other algorithms. Six input units (the six descriptors) resulted from the GA/ANFIS method, and one output unit (the activity) was simulated in all cases. The quality of QSAR

Table 4. ANFIS Trained with Hybrid Backpropagation and Using the Grid Partition Method for Specifying the Membership Functions

no MF	MF type	MSETr	MSETe	outliers[a]
2	bell	0.116	0.125	2
2	Gaussian	0.105	0.026	2
2	triangle	0.227	0.235	3
2	trapezoidal	0.255	0.244	3
3	bell	0.02	0.315	4
3	Gaussian	0.03	0.255	3
3	triangle	0.06	0.266	3
3	trapezoidal	0.07	0.288	4

[a] An outlier was arbitrarily defined as that compound having $|\text{activity}_{obs} - \text{activity}_{pred}| > 0.25$.

was assessed by three statistical variables (training MSETr and testing MSETe mean square errors and the number of outliers − the term outlier was adopted form ref 20; for its definition see the footnote of Table 4):

$$\text{MSE} = \sum_{i=1}^{N}\sum_{j=1}^{g}\frac{(y_{ij} - \text{out}_{ij})^2}{Ng}$$

where N is the number of objects in the examined data set (train, validate, or test), g is the number of output variables, y_{ij} is the element of target matrix **y** ($N \times g$) for the data considered (i.e., training, validate, or test set) and out_{ij} is the element of the output matrix **out** ($N \times g$) of the ANFIS.

The QSAR prediction problem is a typical multivariable nonlinear regression problem where several attributes (descriptors) are used to predict another continuous attribute (biological activity). To proceed with the ANFIS design, we divided the data set, in the unbiased way described above, into a training set of 48 data, a validation set of 10 data, and a testing set of 10 data; the training data set is used for model building, while the validation data set is for model validation. The resultant model is not biased toward the training data set, and thus it is likely to have a better generalization capability for unseen data. Model validation is the process by which the input vectors from input/output data sets on which the FIS was not trained are presented to the trained ANFIS model to see how well the ANFIS model predicts the corresponding data set output values. When validation data is presented to ANFIS as well as training data, the ANFIS model is selected to have parameters associated with the minimum validation data model error. The basic idea behind using a validation data set for model validation is that after a certain point in the training, the model begins overfitting the training data set. In principle, the model error for the validation data set tends to decrease as the training takes place up to the point that overfitting begins, and then the model error for the validation data suddenly increases.

A. Grid Partition.[21,22] ANFIS modeling involves two phases: structure identification and parameter identification. The former is related to finding a suitable number of rules and a proper partition of the feature space. The latter is concerned with the adjustment of system parameters, such as the MF parameters, the linear coefficients, and so on. The problem related to parameter identification was discussed before (learning procedure). The next step to proceed for structure identification is the input partition; to this end the grid partition takes place first, the simplest input partition style. The gridding partition method refers to the manual selection of the number and the type of MFs. Consider the simplest case in the examined QSAR model, taking into account only two input variables, the molecular weight (MW) and molecular volume (Vol). We start with two MFs per input, giving the linguistic labels "low" and "high". With grid partition, each MF of the first input is examined with each MF of the second input, resulting the following four rules:

Rule 1: If MW is low and Vol is low, then $\text{activity}_1 = p_1x + q_1y + r_1$

Rule 2: If MW is low and Vol is high, then $\text{activity}_2 = p_2x + q_2y + r_2$

Rule 3: If MW is high and Vol is low, then $\text{activity}_3 = p_3x + q_3y + r_3$

Rule 4: If MW is high and Vol is high, then $\text{activity}_4 = p_4x + q_4y + r_4$

where $\{p_i, q_i, r_i\}$ are the consequence parameters and $\{a_i, b_i, c_i\}$ are the premise parameters where $i = 1, 2, 3, 4$ (Figure 3). If we had decided three MF per input (with the linguistic labels "low", "middle", and "high"), the rules would be nine, which are presented schematically in Figure 6. The nine rules result simply by combining each of the three MF of MW with each of the three MF of Vol. It is concluded that by increasing the number of MFs per input the number of rules is increased accordingly. Generally, for a study with m inputs by using n rules per input, the gridding partition leads to n^m rules. In the present study the grid partition was examined with two or three MFs (of different type) per input, and the results appear in Table 4. From Table 4 it becomes evident that two Gaussian MFs per input resulted in the lower testing error, and the best predictions to unseen data were with only two outliers. The reduction of outliers is very important, suggesting accurate predictions for all the compounds. The ANFIS was trained instantly (couple of seconds) with 20 epochs, as was suggested from the validation data set. From Table 4 it also becomes evident that the performance of the ANFIS was reduced significantly by increasing the number of MFs to 3 with a simultaneous increase in the testing error and the number of outliers (poor generalization).

B. Subtractive Algorithm. The advantage of fuzzy approach over traditional ones lies in the fact that fuzzy system does not require a detailed mathematical description of the system while modeling. In the design of fuzzy systems, if the number of MFs is large (and therefore the number of fuzzy rules is large), the system requires large computation time and invokes the so-called *curse of dimensionality*. Moreover, the huge rule base may overfit the system and cause the system to lose the capability of generalization. Furthermore, data sets generated from studies such as QSAR often contain conflicting subsets and may not cover the entire input space adequately. It is important to note that an effective partition of the input space can decrease the number of rules and thus increase the speed in both learning and application phases.

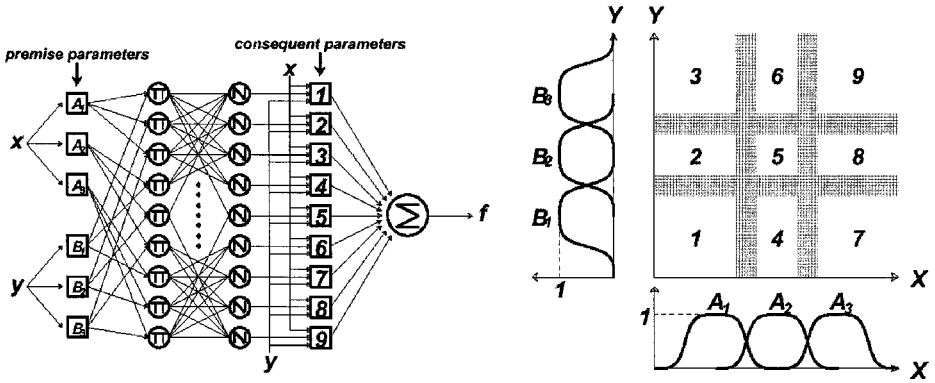

Figure 6. (a) ANFIS architecture for a two-input Sugeno fuzzy model with nine rules; (b) the input space that is grid partitioned into nine fuzzy regions. A_1, A_2, A_3 and B_1, B_2, B_3 correspond to linguistic labels for the inputs x and y, respectively.

A method that provides for some dimension reduction in the fuzzy inference system was used in the present study as an alternative to gridding partition. This method generates an ANFIS structure using the clustering algorithm *subtractive clustering*[23] and generates an FIS with the minimum number of rules required to distinguish the fuzzy qualities associated with each of the clusters. The purpose of clustering is to identify natural groupings of data from a large data set to produce a concise representation of a system's behavior. Given a collection of n data points $\{x_1, ..., x_n\}$ in a p-dimensional space, the subtractive clustering algorithm considers each data point as a candidate for cluster centers. A *density measure*[23] at a data point x_i is defined as

$$D_i = \sum_{j=1}^{n} \exp\left(-\frac{||x_i - x_j||^2}{(r_a/2)^2}\right)$$

where r_a is a positive constant. Hence, a data point will have a high-density value if it has many neighboring data points. The radius r_a defines a neighborhood; data points outside this radius contribute only slightly to the density measure. After the density measure of each data point has been calculated, the data point with the highest density measure is selected as the first cluster center. Let x_{c1} be the point selected and D_{c1} its density measure. Next, the density measure for each data point x_i is revised by the formula

$$D_i = D_i - D_{c1} \exp\left(-\frac{||x_i - x_{c1}||^2}{(r_b/2)^2}\right)$$

where r_b is a positive constant. Therefore, the data points near the first cluster center x_{c1} will have significantly reduced density measures, thereby making the points unlikely to be selected as the next cluster center. The constant r_b is normally larger than r_a to prevent closely spaced cluster centers; generally r_b is equal to $1.5r_a$ as suggested in ref 22. After the density measure for each data point is revised, the next cluster center x_{c2} is selected, and all of the density measures for data points are revised again. A sophisticated stopping criterion for automatically determining the number of clusters (which is implemented in Matlab Fuzzy logic toolbox) can be found in ref 23.

Table 5. ANFIS Trained with Hybrid Back-Propagation and Using the Subclustering Algorithm with Different Radius Values

radius value	MSETr	MSETe	outliers
0.1	0.0011	0.334	4
0.2	0.0025	0.312	3
0.3	0.0089	0.375	4
0.4	0.0125	0.355	3
0.5	0.0255	0.235	3
0.6	0.0811	0.145	2
0.7	0.143	0.255	3
0.8	0.245	0.478	5
0.9	0.445	0.496	5
1	0.578	0.525	6

When applying subtractive clustering to a set of input−output data, each of the cluster centers represents a prototype that exhibits certain characteristics of the system to be modeled. These cluster centers would be reasonably used as the centers for the fuzzy rules' premise in a Sugeno fuzzy model. For instance, assume that the center for the ith cluster is c_i in an M dimension. The c_i can be decomposed into two component vectors p_i and q_i, where p_i is the input part and it contains the first N element of c_i; q_i is the output part and it contains the last $M-N$ elements of c_i. Then, given an input vector x, the degree to which fuzzy rule i is fulfilled is defined by

$$\mu_i = \exp\left(-\frac{||x - p_i||^2}{(r_a/2)^2}\right)$$

Once the premise part has been determined, the consequence part can be estimated by the least-squares method (see Appendix).

The variable radius, with values between 0 and 1, specifies a cluster center's range of influence in each of the data dimensions. Good values for radius are usually between 0.1 and 0.6.[23] A trial and error procedure concerning scalar radius values appears in Table 5. It becomes evident from Table 5 that by increasing the radius from 0.1 to 0.6 the training error increases (from 0.0011 to 0.0811) with a simultaneous decrease in testing error (from 0.334 to 0.145). Increasing the radius further from 0.6 to 1 the training error increases further to 0.578 and the testing error to 0.525 (underfitting). Therefore, it is concluded that the radius of 0.6 resulted in the best performance in terms of speed (instant training) and accuracy (two outliers).

Table 6. Test Set of Pyrimidines Derivatives, the Observed (Experimental) Activities, the Calculated Ones Using ANFIS (with Two Gaussian MFs—See Table 4), Linear and Quadratic PLS, and the Published Results

pyrimidine derivatives	obsvd	ANFIS	quad. PLS	linear PLS	published (ref 19)	published (ref 18)
59	6.180	6.370	6.410	6.612	6.180	6.210
60	6.230	6.410	6.520	6.720	6.300	6.290
61	6.470	6.620	6.710	6.880	6.230	6.390
62	6.650	6.950	6.900	6.980	7.040	6.810
63	6.920	7.125	7.150	7.200	6.610	6.860
64	6.960	6.950	7.150	7.400	6.970	6.980
65	7.160	7.320	7.520	7.600	7.120	6.910
66	7.660	7.910	7.980	8.300	7.430	8.270
67	7.720	8.270	8.700	8.550	7.830	7.280
68	8.180	8.350	8.400	8.500	8.150	7.940
% RSEP		2.926	4.061	5.508	3.605	5.300
q^2		0.909	0.825	0.678	0.862	0.702
$^1s_{cv}$[a]		0.123	0.056	0.117	0.139	0.196

[a] $^1s_{cv}$: standard deviation of errors of predicted values.

C. Generalization. The term generalization means the ability of the examined models to predict the outputs in unseen data (test data set). Although the examined ANFIS models were compared using the MSE term, in Table 6 the relative standard error for predictions (%RSEP) and the cross-validation[24] (q^2), statistical terms for comparing the performance of the examined models (ANFIS, PLS, and published results[18−20]) in the same data set, were also calculated:

$$\%\,\mathrm{RSEP} = \sqrt{\frac{\sum_1^n (y_{\mathrm{obs}} - y_{\mathrm{pred}})^2}{\sum_1^n y_{\mathrm{obs}}{}^2}} \times 100$$

$$q^2 = 1 - \frac{\sum_1^n (y_{\mathrm{obs}} - y_{\mathrm{pred}})^2}{\sum_1^n (y_{\mathrm{obs}} - \bar{y}_{\mathrm{obs}})^2}$$

where y_{obs} are the assayed activities of the molecules, $\bar{y}_{\mathrm{obs}}$ is the mean of y_{obs}, and y_{pred} are the predicted from the examined models molecular activities. The numerator of the q^2 formula is the squared errors of the predictions, and the denominator is a measure of how much variation there is in the actual activities. The q^2 values have been determined solely in the test subset of 10 compounds (Table 6) as this method (external validation)[25−27] provides the most unbiased measure of the predictive ability of the examined models. In neurofuzzy systems, if there are enough data for splitting in train-validate-test subsets (as in the present study—see above), the *leave one out* method is not recommended[26,27] for comparing the predictive abilities of the examined models.

i. ANFIS. From Tables 4 and 5 it is becoming evident that the best configuration for ANFIS was the one with two Gaussian MFs trained with hybrid back-propagation. During the testing procedure (or generalization), the parameters of the 10 compounds unknown to ANFIS were put into the network and the predictive activities of these compounds were obtained (Table 6). It is estimated that the number of outliers in the test data set dropped to 1, reaching a significant level of predictive accuracy to new (unseen) data. In light of this, one can be confident that the ANFIS can be optimized to provide reliable predictions of biological activities of novel pyrimidine variants. Regarding the improved predictions, another benefit of ANFIS should be mentioned: the time for training using both the grid method and the subtractive clustering was decreased dramatically, from a couple of hours mentioned elsewhere to a few seconds, giving the medicinal chemist the opportunity to examine many more ANFIS models in a reasonable period of time.

ii. Linear and Quadratic[28−31] PLS Models. PLS models attempt to maximize covariance (to do both, capture variance and achieve correlation). The question that arises is: how many factors (latent variables) should be chosen? In the case of linear PLS, five latent variables (*lv*) captured 51.45% variance of the dependent variable *y* (activity) while in the case of quadratic PLS, three *lv* captured 68% variance of *y*. The predictive abilities of linear and quadratic PLS are shown in Table 6. It is becoming evident that quadratic PLS outperformed the linear PLS as well as the nonlinear regression[19] while it was worse than the feedforward neural networks.[20]

In Hansch's earlier analysis[19] on this set of compounds the following correlation equation was formulated:

$$\log(1/K_i) = 0.95(\pm 0.24)\mathrm{MR}_5{}' + 0.89(\pm 0.27)\mathrm{MR}_3{}' + 0.80(\pm 0.22)\mathrm{MR}_4 - 0.21(\pm 0.07)\mathrm{MR}_4{}^2 + 1.58(\pm 0.73)\pi_3{}' - 1.77(\pm 0.80)\log(\beta_3 10^{\pi_3{}'} + 1) + 6.65(\pm 0.36)$$

where

$$\log\beta_3 = 0.175,\ \mathrm{MR}_4{}^0 = 1.85(\pm 0.20),\ \text{and}\ \pi_3{}^0 = 0.73(\pm 1.06)$$

$$n = 68,\ r = 0.890,\ s = 0.290,\ F_{3,60} = 19.8$$

In the equation above, K_i is the apparent inhibition constant, the π values represent the hydrophobic constants for substituents in the corresponding positions on the benzyl ring of pyrimidines, and MR is the molar refractivity of substituents. The parameter β is an adjustable parameter of the model, which helps to determine the optimum values of π and MR (π^0 and MR^0). Building a regression equation as complex as the equation above requires a laborious development phase and cannot be inspired by a flash of brilliance. In regression analysis, the inclusion of nonlinear terms is on trial and error basis. For ANFIS this is not necessary. The researcher may simply consider the descriptors, and ANFIS will find the optimum number and shapes of membership functions.

Although several hundreds of descriptors are available,[32] resulting sometimes in chance relationships in the data, in the present study the initial set of descriptors (Table 3) was based on the methodology proposed in other QSAR studies,[2,33,34] which suggest the selection of hydrophobic, electrostatic, and bulk descriptors for

both molecules and substituents as relevant to receptor affinity. From this initial set of descriptors (Table 3), the GA isolated as the most influential three electrostatic descriptors (sum of partial charges of heavy atoms as well as partial charges at positions 2 and 4—see Figure 5), two constitutional descriptors (the MW of whole molecules and the MW of substituents at position 3 of benzyl ring—meta substituents), and one geometrical descriptor (the molecular volume Vol_mol), while ignoring all the hydrophobic descriptors (logP or π values). The MR values at positions 3, 4, and 5 of the benzyl ring were eliminated as redundant due to high correlations with the respective MW, VOL, and SUR values at the same positions ($r^2 > 0.980$).

Comparing the six descriptor (Table 2) to Hansch's four descriptors[19] (see above) we could support that, in both studies as well as in another similar study,[35] the hydrophobic descriptors are considered insignificant for the inhibitory activity of benzylpyrimidines. Furthermore, while Hansch's descriptors[19] concentrate on the benzyl ring, other research groups[36,37] concentrate on the pyrimidine moiety based on the crystallographic analysis[11] as well as on NMR studies[38] of *E. coli* DHFR with trimethoprim which indicate that only the pyrimidine ring of trimethoprim is oriented deep in the active site cleft, while the phenyl moiety extends out toward the surface of DHFR. These studies[36−38] conclude that the potential inhibitory activity of new pyrimidines could be based mainly on electrostatic (intermolecular forces such as hydrogen bonds) and steric interactions (for relative precise matching with the active site of DHFR). The present QSAR model based on the descriptors of Table 2 could be considered in accordance with these prerequisites.

V. Conclusion

In this study, we recognized the well-known facts that the fuzzy logic can encode expert knowledge in a direct and easy way using rules with linguistic labels. The system also uses subjective definitions of membership functions, and wrong membership functions can lead to poor performance and possibly to instability. Therefore, learning techniques could be used to design membership functions automatically, thus reducing development time and cost while improving performance. In this paper, a two-phase neuro-fuzzy system called ANFIS is proposed to resolve problems of the traditional fuzzy models and neural networks. Within this structure, neural networks can improve their transparency, making them closer to fuzzy systems, while fuzzy systems can self-adapt, making them closer to neural networks. The two-phase adaptive neuro-fuzzy system is applied to nonlinear systems, including nonlinear prediction QSAR problems. The results show that our system leads to better predictions than other well-known approaches. The grid method as well as the subtractive algorithm trained almost instantly (a few seconds) with improved prediction capability, making ANFIS a powerful and instant QSAR predictor.

The main features and advantages of the ANFIS developed in this paper are as follows: (i) it is a general framework that combines two technologies, namely neural networks and fuzzy systems; (ii) by using fuzzy techniques, both numerical and linguistic knowledge can be combined into a fuzzy rule base; (iii) the combined fuzzy rule base represents the knowledge of the network structure so that *structure learning* techniques can easily be accomplished; (iv) fuzzy membership functions can be tuned optimally by using learning methods; (v) the architecture requirements are fewer and simpler compared to neural networks, which require extensive trails and errors for optimization of their architecture;[39] and (vi) ANFIS does not require extensive initializations through several random starts before training, as always happens in neural networks. Other advantages of the two-phase neuro-fuzzy hybrid technique in the ANFIS model also include its nonlinear ability, its capacity for fast learning from numerical and linguistic knowledge, and its adaptation capability. We expect that the ANFIS system should be considered further in respect to a wider range of QSAR problems such as classification and 3D QSAR. These applications are the subject of our ongoing research projects.

Acknowledgment. The author thanks the reviewers for their valuable discussions and suggestions, which have helped to improve the quality of this paper.

Appendix

i. Definition of the Gradient Vector. Let $x = [x_1, ..., x_n]^T$ be the parameters in an ANFIS system, and let $f(x)$ be a function of x (like the error function of section ii below), then the derivative of $f(x)$ with respect to x, called the *gradient vector* or *gradient* of $f(x)$, is a column vector denoted by

$$\nabla f(x) = \begin{bmatrix} \partial f(x)/\partial x_1 \\ \vdots \\ \partial f(x)/\partial x_n \end{bmatrix}$$

ii. Learning of the premise parameters. Back-propagation algorithm[40] is for multilayered neural networks. Let $\epsilon_{r,h}$ denote the back-propagation error of the hth node in the rth layer; then the error signal from the final output node for the kth entry can be calculated directly as

$$\epsilon_{5,1} = \frac{\partial (f_k - \hat{f}_k)^2}{\partial \hat{y}_k} = -2(f_k - \hat{f}_k)$$

The back-propagation error[26] from each node in layer 4 to layer 1 is

$$\epsilon_{r,h} = \sum_{t=1}^{M_{r+1}} \epsilon_{r+1,t} \frac{\partial g_{r+1,t}}{\partial O_{r,h}}$$

where $g_{r+1,t}$ is the node function at the rth node of the $(r+1)$th layer, $O_{r,h}$ the output of the hth node of the rth layer, and M_{r+1} is the total number of nodes in the $(r+1)$-th layer. The gradient vector is defined as the derivative of the error measure with respect to each parameter. If $\tilde{n}$ is a parameter of the hth node at layer r, we have

$$\frac{\partial (f_k - \hat{f}_k)^2}{\partial \rho} = \epsilon_{r,h} \frac{\partial g_{r,h}}{\partial \rho}$$

The derivative of the overall error measure E with respect to ρ is

$$\frac{\partial E}{\partial \rho} = \sum_{k=1}^{n} \frac{\partial (f_k - \hat{f}_k)^2}{\partial \rho}$$

Thus the updating formula for ρ is

$$\Delta \rho = -\eta \frac{\partial E}{\partial \rho}$$

in which η is a learning rate which can be further expressed as

$$\eta = \frac{k}{\sqrt{\sum_{\rho} \left(\frac{\partial E}{\partial \rho}\right)^2}}$$

where k is the step size, the length of each gradient transition in the parameter space. Usually, we can change the value of k to vary the speed of convergence. It is observed that if k is small, the gradient method will closely approximate the gradient path, but convergence will be slow since the gradient must be calculated many times. On the other hand, if k is large, convergence will initially be very fast, but the algorithm will oscillate about the optimum. On the basis of these observations, we update k according to the following two heuristic rules, which are suggested also in the literature[41] (Figure 7):

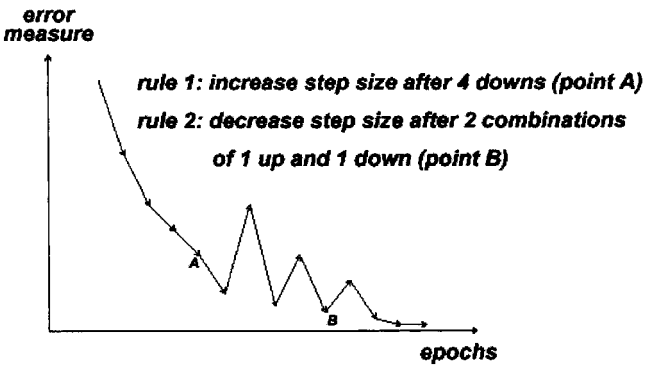

Figure 7. Two heuristic rules for updating step size k.

1. If the error measure undergoes four consecutive reductions, increase k by 10%.

2. If the error measure undergoes two consecutive combinations of one increase and one reduction, decrease k by 10%.

Furthermore, due to this dynamical update strategy, the initial value of k is usually not critical as long as it is not too big (in the present study $k = 0.01$).

iii. Learning the Consequence Parameters. The learning procedure of the consequence parameters is following the very well-known procedure of least-squares estimation which is described nicely in several references.[42−46]

References

(1) Kubinyi, H. Variable selection in QSAR studies. I. An evolutionary algorithm. *Quant. Struct.-Act. Relat.* **1994**, *13*, 285−294.
(2) So, S. S.; van Helden, S. P.; van Geerestein, V. J.; Karplus, M. Quantitative Structure−Activity Relationship Studies of Progesterone Receptor Binding Steroids. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 762−772.
(3) Sugeno, M. *Industrial applications of fuzzy control*; Elsevier: Amsterdam, 1985.
(4) Zadeh, L. A. Fuzzy sets. *Inf. Constr.* **1965,** *8***,** 338−353.
(5) Sugeno, M.; Kang, G. T. Structure identification of fuzzy model. *Fuzzy Sets Syst.* **1988**, *28*, 15−33.
(6) Kosko, B. *Fuzzy Engineering*; Prentice-Hall: Englewood Cliffs, NJ, 1996.
(7) Lee, C. C.; Fuzzy logic in control systems: fuzzy logic controller-part 1. *IEEE Trans. Syst. Man, Cybernetic*s **1990**, *20*, 404−418.
(8) Lee, C. C.; Fuzzy logic in control systems: fuzzy logic controller-part 2. *IEEE Trans. Syst., Man, Cybernetic*s **1990**, *20*, 419−435.
(9) Takagi, T.; Sugeno, M.; Derivation of fuzzy control rules from human operator's control actions. *Proc. of the IFAC Symp. on Fuzzy Information, Knowledge Representation and Decision Analysis*; 1983, pp 55−60.
(10) Takagi, T.; Sugeno, M.; Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst., Man, Cybernetic*s. **1985**, *15*, 116−132.
(11) Matthews, D. A.; Bolin, J.; Burridge, J.; Filman, D.; Volz, K.; Kaufman, B.; Beddell, C. Champness, J.; Stammers, D.; Kraut, J. Refined crystal structures of *Escherichia coli* and chicken liver dihydrofolate reductase containing bound trimethoprim. *J. Biol. Chem*. **1985**, *260*, 381−391.
(12) Cerius-2; Molecular Simulations Inc; Burlington, MA.
(13) Jouan-Rimbaud, D.; Massart, D. L.; de Noord, O. E. Random correlation in variable selection for multivariate calibration with a genetic algorithm. *Chemom. Intell. Lab. Syst.* **1996**, *35*, 213−220.
(14) Wu, W.; Walczak, B.; Massart, D. L.; Heuerding, S.; Erni, F.; Last, I. R.; Prebble, K. A., Artificial neural networks in classification of NIR spectral data: Design of the training set. *Chemom. Intell. Lab. Syst*. **1996**, *33*, 35−46.
(15) Dykstra, O., Jr. The augmentation of experimental data to maximize |X′X|. *Technometrics* **1971**, *13*, 682−688.
(16) Simon V.; Gasteiger, J.; Zupan, J. A combined application of two different neural network types for the prediction of chemical reactivity. *J. Am. Chem. Soc.* **1993**, *115*, 9148−9159.
(17) Massart, D. L.; Vandeginste, B. G. M.; Buydens, L. M. C.; de Jong, S.; Lewi, P. J.; Smeyers-Verbeke, J. Handbook of chemometrics and qualimetrics: Parts A and B, Elsevier Science B. V., Amsterdam, 1998.
(18) Hanch, C.; Li, R.; Blaney, J.; Langridge, R. Comparison of the inhibition of escherichia coli and lactobacillus casei dihydrofolate reductase by 2,4-diamino-5-(substituted-benzyl) pyrimidines: quantitative structure−activity relantionships, X-ray crystallography and computer graphics in structure−activity analysis. *J. Med. Chem*. **1982**, *25*, 777−784.
(19) Selassie, C. D.; Li, R.; Poe, M.; Hanch, C. On the optimization of hydrophobic and hydrophilic substituent interactions of 2,4-diamino-5-(substituted-benzyl) pyrimidines with dihydrofolate reductase. *J. Med. Chem*. **1991**, *34*, 46−54.
(20) 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.
(21) Berthold, M. R.; Huber, K. P. Constructing fuzzy graphs from examples, *Intell. Data Anal*. **1999**, *3* 37−53
(22) Jang, J. S. R.; Sun, C. T.; Mizutani, E. *Neuro-Fuzzy and soft computing*; Prentice-Hall: Englewood Cliffs, NJ, 1997.
(23) Chiu, S. Fuzzy Model Identification Based on Cluster Estimation. *J. Intell., Fuzzy Syst.* **1994**, *2*, 762−767.
(24) Cramer, R. D.; Patterson, D. E.; Bunch, J. D. Comparative Molecular Field Analysis (CoMFA). 1. Effect of Shape on Binding of Steroids to Carrier Proteins. *J. Am. Chem. Soc.* **1988**, *110*, 5959−5967.
(25) Shi, L.; Fang, H.; Tong, W.; Wu, J.; Perkins, R.; Blair, R.; Branham, W.; Dial, S.; Moland, C.; Sheehan, D. QSAR Models Using a Large Diverse Set of Estrogens. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 186−195.
(26) Despagne, F.; Massart, D. L. Neural networks in multivariate calibration. *Analyst* **1998**, *123*, 157−178.
(27) Bishop, C. M. *Neural networks for pattern recognition*; Oxford University Press Inc.: New York, 1995.
(28) Hasegawa, K.; Kimura, T.; Funatsu, K. Nonlinear CoMFA using QPLS as a novel 3D-QSAR approach. *Quant. Struct. Act. Relat.* **1997**, *16*, 219−223.
(29) Yoshida, H.; Funatsu, K. Optimization of the Inner Relation Function of QPLS Using Genetic Algorithm. *J. Chem. Inf. Comput. Sci*. **1997**, *37*, 1115−1121.
(30) Blanco, M.; Coello, J.; Iturriaga, H.; Maspoch, S.; Pages, J. Calibration in nonlinear near-infrared reflectance spectroscopy: a comparison of several methods. *Anal. Chim. Acta* **1999**, *384*, 207−214.
(31) Wold, S. *Chemometric Methods in Molecular Design*; van de Waterbeemd, H., Ed.; VCH: Weinheim, 1995.

(32) Labute, P. A. widely applicable set of descriptors. *J. Mol. Graphics* **2000**, *18*, 464−477.
(33) So, S.-S.; Karplus, M. Genetic neural networks for quantitative structure−activity relationships: Improvements and application of benzodiazepine affinity for benzodiazepine/GABA receptors. *J. Med. Chem.* **1996**, *39*, 5246−5256.
(34) Katritzky, A. R.; Lobanov, V. S.; Karelson, M. The Correlation and Quantitative Prediction of Chemical and Physical Properties from Structure. *Chem. Soc. Rev.* **1995**, 279−287.
(35) Czaplinski, K.; Hansel, W.; Wiese, M.; Seydel, J. New benzylpyrimidines: inhibition of DHFR from various species. QSAR, CoMFA and PC analysis, *Eur. J. Med. Chem.* **1995**, *30*, 779−787.
(36) Kuyper, L.; Garvey, J.; Baccanari, D.; Champness, J.; Stammers, D.; Beddell, R. Pyrolo[2,3-d]pyrimidines and pyrido[2,3-d]pyrimidines as conformationally restricted analoques of the antibacterial agent trimethoprim. *Bioorg. Med. Chem.* **1996**, *4*, 593−602.
(37) Chan, J. H.; Roth, B. 2,4-Diamino-5-benzylpyrimidines as antibacterial agents. 14. 2,3-Dihydro-1-(2,4-diamino-5-pyrimidyl)-1H-indenes as conformationally restricted analogues of trimethoprim. *J. Med. Chem.* **1991**, *34*, 550−555.
(38) Yang, Q.; Huang, F.; Lin, T.; Gelbaum, L.; Howell, E.; Huang, T. Dynamics of trimethoprim bound to dihydrofolate reductase-a deuterium NMR study. *Solid State Nucl. Mag.* **1996**, *7*, 193−201.
(39) Loukas, Y. L. Radial Basis Function Networks in Host:Guest Interactions: Instant and Accurate Formation Constant Calculations. *Anal. Chim. Acta* **2000**, *417*, 221−229.
(40) Rumelhart, D. E.; McClelland, J. L.; The PDP Research Group. In *Parallel Distributed Processing*; MIT Press: Cambridge, 1986; Vol. 1.
(41) Jang, J. S. R.; Sun, C. T. Neuro-fuzzy modeling and control. *Proc. IEEE.* **1995**, *83*, 378−406.
(42) Astrom, K. J.; Wittenmark, B. *Computer Controller Systems: Theory and Design*; Prentice-Hall: Englewood Cliffs, NJ, 1984.
(43) Goodwin, G. C.; Sin, K. S. *Adaptive filtering prediction and control*; Prentice-Hall: Englewood Cliffs, NJ, 1984.
(44) Kalman, R. E. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, *1*, 35−45.
(45) Ljung, L. *System identification: theory for the user*; Prentice-Hall: Englewood Cliffs, NJ, 1987.
(46) Strobach, P. *Linear prediction theory: a mathematical basis for adaptive systems*; Springer-Verlag: Berlin, 1990.

JM000226C